Voyager Data Dictionary for V7.2.5
Including Tricks for Using the Voyager Tables in Queries
April 8, 2013

This document is designed to be your first stop when you are looking for something in the Voyager tables. Since it lists all the tables and fields, you might be able to locate what you want by searching the document in Word or another text editor.

This document also serves as an index to the Voyager class diagrams (formerly called Entity-Relationship diagrams). Ex Libris has provided a set of class diagrams on Doc Portal. Use the V6.1 version with page numbers 1 through 30. CARLI has some additional diagrams, with page numbers 31-40, on the CARLI website. These are available at [http://www.carli.illinois.edu/products-services/i-share/reports](http://www.carli.illinois.edu/products-services/i-share/reports). The page numbers following the table names in this document will point you to the relevant class diagrams.

Finally, this document includes tips about how the tables relate to each other and how to understand what you find in the fields.

The fields for each table are listed in alphabetical order, except that the fields with “_id” in them are listed first. The “_id” fields are important because you will often use them to link tables to each other.

Fields marked with a star (*) are encoded in UTF-8. To make the diacritics and special characters display properly, you will need the utf8to16() function and a Unicode font.

**ACCESS_GROUP**
Data in this table are defined in the SysAdmin client at System, Access Control Groups, and display at OPAC Configuration, Holding Sort Groups, Access Control Groups tab.

For access_group_type, D=Domain Name, I=IP Address, R=IP Range, and P=Patron Group.

- access_group_id number
- access_group_code character 8
- access_group_name character 40
- access_group_type character 1

**ACCESS_GROUP_DATABASES**
This table does not appear to be used for anything.

- access_group_id number
- db_id number
- db_code character 8

**ACCESS_GROUP_DOMAIN**
Data in this table are defined in the SysAdmin client at System, Access Control Groups.

There will be data in this table when access_group_type in ACCESS_GROUP = D.

- access_group_domain_id number

**ACCESS_GROUP_IP**
Data in this table are defined in the SysAdmin client at System, Access Control Groups.

There will be data in this table when access_group_type in ACCESS_GROUP = I or R.

- access_group_id number
- access_group_ip_id number
- max_ip_addr number
- min_ip_addr number

**ACCESS_GROUP_PATRON_GROUP**
Data in this table are defined in the SysAdmin client at System, Access Control Groups.

- access_group_id number
- patron_group_id number

**ACCESS_GROUP_SORT_GROUP**
Data in this table are defined in the SysAdmin client at System, Access Control Groups.

- access_group_id number
- sort_group_id number

**ACCOUNT_LOCATION** p. 39

- account_id number
ACQ_LOCATIONS p. 31
Data in this table are defined in the SysAdmin client at Acquisitions, Policy Definitions, Locations tab.
acq_policy_id number
location_id number
destination_loc character 1
order_default_item_type number
order_loc character 1
order_opac character 1
print_location number
receive_default_item_type number
receive_loc character 1

ACQ_OPERATOR p. 31
Data in this table display in the SysAdmin client at Security, Acquisitions/Serials Profiles, Operator tab and display at the Operator, Current Profiles tab.
acq_profile_id number
operator_id character 10

ACQ_POLICY_GROUP p. 31, 39
Data in this table are defined in the SysAdmin client at Acquisitions, Policy Definitions, Policy tab.
acq_policy_id number
dup_profile_id number
acq_policy_name character 40

ACQ_PROFILE p. 31
Data in this table are defined in the SysAdmin client at Security, Acquisitions/Serials Profiles, Profile Values and Profile Values Cont. tabs, and display at the Operator, Current Profiles tab.
acq_profile_id number
acq_profile_name character 25
bind_edit_issue character 1
bind_edit_vol character 1
bind_print character 1
bind_view_vol_issue character 1
change_fund_alloc character 1

ACCOUNT NOTE p. 39
account_id number
currency_maintenance character 1
edi_incoming character 1
edi_outgoing character 1
fiscal_close character 1
hold_ignore_ownership character 1
invoice_add_update character 1
invoice_approve character 1
invoice_delete character 1
invoice_view_only character 1
item_add_update character 1
item_delete character 1
item_view_only character 1
ledger_add_update character 1
ledger_delete character 1
ledger_view_only character 1
modifyEDIoutgoing character 1
mono_claims character 1
order_add_update character 1
order_approve character 1
order_delete character 1
order_view_only character 1
override_commit character 1
override_expense character 1
pattern_add_update character 1
pattern_delete character 1
pattern_view_only character 1
problems_claims_view_only character 1
receive character 1
serial_checkin character 1
serial_claims character 1
serials_view_only character 1
vendor_add_update character 1
vendor_delete character 1
vendor_view_only character 1

ACQ_SECURITY_LOCS p. 31
Data in this table are defined in the SysAdmin client at Security, Acquisitions/Serials Profiles, Locations tab.
acq_profile_id number
location_id number

ACTION_TYPE p. 32, 33
This table is used with the AUTH_HISTORY, BIB_HISTORY, and MFHD_HISTORY tables.
action_type_id number
action_type character 20

ADDRESS_TYPE p. 20
This table is used with the PATRON_ADDRESS table.

The change_line_item_bib field is new in V7.0.
The starred field in this table is in UTF-8.

When you are searching record_segment, it is helpful to know that CHR(31) is the subfield delimiter, CHR(30) is the end of field delimiter, and CHR(29) is the end of record delimiter.

auth_id number
*record_segment character 990
seqnum number

AUTH_HEADING  p. 10, 32
The starred field in this table is in UTF-8.

auth_id number
heading_id_pointee number
heading_id_pointer number
*display_heading character 330
reference_type character 1
scope_note_present character 1

AUTH_INDEX  p. 14, 32
The starred fields in this table are in UTF-8.

This is the table that Voyager uses to index the 010 and 035 fields in authorities. The index_code tells you what fields and subfields are being indexed. A10A indexes 010$a, A10Z indexes 010$z, A350 and A35A both index 035$a, but they are formatted differently, and A35Z would index 035$z, but it appears that the LC authority records do not include this subfield. If you want to know more about what is indexed, take a look at the indexrules field in the SEARCHPARM table.

auth_id number
*display_heading character 150
index_code character 4
*normal_heading character 150

**AUTH_MASTER**  p. 10, 14, 32, 36
If a record has not been modified, the modify_date is null.
auth_id number
export_ok_location_id number
export_ok_opid character 10
create_date date
export_date date
export_ok character 1
export_ok_date date
update_date date

**AUTH_SUBDIVISION**  p. 10, 32
The starred field in this table is in UTF-8.
auth_id number
subdiv_id_pointee number
subdiv_id_pointer number
*display_subdiv character 330
reference_type character 1

**BASE_CURRENCY**
Data in this table are defined in the SysAdmin client at System, Base Currency.
base_country_name character 25
base_currency_code character 3
base_currency_name character 25
base_decimals number
decimal_delimiter character 1

**BIBCOMPOSITEINDEX_VW**
composite_searchcode character 4
component_searchcode character 4

**BIBHISTORY_VW**
bib_id number
create_location_id number
create_operator_id character 10
update_location_id number
update_operator_id character 10
create_date date
update_date date

**BIBLOC_VW**
bib_id number
marcloccode character 3

**BIBSORTING_VW**
This table provides a very convenient way to sort by author and/or title, but it is slow. If you need a faster solution, use BIB_INDEX.
bib_id number
display_author character 150
display_title character 150
normal_author character 150
normal_title character 150
pub_date character 4

**BIB_DATA**  p. 33
The starred field in this table is in UTF-8.
When you are searching record_segment, it is helpful to know that CHR(31) is the subfield delimiter, CHR(30) is the end of field delimiter, and CHR(29) is the end of record delimiter.
bib_id number
*record_segment character 990
seqnum number

**BIB_FACET**
This table is part of Voyager’s Geospatial module. We’re not using this module yet, so the table is not useful.
The starred fields in this table are in UTF-8.
bib_id number
*facet1 character 20
*facet2 character 20
*facet3 character 20
index_code character 4

**BIB_FORMAT_DISPLAY**
Data in this table are defined in the SysAdmin client at Search, Title List Material Type Display.
bib_format character 2
bib_format_display character 20

**BIB_HEADING**  p. 10, 33
The starred field in this table is in UTF-8.
bib_id number
heading_id number
*display_heading character 330
suppress_in_opac character 1

**BIB_HISTORY**  p. 33
There’s an error in some versions of the E-R diagrams. Action_type_id has a value between 1
and 6 and it is interpreted by linking to the ACTION_TYPE table.

When a bib record is deleted, its BIB_HISTORY records are deleted too.

Voyager’s marcexport utility uses the create_date and update_date in BIB_MASTER, not the action_date in BIB_HISTORY.

The encoding_level and suppress_in_opac are the values after the transaction.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>action_type_id</td>
<td>number</td>
<td>Action type identifier</td>
</tr>
<tr>
<td>bib_id</td>
<td>number</td>
<td>Bibliographic identifier</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
<td>Location identifier</td>
</tr>
<tr>
<td>operator_id</td>
<td>character 10</td>
<td>Operator identifier</td>
</tr>
<tr>
<td>action_date</td>
<td>date</td>
<td>Action date</td>
</tr>
<tr>
<td>encoding_level</td>
<td>character 1</td>
<td>Encoding level</td>
</tr>
<tr>
<td>suppress_in_opac</td>
<td>character 1</td>
<td>Suppress in OPAC</td>
</tr>
</tbody>
</table>

BIB_INDEX p. 14, 33

The starred fields in this table are in UTF-8.

This is the table that Voyager uses for left anchored searches and limits, so it is very useful for bibliographies. Titles, subjects, dates, languages, and many other fields are indexed here. The index_code gives you a clue as to what fields and subfields are being indexed. If you want to know precisely what is indexed, take a look at the indexrules field in the SEARCHPARM table. Note that the language code in the 008L index is lower case in both of the heading fields.

If you need data from a bib record that are not available in BIB_TEXT, check to see if they are available here. Using BIB_INDEX and BIB_TEXT is more efficient than using the BLOB functions.

The OCLC control number is indexed in BIB_INDEX in 2 ways. If index_code is 0350, normal_heading is “OCOLC 12345678”. If index_code is 035A, normal_heading is “12345678”.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td>Bibliographic identifier</td>
</tr>
<tr>
<td>*display_heading</td>
<td>character 150</td>
<td>Display heading</td>
</tr>
<tr>
<td>index_code</td>
<td>character 4</td>
<td>Index code</td>
</tr>
<tr>
<td>*normal_heading</td>
<td>character 150</td>
<td>Normal heading</td>
</tr>
</tbody>
</table>

BIB_ITEM p. 11, 18, 27, 33

This table is not completely reliable, particularly for “bound withs”. It is safer to use BIB_MFHD and MFHD_ITEM instead.

Don’t use this table in queries that include the MFHD_MASTER or BIB_MFHD tables, or you’ll probably get redundant rows and bad counts.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add_date</td>
<td>date</td>
<td>Add date</td>
</tr>
<tr>
<td>bib_id</td>
<td>number</td>
<td>Bibliographic identifier</td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
<td>Item identifier</td>
</tr>
<tr>
<td>operator_id</td>
<td>character 10</td>
<td>Operator identifier</td>
</tr>
</tbody>
</table>

BIB_LOCATION p. 33

This table provides an easy mapping from bibs to the locations in the MFHDs. It is fine for most purposes, but be aware that it isn’t reliable for “bound withs.”

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td>Bibliographic identifier</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
<td>Location identifier</td>
</tr>
</tbody>
</table>

BIB_MASTER p. 2, 10, 11, 14, 15, 18, 22, 27, 30, 33

The exists_in_dps and exists_in_dps_date fields are new with V6.5.4. DPS is Ex Libris’ Digital Preservation System.

Voyager’s marcexport utility uses the create_date and update_date in BIB_MASTER, not the action_date in BIB_HISTORY.

If a record has not been modified, the modify_date is null.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td>Bibliographic identifier</td>
</tr>
<tr>
<td>export_ok_location_id</td>
<td>number</td>
<td>Export OK location</td>
</tr>
<tr>
<td>export_ok_opid</td>
<td>character 10</td>
<td>Export OK operator</td>
</tr>
<tr>
<td>library_id</td>
<td>number</td>
<td>Library identifier</td>
</tr>
<tr>
<td>create_date</td>
<td>date</td>
<td>Create date</td>
</tr>
<tr>
<td>exists_in_dps</td>
<td>character 1</td>
<td>Exists in DPS</td>
</tr>
<tr>
<td>exists_in_dps_date</td>
<td>date</td>
<td>Exists in DPS date</td>
</tr>
<tr>
<td>export_date</td>
<td>date</td>
<td>Export date</td>
</tr>
<tr>
<td>export_ok</td>
<td>character 1</td>
<td>Export OK</td>
</tr>
<tr>
<td>export_ok_date</td>
<td>date</td>
<td>Export OK date</td>
</tr>
<tr>
<td>suppress_in_opac</td>
<td>character 1</td>
<td>Suppress in OPAC</td>
</tr>
<tr>
<td>update_date</td>
<td>date</td>
<td>Update date</td>
</tr>
</tbody>
</table>

BIB_MEDIUM

The medium field holds the first byte of the 007 from a bib record. Voyager uses this table to limit searches by “medium” in the staff clients or “additional format specification” in Web Voyage.
bib_id number
medium character 1

**BIB_MFHD** p. 2, 18, 22, 27, 30, 33
bib_id number
mfhd_id number

**BIB_SUBDIVISION** p. 10, 33
The starred field in this table is in UTF-8.
bib_id number
subdiv_id number
*display_subdiv character 330

**BIB_TEXT** p. 11, 14, 27, 33, 36
The starred fields in this table are in UTF-8.

If you need data from a bib record that are not available in BIB_TEXT, check to see if they are in BIB_INDEX (for fields in left-anchored indexes) or ELINK_INDEX (for URLs). All of these are more efficient than using the BLOB functions.

If you need data from a fixed field, some of them are found here. Some are in the MARC*VW tables. The 007/0 (Category of Materials) is in the BIB_MEDIUM table. Other fixed fields can be extracted from FIELD_008 in BIB_TEXT using the Mid function. Just remember that most fixed fields are specific to a record type so you’ll need to check bib_format, and that MARC calls the first byte “0” whereas Mid calls the first byte “1”. For example, to get Nature of Contents (008 bytes 24-27 for bibs): SELECT Mid([bib_format],1,1) AS RecType, Mid([field_008],25,4) AS 008_24to27 FROM BIB_TEXT WHERE (((Mid([bib_format],1,1))) In (‘a’,’h’,’t’));

If you’re thinking of using begin_pub_date in a criterion, consider using the indexed version of this field. It’s in the BIB_INDEX table, in the normal_heading field when index_code=008D.

If you’re thinking of using language in a criterion, consider using the indexed version of this field. It’s in the BIB_INDEX table, in the normal_heading field when index_code=008L. Note that the value “n/a” appears as “N/A” in that table.

If you’re thinking of using place_code in a criterion, consider using the indexed version of this field. It’s in the BIB_INDEX table, in the normal_heading field when index_code=008P.

Here’s how MARC tags map to fields in BIB_TEXT:

<table>
<thead>
<tr>
<th>MARC Tag</th>
<th>BIB_TEXT Field</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leader</td>
<td>record_status</td>
<td></td>
</tr>
<tr>
<td>Leader</td>
<td>bib_format</td>
<td></td>
</tr>
<tr>
<td>Leader</td>
<td>encoding_level</td>
<td></td>
</tr>
<tr>
<td>Leader</td>
<td>descrip_form</td>
<td></td>
</tr>
<tr>
<td>008</td>
<td>field_008</td>
<td></td>
</tr>
<tr>
<td>008 byte 6</td>
<td>date_type_status</td>
<td></td>
</tr>
<tr>
<td>008 bytes 7-10</td>
<td>begin_pub_date</td>
<td></td>
</tr>
<tr>
<td>008 bytes 11-14</td>
<td>end_pub_date</td>
<td></td>
</tr>
<tr>
<td>008 bytes 7-10 - bytes 11-14</td>
<td>pub_dates_combined</td>
<td></td>
</tr>
<tr>
<td>008 bytes 15-17</td>
<td>place_code</td>
<td></td>
</tr>
<tr>
<td>008 bytes 22-23</td>
<td>map_projection</td>
<td></td>
</tr>
<tr>
<td>008 bytes 35-37</td>
<td>language</td>
<td></td>
</tr>
<tr>
<td>010 abz8</td>
<td>lccn</td>
<td></td>
</tr>
<tr>
<td>020 a</td>
<td>isbn</td>
<td></td>
</tr>
<tr>
<td>022 a</td>
<td>issn</td>
<td></td>
</tr>
<tr>
<td>024 a</td>
<td>other_std_num</td>
<td></td>
</tr>
<tr>
<td>027 a</td>
<td>stdtech</td>
<td></td>
</tr>
<tr>
<td>028 all subfields</td>
<td>publisher_number</td>
<td></td>
</tr>
<tr>
<td>030 a</td>
<td>coden</td>
<td></td>
</tr>
<tr>
<td>035 a</td>
<td>network_number</td>
<td></td>
</tr>
<tr>
<td>037 a</td>
<td>stock_number</td>
<td></td>
</tr>
<tr>
<td>074 a</td>
<td>gponum</td>
<td></td>
</tr>
<tr>
<td>100 abcdkq</td>
<td>author</td>
<td></td>
</tr>
<tr>
<td>110 abcdgkn</td>
<td>author</td>
<td></td>
</tr>
<tr>
<td>111 acdegnk</td>
<td>author</td>
<td></td>
</tr>
<tr>
<td>245 abcfghknps</td>
<td>title</td>
<td></td>
</tr>
<tr>
<td>245 ab</td>
<td>title_brief</td>
<td></td>
</tr>
<tr>
<td>130 adfgklmnpors</td>
<td>uniform_title</td>
<td></td>
</tr>
<tr>
<td>240 adfgklmnpors</td>
<td>uniform_title</td>
<td></td>
</tr>
<tr>
<td>243 adfgklmnpors</td>
<td>uniform_title</td>
<td></td>
</tr>
<tr>
<td>250 all subfields</td>
<td>edition</td>
<td></td>
</tr>
<tr>
<td>440 anpv</td>
<td>series</td>
<td></td>
</tr>
<tr>
<td>490 av</td>
<td>series</td>
<td></td>
</tr>
<tr>
<td>260 abc</td>
<td>imprint</td>
<td></td>
</tr>
<tr>
<td>260 a</td>
<td>pub_place</td>
<td></td>
</tr>
<tr>
<td>260 b</td>
<td>publisher</td>
<td></td>
</tr>
<tr>
<td>260 c</td>
<td>publisher_date</td>
<td></td>
</tr>
<tr>
<td>255 abc</td>
<td>map_math_data</td>
<td></td>
</tr>
</tbody>
</table>

bib_id number
*author character 255
begin_pub_date character 4
bib_format character 2
*coden character 6
date_type_status character 1
descrip_form character 1
*edition character 100
coding_level character 1
date_format character 4
gponum character 20
*imprint character 200
isbn character 50
issn character 20
language character 3
lccn character 20
*map_math_data character 255
map_projection character 2
*network_number character 30
*other_std_num character 30
place_code character 3
dated character 9
*pub_place character 100
*publisher character 150
*pub_date character 25
*publisher_number character 40
record_status character 1
*series character 255
*stdtech character 30
*stock_number character 50
*title character 255
*title_brief character 150
*uniform_title character 255

**BIB_TEXT**

Data in this table are defined in the SysAdmin client at Search, Title List Column Names.

**BIB_TEXT_DISPLAYFIELD**

bib_text_field character 30
display_name character 40

**BIB_USAGE_LOG**

The starred fields in this table are in UTF-8.

Records are written in this table and the OPAC_SEARCH_LOG table when OPAC Bib Usage logging is turned on. The table is documented in the Voyager Technical Users Guide.

A client_type of G or W indicates WebVoyage.

bib_id number
location_id number
operator_id character 10
session_id character 16
client_ip character 15
client_type character 1
*stat_string character 15
use_date date
use_type character 1

**BIB_VW**

bib_id number
create_location_id number
mfhd_id number
mfhd_create_location_id number
mfhd_location_id number
call_no character 300
call_no_type character 1
create_date
create_operator character 10
mfhd_create_date
mfhd_create_operator character 10
mfhd_location character 25
mfhd_location_code character 10
normalized_call_no character 300
sort_title character 150
title character 150

**BINDERY_COPY** p. 1

bindery_copy_id number
component_id number
copy_id number

**BINDERY_COPY_DATA** p. 1

bindery_copy_id number
bindery_data_id number
bindery_data_type_id number
bindery_data character 1000

**BINDERY_COPY_DATA_TYPE** p. 1

bindery_data_type_id number
bindery_copy_data_type_desc character 25

**BINDERY_VOLUME** p. 1

bindery_copy_id number
bindery_volume_id number
item_id number
bind_on_date date
caption character 256
chron character 80
freetext character 256
item_enum character 80
other_volume_data character 200
volume_note character 200
year character 20

**BINDERY_VOLUME_ISSUES** p. 1

bindery_volume_id number
component_id number
copy_id number
issue_id number
sequence_number number

BOOKING_RESULT
This table is part of Voyager’s Media Scheduling module.
  booking_result_id number
  booking_result character 20

BROWSE_STATS
The starred fields in this table are in UTF-8.
This table is used by Voyager as a shortcut for left-anchored searches. It’s not very useful for queries.

With V8.2, sub_type is expanded to 12 characters.
  stat_id number
  *stat_sample character 50
  stat_type character 1
  sub_type character 1
  subsub_type number

CACHE_MAPS
This table is used by Voyager as a shortcut for left-anchored searches. It’s not very useful for queries.
  cache_id number
  arg character 60
  bits number
  code character 4
  date_updated date
  segsize number

CACHE_SEGS
This table is used by Voyager as a shortcut for left-anchored searches. It’s not very useful for queries.
  cache_id number
  record_segment long raw 0
  seqnum number

CALENDAR  p. 35
Data in this table are defined in the SysAdmin client at Circulation, Calendars.
  calendar_id number
  end_of_term_date date
  lead_days number

CALL_NO_HIERARCHY  p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Call Number Hierarchy.
  call_no_hierarchy_id number
  call_no_type character 1
  code character 8
The data in this table are defined by the Ex Libris and cannot be changed.

This table is used for call number processing during bulk imports. It is not a complete list of values in CALL_NO_TYPE in MFHD_MASTER.

**CALL_NO_TYPE** p. 27, 34

The print_group_id field can be linked to the group_id field in CALL_SLIP_GROUP_LOCATION and CALL_SLIP_PRINT_GROUP.

If status_opid is blank, the last action (most likely a cancel) was done via Web Voyage.

The rest of the information about this table is relevant only for sites using UB.

The patron_db_id field gives the affiliation of the patron. For patrons of your library, it may be either zero or –1 or null. For patrons of other libraries, use the VOYAGER_DATABASES table to translate.

The pickup_db_id can also be translated with VOYAGER_DATABASES. For requests that will be picked up at your library, it will be either zero or null.

When a call slip is archived, none of the dates in the record are changed. When a call slip is promoted to another library, the call_slip record is archived immediately. The STATUS is set to 9. Otherwise, CALL_SLIP records are archived by the nightly circjob8 after the archive interval set in SysAdmin has passed.

The DATE_PROCESSED field is null if STATUS is 1, 2, or 3. Otherwise, it’s the same as the STATUS_DATE field. It’s easier to use

STATUS_DATE so you don’t have to remember this.

If a call slip is neither filled nor unfilled before the expire period for this call slip queue, the nightly circjob8 changes its status to Expired, which makes it eligible for promotion by circjob32.

For UB requests that have been promoted to this library, date_requested is the date on which the call slip arrived here, not the date on which the patron placed the request. You can find out where the request has been previously and get a closer approximation of the date the patron made the request by looking at the REQUEST_HISTORY table.

For call slips that were promoted to this library by circjob32, item_id=0 until the call slip is filled.

**CALL_SLIP** p. 2, 12, 30

When a call slip is archived, the call_slip_id gets copied to archive_id. You can use archive_id to link to request_history.
mfhd_id number
patron_db_id number
patron_group_id number
patron_id number
pickup_db_id number
pickup_location_id number
print_group_id number
status_opid character 10
date_processed date
date_requested date
item_chron character 80
item_enum character 80
item_year character 20
no_fill_reason number
not_needed_after number
note character 100
reply_note character 100
status number
status_date date

CALL_SLIP_PRINT_GROUP  p. 2, 30
Data in this table are defined in the SysAdmin client at Call Slips, Queues.

There is a record in this table for each call slip queue.

The group_id field can be used to link to print_group_id in CALL_SLIP and CALL_SLIP_ARCHIVE.

The values for process_method are C=automatically charge to the patron, H=place in On Hold status, and T= place in In Transit On Hold status

default_item_type_id number
group_id number
location_id number
archive_interval character 1
archive_period number
cat_review character 1
circ_review character 1
default_group character 1
expire_interval character 1
expire_period number
group_code character 10
group_name character 25
patron_info character 1
process_method character 1

CALL_SLIP_REASSIGN... Tables
When a call slip is reassigned to a different queue, this table shows the queue that it used to be in and who reassigned it. The status of the call slip is 3=Reassigned until the slip is filled, unfilled, expired, or canceled. The reassignment record is archived when the call slip is archived.

CALL_SLIP_REASSIGNMENT
call_slip_id number
operator_id character 10
print_group_id number
reassign_date date

CALL_SLIP_REASSIGN_ARCHIVE
archive_id number
operator_id character 10
print_group_id number
reassign_date date
CALL_SLIP_STATS
If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.
call_slip_id number
patron_stat_id number

CALL_SLIP_STATUS_TYPE p. 2, 12
status_desc character 25
status_type number

CAT_CONTROL_BARCODE p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Bulk Import Rules, Barcode tab.
import_rule_id number
field character 3
indicator1 character 1
indicator2 character 1
sequence number
subfield character 1

CAT_CONTROL_CALL_NO p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Call Number Hierarchy, Call Number tab.
call_no_hierarchy_id number
cutter_subfield character 1
field character 3
indicator1 character 1
indicator2 character 1
main_subfield character 1
sequence number

CAT_CONTROL_ITEM_TYPE p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Bulk Import Rules, Item Type tab.
The fixed_start and fixed_end fields do not appear to be used for anything.
import_rule_id number
field character 3
fixed_end number
fixed_start number
indicator1 character 1
indicator2 character 1
sequence number
subfield character 1

CAT_OPERATOR p. 23, 34
Data in this table display in the SysAdmin client at Security, Operator Profiles, Current Profiles tab.
cat_profile_id number
operator_id character 10

CAT_POLICY_DUP p. 34
Data in this table display in the SysAdmin client at Cataloging, Policy Definitions.
cat_policy_id number
dup_profile_id number

CAT_POLICY_GROUP p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Policy Definitions.
The opac_display field does not appear to be used for anything.
cat_policy_id number
cat_policy_name character 40
nuc_code character 15
opac_display character 1

CAT_POLICY_HIERARCHY p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Policy Definitions, Default Policies tab.
call_no_hierarchy_id number
cat_policy_id number

CAT_POLICY_LOCS p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Policy Definitions, Locations tab.
The circ_location field does not appear to be used for anything.
cat_group_id number
location_id number
call_no_type character 1
cataloging_location character 1
circ_location character 1
default_item_type number
nuc_code character 15
routing_location character 1

CAT_PROFILE
Data in this table are defined in the SysAdmin client at Security, Cataloging Profiles, Profile Values and Profile Values Cont. tabs.
<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat_profile_id</td>
<td>number</td>
</tr>
<tr>
<td>auth_add</td>
<td>character 1</td>
</tr>
<tr>
<td>auth_delete</td>
<td>character 1</td>
</tr>
<tr>
<td>auth_export_ok</td>
<td>character 1</td>
</tr>
<tr>
<td>auth_update</td>
<td>character 1</td>
</tr>
<tr>
<td>bib_add</td>
<td>character 1</td>
</tr>
<tr>
<td>bib_delete</td>
<td>character 1</td>
</tr>
<tr>
<td>bib_export_ok</td>
<td>character 1</td>
</tr>
<tr>
<td>bib_update</td>
<td>character 1</td>
</tr>
<tr>
<td>bib_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>cat_profile_name</td>
<td>character 25</td>
</tr>
<tr>
<td>change_ownership</td>
<td>character 1</td>
</tr>
<tr>
<td>global_replace</td>
<td>character 1</td>
</tr>
<tr>
<td>hold_add</td>
<td>character 1</td>
</tr>
<tr>
<td>hold_delete</td>
<td>character 1</td>
</tr>
<tr>
<td>hold_export_ok</td>
<td>character 1</td>
</tr>
<tr>
<td>hold_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>item_add</td>
<td>character 1</td>
</tr>
<tr>
<td>item_delete</td>
<td>character 1</td>
</tr>
<tr>
<td>item_update</td>
<td>character 1</td>
</tr>
<tr>
<td>item_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>marcauth_add_update</td>
<td>character 1</td>
</tr>
<tr>
<td>marcauth_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>marcbib_add_update</td>
<td>character 1</td>
</tr>
<tr>
<td>marcbib_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>marchold_add_update</td>
<td>character 1</td>
</tr>
<tr>
<td>marchold_view_only</td>
<td>character 1</td>
</tr>
<tr>
<td>mfhd_export_ok</td>
<td>character 1</td>
</tr>
<tr>
<td>use_template</td>
<td>character 1</td>
</tr>
</tbody>
</table>

**CAT_SECURITY_LOCS** p. 23
Data in this table display in the SysAdmin client at Security, Cataloging Profiles, Locations tab.

**CHARACTER_SET** p. 3
char_set_id number
char_set_code character 1
char_set_name character 30

**CHRON** p. 25
chron_type_id number
chron_seq number
chron_value character 20

**CHRON_TYPE** p. 25, 40
chron_type_id number
chron_name character 40
chron_type_code character 2

**CIRCCCHARGES_VW**
This view has a number of quirks and it is not efficient. Consider using the tables directly instead.

If an item has been deleted since it circulated, its charges are not available in this view.

The `gov_item_type` fields are the item's current type, in other words, the temp item type, if there is one, otherwise the item type. Similarly, the `gov_location` fields are the item's temp location, if there is one, otherwise the perm location.

The charge_date_time field is the date and time the item was charged. The charge_date_only field is just the date of the charge. Access treats this as if the charge were done at 12:00 a.m.

**CIRCRENEW_VW**
This view has a number of quirks and it is not efficient. Consider using the tables directly instead.

If an item has been deleted since it circulated, its renewals are not available in this view.

The `gov_item_type` fields are the item's current type, in other words, the temp item type, if there is one, otherwise the item type. Similarly, the `gov_location` fields are the item's temp location, if there is one, otherwise the perm location.
is one, otherwise the item type. Similarly, the
gov_location fields are the item’s temp location, if
there is one, otherwise the perm location.

The charge_date_time field is the date and time
the item was charged. The charge_date_only
field is just the date of the charge. Access treats
this as if the charge were done at 12:00 a.m. The
two renew_date fields function similarly.

bib_id number
charge_oper_id character 10
item_id number
mfhd_id number
patron_group_id number
renew_oper_id character 10
charge_date_only date
charge_date_time date
charge_location number
charge_location_code character 10
charge_location_name character 25
gov_item_type character 25
gov_item_type_code character 10
gov_location character 25
gov_location_code character 10
location_name character 25
patron_group_code character 10
patron_group_name character 25
perm_item_type character 25
perm_item_type_code character 10
perm_location character 25
perm_location_code character 10
renew_date_only date
renew_date_time date
renew_location_code character 10
renewal_count number

CIRC_ALERTS p. 4
alert_id number
alert_name character 30
alert_text character 100
alert_type number

CIRC_ALERT_CONDITIONS p. 4
Data in this table are defined in the SysAdmin
client at Circulation, Policy Definitions, Alerts tab.

If you get the message, “Type mismatch in
expression”, when you use this table, see
Appendix A for a solution.

alert_id number
loc_id number

CIRC_ALERT_TYPES p. 4
Data in this table are defined in the SysAdmin
client at Circulation, Policy Definitions, Alerts tab.

alert_type number
alert_type_desc character 100

CIRC_BLOCKS p. 24
Data in this table are defined in the SysAdmin
client at Security, Circulation Profiles, Patron
Blocks and Item Blocks tabs.

block_id number
block_display_name character 100
block_name character 30
block_type character 6

CIRC_BLOCK_OVERRIDE p. 24
Data in this table are defined in the SysAdmin
client at Security, Circulation Profiles, Patron
Blocks and Item Blocks tabs.

block_id number
circ_profile_id number

CIRC_CLUSTER p. 2, 6, 8, 11, 15, 20, 21, 28, 29, 30
Data in this table are defined in the SysAdmin
client at Circulation, Cluster.

With V6.5, the ub_block_local_patrons is
removed.

circ_cluster_id number
circ_cluster_code character 10
circ_cluster_name character 100
default_pickup_location number

CIRC_GROUP CALENDAR p. 35
Data in this table display in the SysAdmin client
at Circulation, Policy Definitions, Calendar tab.

calendar_id number
circ_group_id number

CIRC_OPERATOR p. 24, 35
Data in this table display in the SysAdmin client at Security, Operator Profiles, Current Profiles tab.

- **circ_profile_id** number
- **operator_id** character 10

**CIRC_POLICY_GROUP** p. 6, 7, 35
Data in this table are defined in the SysAdmin client at Circulation, Policy Definitions, Policies tab.

- **circ_cluster_id** number
- **circ_group_id** number
- **circ_group_name** character 40
- **closed_days_for_fines** character 1
- **closed_days_for_loans** character 1
- **early_pickup_window** number
- **extend_recall_due_date** character 1
- **fixed_due_time** character 1
- **lost_process_fee** character 1
- **max_fine_fee_for_lost** character 1
- **process_fee** number
- **renew_if_hold** character 1
- **renew_if_overdue** character 1
- **renew_if_recall** character 1
- **unclaimed_interval** number

**CIRC_POLICY_LOCS** p. 4, 6, 7
Data in this table are defined in the SysAdmin client at Circulation, Policy Definitions, Locations tab.

- **circ_group_id** number
- **location_id** number
- **automated_storage** character 1
- **circ_location** character 1
- **collect_fines** character 1
- **courtesy_discharge** character 1
- **default_item_type** number
- **default_location** number
- **hold_life** number
- **hold_shelf_life** number
- **hold_shelf_life_interval** character 1
- **opac_circ_desk** character 1
- **patron_avail_items_alert** character 1
- **patron_fine_fee_alert** character 1
- **pickup_location** character 1
- **print_date_dues** character 1
- **print_discharge_receipts** character 1
- **print_fine_receipts** character 1
- **print_hold_slips** character 1
- **print_location** number
- **print_routing_slips** character 1
- **recall_life** number
- **shelving_interval** character 1
- **shelving_period** number
- **suppress_fly_items** character 1
- **transit_period** number

**CIRC_POLICY_MATRIX** p. 6, 7, 35
Data in this table are defined in the SysAdmin client at Circulation, Policy Definitions.

The value of loan_interval may be $M$=minutes, $H$=hours, $D$=days, $T$=term, or $I$=indefinite. For Term loans and Indefinite loans, the value of loan_period does not matter and it may be zero. For all other intervals, if loan_period=0, then the item does not circulate.

If the value of item_type_id and patron_group_id is zero, this matrix entry applies to all item types and patron groups, in this circ policy group.

Hold_shelf_life is the number of days a local item can sit on the hold shelf waiting for the patron to pick it up. Don’t confuse this with hold_life in CIRC_POLICY_LOCS.

Lost_notice_interval is the number of days overdue after which the item is given the “Lost-System Applied” status.
loan_period number
lost_notice_interval number
max_fine number
max_recall_fine number
other_notice_count number
other_notice_interval number
place_call_slip character 1
place_hold character 1
place_recall character 1
place_ub_request character 1
recall_fine_interval character 1
recall_fine_rate number
recall_grace_period number
recall_min_loan number
recall_notice_count number
recall_notice_interval number
recall_return_period number
renew_from_due_date character 1
renewal_count number
renewal_interval character 1
renewal_period number

CIRC_PROFILE  p. 24
Data in this table are defined in the SysAdmin client at Security, Circulation Profiles and display at Operator, Current Profiles.

circ_profile_id number
add_fines character 1
change_discharge_date character 1
change_due_date character 1
charge_renew character 1
circ_profile_name character 25
discharge character 1
distribution_item_create character 1
distribution_item_delete character 1
distribution_item_distribute char
distribution_item_order character 1
distribution_item_receive character 1
distribution_item_update character 1
distribution_item_view character 1
edit_stub_patron character 1
forgive_fines character 1
hold_ignore_ownership character 1
item_add_update character 1
item_delete character 1
item_status character 1
manually_map_patron character 1
mfhd_update character 1
patron_add_update character 1
patron_counters character 1
patron_delete character 1
patron_mask_ssn character 1
patron_proxy_maintain character 1
patron_view_only character 1
pay_fines character 1
pg_restrict_circ character 1
pg_restrict_maint character 1
pg_restrict_view character 1
recahold_add_update character 1
recahold_resquence character 1
reserve_add_update character 1
update_pin character 1
view_patron_circ_history character 1

CIRC_SECURITY_LOCS  p. 24
Data in this table are defined in the SysAdmin client at Security, Circulation Profiles, Locations tab.
circ_profile_id number
location_id number

CIRC_SECURITY_PG  p. 24
Data in this table are defined in the SysAdmin client at Security, Circulation Profiles, Patron Groups tab.
circ_profile_id number
patron_group_id number

CIRC_TRANS... Tables
Circulation transactions are recorded in CIRC_TRANSACTIONS until the item is discharged, after which they are moved to CIRC_TRANS_ARCHIVE. Consequently, the discharge... fields in CIRC_TRANSACTIONS are always blank.

When a transaction is archived, the value of circ_transaction_id is changed. In both tables, circ_transaction_id is assigned sequentially as a record is added.

The db_id field gives the affiliation of the patron. It can be translated using the VOYAGER_DATABASES table. For patrons affiliated with your library, db_id may be either zero or null.

For items charged at another library, charge_location=0 and charge_oper_id='SYS-UB'.
For items discharged at another library, discharge_location=0 and discharge_oper_id='SYS-UB'.

15
For many circ statistics, you will want to combine data from these two tables. Voyager provides an Access query called “Circulation Transactions (Charges)” which does this for you. From an Add Tables window in Access, click the Queries tab and you’ll find it.

The charge_type and discharge_type fields have 2 values, N for Normal and O for Override.

If patron_id_proxy is neither zero nor null nor equal to the patron_id, then the charge was done by a proxy patron. The patron_id is the sponsor’s patron_id and patron_id_proxy is the proxy’s patron_id.

When circulation staff do an on-the-fly charge, that is, when they charge an item that is not in the database, Voyager creates an item, a MFHD and, if needed, a bib record. When the item is discharged, these records are retained unless circ staff do something to delete them, so they are available for circ statistics. However, if your circ staff routinely delete these records and you want on-the-fly circ to be included in your circ statistics, be sure to change the link properties on the link from the circ transaction table(s) to the item table.

**CIRC_TRANSACTIONS**  p. 6, 12

The current_due_date is the due date that you normally want to use. It is set when the item is initially charged and changed when the item is renewed or recalled. The charge_due_date field is the due date at the time the item was initially charged; it never changes. If the item was recalled, the new due date will be in recall_due_date. If the item was renewed, the dates of each renewal will be in the RENEW* tables.

charge_oper_id character 10
circ_policy_matrix_id number
circ_transaction_id number
db_id number
discharge_oper_id character 10
item_id number
patron_group_id number
patron_id number
patron_id_proxy number
charge_date date
charge_due_date date
charge_location number
charge_type character 1
courtesy_notice_date date
current_due_date date
discharge_date date
discharge_location number
discharge_type character 1
over_recall_notice_count number
over_recall_notice_date date
overdue_notice_count number
overdue_notice_date date
recall_date date
recall_due_date date
recall_notice_count number
recall_notice_date date
renewal_count number

**CIRC_TRANSACTION_STATS**  p. 5

This table may be linked to CIRC_TRANS_ARCHIVE, but not to CIRC_TRANSACTIONS. Refer to the notes on the PATRON_STATS table.

circ_transaction_id number
patron_stat_id number

**CIRC_TRANS_ARCHIVE**  p. 5, 12

Due_date is the due date at the time the item was initially charged out. If the item was renewed, the new due date is not stored in this table.

The patron_id field will always be 0 unless you have checked Retain Patron ID for Circ History in SysAdmin.

charge_oper_id character 10
circ_policy_matrix_id number
circ_transaction_id number
db_id number
discharge_oper_id character 10
item_id number
patron_group_id number
patron_id number
patron_id_proxy number
charge_date date
date
discharge_location number
discharge_type character 1
courtesy_notice_date date
discharge_date date
discharge_location number
discharge_type character 1
due_date date
over_recall_notice_count  number
over_recall_notice_date  date
overdue_notice_count  number
overdue_notice_date  date
recall_date  date
recall_due_date  date
recall_notice_count  number
recall_notice_date  date
renewal_count  number

CIRC_TRANS_EXCEPTION
Unusual circulation activity is recorded here during the day. The table is cleared out nightly when the circ transactions exception report is run.
circ_trans_except_id  number
item_id  number
patron_id  number
trans_except_oper_id  character 10
item_location  number
trans_except_date  date
trans_except_location  number
trans_except_type  number

CIRC_TRANS_EXCEPT_TYPE
exception_desc  character 50
exception_type  number

CLAIM_TYPES  p. 17, 25
Data in this table are defined in the SysAdmin client at Acquisitions, Claim Types.
claim_type  number
claim_type_desc  character 70
edi_code  character 11

CLASS_SECTION  p. 8
This table is part of Reserves functionality.
circ_cluster_id  number
section_id  number
normal_section_number  character 10
number_of_students  number
section_number  character 10

COMPLEX_COMP_PATTERN  p. 40
If a record has not been modified, the modify_date is null.
cp_id  number
create_location_id  number
update_location_id  number
create_date  date
create_opid  character 10
pattern_name  character 40
pattern_name_norm  character 40
update_date  date
update_opid  character 10

COMPONENT  p. 1, 25, 26
The values of predict are: Y=yes, the component uses a predictive pattern, N=the component uses a non-predictive pattern, S=the pattern has been closed, M=the pattern has yet to be set, C=the component uses a complex pattern.
component_id  number
item_type_id  number
next_issue_id  number
subscription_id  number
category  number
claim_interval  number
component_name  character 100
component_name_norm  character 100
create_items  character 1
note  character 256
predict  character 1
unit_title  number

COMPONENT_ALTCHRONDAY  p. 25
component_id  number
chron_day  number
type_of_day  character 3

COMPONENT_CHRONDAY  p. 25
component_id  number
chron_day  number
type_of_day  character 3

COMPONENT_ISSUES_ROUTED  p. 26
component_id  number
issue_id  number
routing_list_id  number

COMPONENT_ISSUE_DAY  p. 25
component_id  number
expected_day  number
type_of_day  character 3
### COMPONENT_PATTERN p. 25
- component_id number
- end_issue_id number
- pattern_id number
- start_issue_id number
- alt_lv1_inc_at number
- alt_lv2_inc_at number
- end_date date
- frequency_code character 1
- lv1_inc_at number
- lv2_inc_at number
- lv3_inc_at number
- lv4_inc_at number
- lv5_inc_at number
- lv6_inc_at number
- regularity character 12
- regularity_marc character 50

### COMPONENT_ROUTING p. 26
- component_id number
- routing_list_id number

### CONVERSION_RATE_AUDIT
- audit_id number
- currency_id number
- rate_create_operator_id character 10
- conversion_rate number
- rate_create_date_time date

### COURSE p. 8
This table is part of reserves.
- circ_cluster_id number
- course_id number
- begin_date date
- course_name character 40
- course_number character 10
- end_date date
- normal_course_name character 40
- normal_course_number character 10

The CP... Tables
These tables are part of the implementation of complex serial patterns in the acquisitions module.

### CP_CELL p. 40
- cp_issue_id number
- cp_level_id number
- level_increment character 80

### CP_DOMAIN_TYPE p. 40
If domain=c, then enum_chron_type_id can be linked to CHRON_TYPE. If domain=e, then enum_chron_type_id can be linked to ENUMERATION_TYPE.
- cp_domain_type_id number
- enum_chron_type_id number
- domain character 1

### CP_ISSUE p. 40
- cp_id number
- cp_issue_id number
- cp_issue number
- expected_date_inc number
- time_unit_code character 1

### CP_LEVEL p. 40
- cp_domain_type_id number
- cp_id number
- cp_level_id number
- caption character 50
- cp_level number
- is_constant character 1
- print_order number
- reg_or_alt character 1

### CURRENCY_CONVERSION
- create_operator_id character 10
- currency_id number
- rate_create_operator_id character 10
- conversion_rate number
- country_name character 45
- create_date date
- currency_code character 3
- currency_name character 35
- decimal_delimiter character 1
- decimals number
- normal_country_name character 45
- normal_currency_code character 3
- normal_currency_name character 35
- rate_create_date_time date

### DATABASE_ADDRESS
Data in this table are defined in the SysAdmin client at Search, Database Definitions, Definitions tab.
- db_id number
- application_type character 20
- db_addr character 100
- db_port number

### DATABASE_LICENSE
Ex Libris says that this table keeps track of active connections to Voyager. It is not used to track simultaneous user licenses. It could be used for this, but Oracle does it instead. When a connection times out, its record is deleted.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>license_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>session_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>db_code</td>
<td>character 8</td>
<td></td>
</tr>
<tr>
<td>init_date</td>
<td>date</td>
<td></td>
</tr>
<tr>
<td>module</td>
<td>character 20</td>
<td></td>
</tr>
</tbody>
</table>

**DEPARTMENT** p. 8

This table is part of reserves.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>circ_cluster_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>department_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>department_code</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>department_name</td>
<td>character 40</td>
<td></td>
</tr>
<tr>
<td>normal_dept_code</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>normal_dept_name</td>
<td>character 40</td>
<td></td>
</tr>
</tbody>
</table>

**DEWEYCLASS_VW**

There’s an entry in this table for every call number encoded as Dewey (MFHD 852 first indicator = 1) even if that encoding is wrong. CLASS is the first 3 characters of 852$h. LONGCLASS is the entire 852$h.

If you want to sort a report (not a query, a report) by longclass, you will have to use the Left function to truncate it to less than 255 characters.

To add a description of each Dewey class to your queries, link the class field in DEWEYCLASS_VW to the DeweyNum field in the DeweyDetailed table and show the Description field.

A set of techniques for producing statistics by more precise call number ranges is given in “Reports with Call Number Ranges: How to Request Then and How to Write Them” at http://www.carli.illinois.edu/products-services/i-share/reports/secure/callnumrange

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mfhd_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>class</td>
<td>character 3</td>
<td></td>
</tr>
<tr>
<td>longclass</td>
<td>character 300</td>
<td></td>
</tr>
</tbody>
</table>

**DISTRIBUTION_ITEM** p. 9

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>create_location_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>create_opid</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>modify_location_id</td>
<td>number</td>
<td></td>
</tr>
</tbody>
</table>

**DISTRIBUTION_ORDER** p. 9

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>distribution_order_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>order_location_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>order_opid</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>vendor_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>expected_date</td>
<td>date</td>
<td></td>
</tr>
<tr>
<td>not_yet_received</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>order_complete</td>
<td>character 1</td>
<td></td>
</tr>
<tr>
<td>order_date</td>
<td>date</td>
<td></td>
</tr>
<tr>
<td>order_quantity</td>
<td>number</td>
<td></td>
</tr>
</tbody>
</table>

**DISTRIBUTION_RECEIPT** p. 9

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>distribution_order_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>distribution_receipt_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>receipt_location_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>receipt_opid</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>receipt_date</td>
<td>date</td>
<td></td>
</tr>
<tr>
<td>receipt_quantity</td>
<td>number</td>
<td></td>
</tr>
</tbody>
</table>

**DISTRIBUTION_TRANSACTION** p. 9

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>distribution_location_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>distribution_opid</td>
<td>character 10</td>
<td></td>
</tr>
<tr>
<td>distribution_transaction_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>patron_group_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>patron_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>distribution_date</td>
<td>date</td>
<td></td>
</tr>
</tbody>
</table>

**DUPE_PROFILE_MERGE** p. 34

The starred field in this table is in UTF-8.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dup_profile_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>marc_field</td>
<td>character 3</td>
<td></td>
</tr>
<tr>
<td>marc_ind1</td>
<td>character 1</td>
<td></td>
</tr>
<tr>
<td>marc_ind2</td>
<td>character 1</td>
<td></td>
</tr>
<tr>
<td>*nuc5</td>
<td>character 15</td>
<td></td>
</tr>
</tbody>
</table>

**DUP_DETECTION_PROFILE** p. 34

Data in this table are defined in the SysAdmin client at Cataloging, Authority Duplicate Detection Profiles, Profile tab, and at Cataloging,
Bibliographic Duplicate Detection Profiles, Profile tab, and at Search, Indexes, Bibliographic Record Linking, and at Search, Hook to Holdings.

The discard_unmatched field indicates that an incoming bib record should be discarded if it does not match a record in the database.

dup_profile_id number
discard_unmatched character 1
cancellation character 1
displayfield1 character 30
displayfield2 character 30
displayfield3 character 30
dup_handling character 1
dup_profile_code character 8
dup_profile_name character 25
dup_replace number
dup_warn number
record_type character 1
seqnum number

DUP_PROFILE_FIELDS p. 34
Data in this table are defined at Cataloging, Authority Duplicate Detection Profiles, Field Definitions tab, and at Cataloging, Bibliographic Duplicate Detection Profiles, Field Definitions tab, and at Search, Indexes, Bibliographic Record Linking.

dup_profile_id number
fieldoverride character 3
indicator_1 character 1
indicator_2 character 1
searchcode character 4
seqnum number
subfieldoverride character 10
weight number

DUP_PROFILE_QUALITY p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Authority Duplicate Detection Profiles, Quality Hierarchy tab, and at Cataloging, Bibliographic Duplicate Detection Profiles, Quality Hierarchy tab, and at Cataloging, Bibliographic Duplicate Detection Profiles, Merge Data tab.

The starred fields in this table are in UTF-8.
dup_profile_id number
encoding_level character 1
*modifying_agency character 15
*nuc_code character 15
record_type character 2
seqnum number

EDI_CODE_REF
code character 3
descr character 70
usage number

EDI_CODE_USAGES
data_element character 4
descr character 70
usage number

EDI_CONNECTION_PROFILE
If a record has not been modified, the modify_date is the same as the create_date.
create_opid character 10
location_id number
profile_id number
update_opid character 10
vendor_id number
create_date date
library_envelope_address character 55
library_inside_address character 2
update_date date
use_iv character 1
use_mc character 1
use_mr character 1
use_po character 1
use_sc character 1
use_sr character 1
use_vendor_account character 1
use_xm character 1
vendor_envelope_address character 55
vendor_inside_address character 25

EDI_CURSOR
cursor_id number
file_id number
msg_id number
file_name character 30
file_position number
msg_delimiters character 6

EDI_EVENT_TYPES
event_desc character 25
event_type number

EDI_FILE
If a record has not been modified, the modify_date is the same as the create_date.
create_op_id character 10
file_id number
update_op_id character 10
create_date
type file_name character 30
file_size number
file_status number
file_type character 1
file_update_date
group_count number
message_count number
trans_count number
update_date
date

EDI_HISTORY
If a record has not been modified, the modify_date is the same as the create_date.
create_op_id character 10
event_id number
file_id number
msg_id number
update_op_id character 10
create_date
event_type number
update_date
date
date

EDI_MESSAGE
If a record has not been modified, the modify_date is the same as the create_date.
create_op_id character 10
db_ref_id number
file_id number
location_id number
msg_id number
update_op_id character 10
vendor_id number
create_date
create_loc number
data_present character 1
doc_msg_code character 3
file_end_pos number
file_start_pos number
group_index number
line_item_count number
load_or_append_date
date
matching_profile number
msg_date
date
msg_delimiters character 6
msg_direction character 1
msg_number character 35
msg_status number
msg_type character 6
msg_type_code number
msg_version_code number
receiver_code character 55
sender_code character 55
total_amount number
trans_index number
update_date
date
update_loc number
date

EDI_MISSING_LINE_ITEM
exception_id number
line_id number
print_std_num character 2
problem_code number
std_number character 40
title character 100
vendor_ref_num character 35
vendor_ref_qual character 3
vendor_title_num character 40
date
date
date

EDI_NOTE
event_id number
note_code number
position number
date

EDI_SECTION
msg_id number
section_id number
section_ordinal number
section_type character 3
seg_count number
segments long raw 0

EITEM  p. 8, 36
This table is part of Reserves functionality.

The starred field in this table is in UTF-8.

An electronic item has a MFHD and a bib, which you can link to just as you would link physical items.

If a record has not been modified, the modify_date is null.
create_location_id number
create_opid character 10
eitem_id number
mfhd_id number
update_location_id number
date

This table is part of Reserves functionality.

eitem_id number
eitem_note_type_id number
note character 2000

This table is part of Reserves functionality.

note_desc character 25
note_type number

ELINK_INDEX is a very handy place to find URLs from various types of records.

Record_type is supposed to be interpreted by the ELINK_RECORD_TYPE table, but there are some errors. Actual values for record_type are A for Authority, B for Bibliographic, E for Electronic item, and M for MFHD. (ELINK_RECORD_TYPE has a row, I for Item, but you can’t have a URL in an item.)

The record_id is either an auth_id, a bib_id, an eitem_id, or a mfhd_id, depending on the value of record_type.

As a general rule, the link field is 856$u and the link_text field is subfields $z and $3. See Appendix B for more details. It also holds bib subfields 505$u, 506$u, 514$u, 520$u, 530$u, 540$u, 545$u, 552$u, 583$u, and 856$g, as well as MFHD subfields 563$u and 583$u.

eq d
record_id number
update_opid character 10

equip_id number
equip_type_id number
group_equip_id number
media_room_id number
temp_room_id number
update_location_id number
update_opid character 10
create_date date
date_purchased date
dealer character 100
dealer_normalized character 100
equip_format character 25
Data in this table are defined in the SysAdmin client at Circulation, Calendars.

calendar_id number
time_exception_closehour number
time_exception_date date
time_exception_hourly_effect number
time_exception_loan_due number
time_exception_open character 1
time_exception_openhour number

EXCEPTION_TYPES
exception_type number
exception_type_desc character 20

FIELDWEIGHTS
Data in this table are defined in the SysAdmin client at Search, Indexes, Field Weighting.

fieldcode character 4
fieldweight number

FINE_FEE
When a fine is paid, the fine_fee_balance goes to zero, but the record is not deleted from this table.
The db_id field is not maintained by Voyager, so don’t use it. If you need to know the affiliation of the patron who owes you a fine, use the db_id field in the PATRON table.

The fine_fee_location may be zero if the item was returned at a library other than its home library. In this case, the operator_id may be null or “SYS-UB”.

The create_date field is not filled in for manually applied fines and fees. Some libraries manually apply most fines and fees, even those for overdue and lost items.

db_id number
fine_fee_id number
item_id number
operator_id character 10
patron_id number
create_date date
due_date date
fine_fee_amount number
fine_fee_balance number
fine_fee_location number
fine_fee_note character 1000
fine_fee_notice_date date
fine_fee_type number
orig_charge_date date

FINE_FEE_TRANSACTIONS p. 19
fine_fee_id number
fine_fee_trans_id number
operator_id character 10
trans_amount number
trans_date date
trans_location number
trans_method number
trans_note character 1000
trans_type number

FINE_FEE_TRANS_METHOD p. 19
method_desc character 25
method_type number

FINE_FEE_TRANS_TYPE p. 19
Data in this table are defined in the SysAdmin client at System, Fines/Fees, Payment tab.

transaction_desc character 25
transaction_type number

FINE_FEE_TYPE p. 19
Data in this table are defined in the SysAdmin client at System, Fines/Fees, Payment tab.

fine_fee_code character 10
fine_fee_desc character 25
fine_fee_type number

FISCAL_PERIOD p. 13
Data in this table are defined in the SysAdmin client at Acquisitions, Fiscal Periods.

The fiscal_period_id field in FISCAL_PERIOD can be used to link to fiscal_year_id field in LEDGER. This isn’t obvious from the names.

end_date date
fiscal_period_id number
fiscal_period_name character 25
start_date date

FREQUENCY p. 25
freq_calc_type character 1
freq_increment number
frequency_code character 1
frequency_desc character 25

FUND... Tables
Remember that a fund_id does not uniquely identify a fund. It’s the combination of fund_id and ledger_id that uniquely identifies a fund. Consequently, you need to link by both of these fields when you are linking among the FUND..., PO_FUNDS and LINE_ITEM_FUNDS tables.

FUND p. 13, 22
There’s no table that translates the values in the category field. The values are 0=Summary, 1=Allocated, 2=Reporting.

To get to the parent fund, add a second FUND table to your query, linking ledger_id to ledger_id and parent_fund_id to fund_id.

If a record has not been modified, the modify_date is the same as the create_date.

create_opid character 10
fund_id number
institution_fund_id character 50
ledger_id number
update_opid character 10
<table>
<thead>
<tr>
<th>Field</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocation_decrease</td>
<td>number</td>
<td>Decrease in allocation</td>
</tr>
<tr>
<td>allocation_increase</td>
<td>number</td>
<td>Increase in allocation</td>
</tr>
<tr>
<td>begin_date</td>
<td>date</td>
<td>Start date of the period</td>
</tr>
<tr>
<td>category</td>
<td>number</td>
<td>Category number</td>
</tr>
<tr>
<td>commit_freeze</td>
<td>date</td>
<td>Commit freeze date</td>
</tr>
<tr>
<td>commit_pending</td>
<td>number</td>
<td>Commit pending number</td>
</tr>
<tr>
<td>commitments</td>
<td>number</td>
<td>Total commitments</td>
</tr>
<tr>
<td>create_date</td>
<td>date</td>
<td>Creation date</td>
</tr>
<tr>
<td>end_date</td>
<td>date</td>
<td>End date of the period</td>
</tr>
<tr>
<td>expend_freeze</td>
<td>date</td>
<td>Expend freeze date</td>
</tr>
<tr>
<td>expend_only</td>
<td>character 1</td>
<td>Expend only flag</td>
</tr>
<tr>
<td>expend_pending</td>
<td>number</td>
<td>Expend pending number</td>
</tr>
<tr>
<td>expenditures</td>
<td>number</td>
<td>Total expenditures</td>
</tr>
<tr>
<td>fund_code</td>
<td>character 10</td>
<td>Fund code</td>
</tr>
<tr>
<td>fund_name</td>
<td>character 25</td>
<td>Fund name</td>
</tr>
<tr>
<td>fund_type</td>
<td>number</td>
<td>Fund type number</td>
</tr>
<tr>
<td>normal_fund_code</td>
<td>character 10</td>
<td>Normal fund code</td>
</tr>
<tr>
<td>normal_fund_name</td>
<td>character 25</td>
<td>Normal fund name</td>
</tr>
<tr>
<td>original_allocation</td>
<td>number</td>
<td>Original allocation</td>
</tr>
<tr>
<td>overcommit</td>
<td>character 1</td>
<td>Overcommit flag</td>
</tr>
<tr>
<td>overcommit_percent</td>
<td>number</td>
<td>Overcommit percentage</td>
</tr>
<tr>
<td>overcommit_warn</td>
<td>number</td>
<td>Overcommit warn number</td>
</tr>
<tr>
<td>overexpend</td>
<td>character 1</td>
<td>Overexpend flag</td>
</tr>
<tr>
<td>overexpend_percent</td>
<td>number</td>
<td>Overexpend percentage</td>
</tr>
<tr>
<td>overexpend_warn</td>
<td>number</td>
<td>Overexpend warn number</td>
</tr>
<tr>
<td>parent_fund</td>
<td>character 25</td>
<td>Parent fund name</td>
</tr>
<tr>
<td>policy_name</td>
<td>character 40</td>
<td>Policy name</td>
</tr>
<tr>
<td>ledger_name</td>
<td>character 40</td>
<td>Ledger name</td>
</tr>
<tr>
<td>normal_fund_name</td>
<td>character 25</td>
<td>Normal fund name</td>
</tr>
<tr>
<td>normal_ledger_name</td>
<td>character 40</td>
<td>Normal ledger name</td>
</tr>
<tr>
<td>parent_fund_id</td>
<td>number</td>
<td>Parent fund ID</td>
</tr>
<tr>
<td>policy_name</td>
<td>character 40</td>
<td>Policy name</td>
</tr>
<tr>
<td>fund_category</td>
<td>character 9</td>
<td>Fund category</td>
</tr>
<tr>
<td>fund_name</td>
<td>character 25</td>
<td>Fund name</td>
</tr>
<tr>
<td>fund_type</td>
<td>character 25</td>
<td>Fund type</td>
</tr>
<tr>
<td>fundline</td>
<td>character 255</td>
<td>Fund line</td>
</tr>
</tbody>
</table>

**FUNDLEDGER_VW**

<table>
<thead>
<tr>
<th>Field</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fiscal_period_id</td>
<td>number</td>
<td>Fiscal period ID</td>
</tr>
<tr>
<td>fund_id</td>
<td>number</td>
<td>Fund ID</td>
</tr>
<tr>
<td>institution_fund_id</td>
<td>character 50</td>
<td>Institution fund ID</td>
</tr>
<tr>
<td>ledger_id</td>
<td>number</td>
<td>Ledger ID</td>
</tr>
<tr>
<td>parent_fund_id</td>
<td>number</td>
<td>Parent fund ID</td>
</tr>
<tr>
<td>begin_date</td>
<td>date</td>
<td>Beginning date of the period</td>
</tr>
<tr>
<td>cash_balance</td>
<td>number</td>
<td>Cash balance</td>
</tr>
<tr>
<td>commit_pending</td>
<td>number</td>
<td>Commit pending number</td>
</tr>
<tr>
<td>commitments</td>
<td>number</td>
<td>Total commitments</td>
</tr>
<tr>
<td>current_allocation</td>
<td>number</td>
<td>Current allocation</td>
</tr>
<tr>
<td>end_date</td>
<td>date</td>
<td>Ending date of the period</td>
</tr>
<tr>
<td>expend_pending</td>
<td>number</td>
<td>Expend pending number</td>
</tr>
<tr>
<td>expenditures</td>
<td>number</td>
<td>Total expenditures</td>
</tr>
<tr>
<td>fiscal_period_end</td>
<td>date</td>
<td>Fiscal period end</td>
</tr>
<tr>
<td>fiscal_period_name</td>
<td>character 25</td>
<td>Fiscal period name</td>
</tr>
<tr>
<td>fiscal_period_start</td>
<td>date</td>
<td>Fiscal period start</td>
</tr>
<tr>
<td>free_balance</td>
<td>number</td>
<td>Free balance</td>
</tr>
<tr>
<td>fund_category</td>
<td>character 9</td>
<td>Fund category</td>
</tr>
<tr>
<td>fund_name</td>
<td>character 25</td>
<td>Fund name</td>
</tr>
<tr>
<td>fund_type</td>
<td>character 25</td>
<td>Fund type</td>
</tr>
<tr>
<td>fundline</td>
<td>character 255</td>
<td>Fund line</td>
</tr>
</tbody>
</table>

**FUNDNOTE**

- The ledger_id field is always set to 0.
- Consequently, a fund note persists from fiscal year to fiscal year.
  - fund_id number
  - ledger_id number
  - fund_note character 1900

**FUND_PAYMENT**

- fund_id number
- ledger_id number
- payment_id number
- amount number
- percentage number
- split_fund_seq number

**FUND_TRANSACTION**

- If trans_type=4 (commitment) then reference_no is a PO number. If trans_type=5 (expenditure) then reference_no is an invoice number.

- If trans_type=6 then fund_id is the fund from which the transfer came. If trans_type=7 then fund_id is the fund to which the money was transferred. Be sure you link on ledger_id too.
  - audit_id number
  - fund_id number
  - ledger_id number
  - operator_id character 10 | Operator ID |
  - amount number
  - note character 1900
  - reference_no character 25 | Reference number |
  - statistical_fund number
  - trans_date date
  - trans_type number

**FUND_TYPE**

- Data in this table are defined in the SysAdmin client at Acquisitions, Fund Types.
  - fund_type_id number
  - commit_warning number
  - expend_warning number
  - fund_type_name character 25 | Fund type name |
overcommit_limit number
overexpend_limit number
undercommit number
underexpend number

**GEO_COORD_TYPE**
The starred field in this table is in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.
- *coord_name character 25
- coord_type number

**GEO_FORMAT_TYPE**
The starred field in this table is in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.
- coord_type number
- *format_name character 30
- format_type number

**GEO_SEARCH**
The starred field in this table is in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.
- search_name character 25
- search_type number

**GEO_UNITS**
The starred field in this table is in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.
- unit_name character 25
- unit_type number

**GLOBAL_PARM**
In the SysAdmin client, there’s a Miscellaneous section under Circulation, OPAC Configuration, and System. Some of the data on these screens is stored in GLOBAL_PARM, some is in MISCELLANEOUS. Some data from GLOBAL_PARM doesn’t appear in the Voyager clients at all.

The parms are:

AdvanceShortLoan
AllowCallslipBibSelect (Might be obsolete)
AllowCallslipReassign (Might be obsolete)
CALLNOPREFIXSUFFIX (If set to Y, then 852$km are included in mfhd_master.display_call_no; otherwise not.)

**CITATION_CALL_SLIP**
DeletePatronHistFines
DisplaySubfieldSeparators
EmailFineFeeNotice
EmailStatementOfFineFee
ILLCutOffDays
PACKAGE (Used to enable various optional modules)
ShortLoansIncrement
ShortLoansStart
ShortLoansTimeBuffer
StopCRProcessing
UBPatronUpdate (Obsolete. Deleted with 7.2.2.)
demerits
demeritsdisplay
saved_records_display1
saved_records_display2
saved_records_display3

parm character 25
value character 50

**HEADING**  p. 10, 32
The starred fields in this table are in UTF-8.

If a record has not been modified, the modify_date is null.

- heading_id number
- create_date date
- *display_heading character 300
- heading_type character 1
- index_type character 1
- *normal_heading character 300
- opacbibbs number
- opacrefs character 5
- staffbibbs number
- staffrefs character 5
- update_date date

**HEADING_CHANGE**
The starred field in this table is in UTF-8.

This tables links HEADING_CHANGE_FIELDS and HEADING_CHANGE_QUEUE.
There are some extra records in the table. If they cause catjob 13 to bomb, change the process_flag to Y. (See Knowledge Base 16384-1315.)

```
heading_change_id number
heading_id_new number
heading_id_old number
heading_queue_id number
change_date date
index_type character 1
*new_heading character 330
process_flag character 1
```

**HEADING_CHANGE_FIELDS**
The starred fields in this table are in UTF-8.

This table has a row for each bib to be changed.

```
heading_change_id number
rec_id number
change_date date
marc_ind1 character 1
marc_ind2 character 1
marc_tag character 3
*new_field character 330
*old_field character 330
rec_type character 1
```

**HEADING_CHANGE_QUEUE**
This table has a row for each entry in the change queue.

```
heading_id_new number
heading_id_old number
heading_queue_id number
rec_id number
change_date date
index_type character 1
process_flag character 1
rec_type character 1
```

**HEADING_SUBDIVISION**  p. 10, 32
```
heading_id number
subdiv_id number
```

**HEADING_TYPE**  p. 10, 32
Most of the data in this table are set by Ex Libris and cannot be changed by the customer, but the staffsuppress can be set in the SysAdmin client at Search, Heading Filters.

The starred fields in this table are in UTF-8.

```
*heading_code character 20
```

```
heading_type character 1
*heading_type_desc character 50
index_type character 1
staffsuppress character 1
```

**HEADING_VW**
```
auth_id number
heading_id number
create_date date
display_heading character 300
heading_type character 50
index_name character 30
normal_heading character 300
opacibs number
reference_type character 20
```

**HOLD_RECALL... Tables**
These tables are used for two distinct purposes.

If a hold or recall is placed for a patron, record of it appears in these tables from the time the hold or recall is placed. In this case, call_slip_id is zero.

If a call slip or a UB request is made for a patron, record of it appears in these tables for the pick up library from the time the item is routed to the pick up library. In this case, call_slip_id will not be zero.

**HOLD_RECALL**  p. 11, 12
If holding_db_id is zero or null, then the item or title belongs to your library. In this case, call_slip_id will point to a call slip in your database. If holding_db_id is neither zero nor null, the item or title belongs to another library. You can use VOYAGER_DATABASES to find out what library it belongs to. In this case, call_slip_id will NOT point to a call slip in your database.

When a hold or recall is archived, it is moved from HOLD_RECALL to HOLD_RECALL_ARCHIVE and any items are moved from HOLD_RECALL_ITEMS to HOLD_RECALL_ITEM_ARCHIVE. When this happens, the hold_recal_id changes. This number is assigned sequentially as records are added to HOLD_RECALL and HOLD_RECALL_ARCHIVE.
The values of request_level are C=Copy Level and T=Title Level.

The values of hold_recall_type are H=Hold and R=Recall.

The request_item_count is the number of items in HOLD_RECALL_ITEMS that could fill this hold or recall. If the value is -1, then the hold or recall has been cancelled.

If the item on hold belongs to another library, the call_slip_id will be the call_slip in the item’s home database, not your own.

The patron_group_id is null for UB transactions where the item is on hold someplace other than its home library.

If the item on hold belongs to another library, the HOLD_RECALL record is created when the item is discharged to the hold shelf. An available item notice is sent next time circjob5 runs. The create_date is the date of the patron’s request, not the date the HOLD_RECALL is created. The expire_date is set to the date the HOLD_RECALL is created plus the hold_shelf_life set in your library’s sys admin settings.

A HOLD_RECALL that is not picked up is archived by circjob6 when the expire_date has passed.

With V7.0, the 3 modify fields are added because holds and recalls can now be edited. If a record has not been modified, the modify_date is null.

If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

bib_id number
call_slip_id number
create_location_id number
create_opid character 10
hold_recall_id number
holding_db_id number
modify_location_id number
modify_opid character 10
patron_group_id number
patron_id number
request_group_id number

available_notice_count number
available_notice_date date
create_date date
expire_date date
hold_recall_type character 1
modify_date date
patron_comment character 100
pickup_location number
request_item_count number
request_level character 1

HOLD_RECALL_ARCHIVE  p. 11
If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

bib_id number
call_slip_id number
create_location_id number
hold_recall_id number
holding_db_id number
modify_location_id number
modify_opid character 10
patron_group_id number
patron_id number
request_group_id number
available_notice_count number
available_notice_date date
create_date date
create_opid character 10
expire_date date
hold_recall_type character 1
modify_date date
patron_comment character 100
pickup_location number
request_item_count number
request_level character 1

HOLD_RECALL_ITEMS  p. 11, 12
For a copy-level hold, there will be a row in HOLD_RECALL_ITEMS to identify the specific item.

If the item on the hold shelf belongs to another I-Share library, the item_id will be the item_id in the item’s home library, not yours. You can tell what library the item belongs to by checking holding_db_id in the corresponding HOLD_RECALL record.

hold_recall_id number
item_id number
hold_recall_status number
hold_recall_status_date date
hold_recall_type character 1
queue_position number

**HOLD_RECALL_ITEM_ARCHIVE** p. 11
hold_recall_id number
item_id number
hold_recall_status number
hold_recall_status_date date
hold_recall_type character 1

**HOLD_RECALL_STATS** p. 11
hold_recall_id number
patron_stat_id number

**HOLD_RECALL_STATUS** p. 11
hr_status_desc character 25
hr_status_type number

**IMPORT_RULE** p. 3, 34
Data in this table are defined in the SysAdmin client at Cataloging, Bulk Import Rules, Rules tab.

auth_dup_profile_id number
bib_dup_profile_id number
char_set_id number
import_rule_id number
import_rule_po_id number
library_id number
bib_dup_exist character 1
bib_to_mfhd character 1
cat_review character 1
code character 8
create_mfhds_items character 1
create_mfhds_only character 1
ignore_opac_suppress character 1
loc_field character 3
loc_ind1 character 1
loc_ind2 character 1
loc_subfield character 1
name character 25
order_create character 1
suppress_in_opac character 1

**IMPORT_RULE_BIBTOMFHD** p. 34
Data in this table display in the SysAdmin client at Cataloging, Bulk Import Rules, Rules tab.

import_rule_id number
mfhd_field character 3

**IMPORT_RULE_PO** p. 34

Data in this table display in the SysAdmin client at Cataloging, Bulk Import Rules.

account_id number
import_rule_po_id number
location_id_order number
vendor_id number
copy_default number
copy_field character3
copy_ind1 character1
copy_ind2 character1
copy_subfield character1
currency_code character3
fund_code character10
fund_field character3
fund_ind1 character1
fund_ind2 character1
fund_subfield character1
instruction_field character3
instruction_ind1 character1
instruction_ind2 character1
instruction_subfield character1
line_item_type_default number
line_item_type_field character3
line_item_type_ind1 character1
line_item_type_ind2 character1
line_item_type_subfield character1
notes_field character3
notes_ind1 character1
notes_ind2 character1
notes_subfield character1
order_type number
piece_field character3
piece_ind1 character1
piece_ind2 character1
piece_subfield character1
price_default number
price_field character3
price_ind1 character1
price_ind2 character1
price_subfield character1
title_ind1 character1
title_ind2 character1
title_no_field character3
title_no_subfield character1

**INDEX_TYPE** p. 10, 32
index_name character 30
index_type character 1
**INSTRUCTOR**  p. 8  
This table is part of reserves.  
circ_cluster_id number  
instructor_id number  
first_name character 40  
last_name character 50  
normal_last_name character 50  
title character 10

**INTERVAL_TYPE**  
interval_desc character 25  
interval_type character 1

**INVOICE**  p. 13, 17, 22  
The total field is reliable; the invoice_total is not.  
account_id number  
create_location_id number  
create_opid character 10  
update_location_id number  
update_opid character 10  
vendor_id number  
adJUSTMENTS_SUBTOTAL number  
bill_location number  
check_number character 40  
conversion_rate number  
currency_code character 3  
edi_ref number  
expend_date date  
Invoice_create_date date  
Invoice_date date  
Invoice_number character 25  
Invoice_quantity number  
Invoice_status number  
Invoice_status_date date  
Invoice_total number  
Invoice_update_date date  
line_item_count number  
line_item_subtotal number  
normal_check_number character 40  
normal_invoice_number character 25  
total number  
voucher_number character 25

**INVOICE_FUNDS**  p. 13  
This table sometimes gets out of synch with reality. A FullFundRepair will fix this. The INVOICE_LINE_ITEM_FUNDS table is more reliable.  
fund_id number  
invoice_id number  
ledger_id number  
commit_pending number  
commit_pending number  
commitments number  
expend_pending number  
expenditures number

**INVOICE_LINE_ITEM**  p. 17, 22  
This is the table that lets you move between a PO and its invoice.  
If a record has not been modified, the modify_date is null.  
create_opid character 10  
inv_line_item_id number  
invoice_id number  
line_item_id number  
update_opid character 10  
create_date date  
edi_ref number  
line_price number  
piece_identifier character 50  
prepay_amount number  
quantity number  
unit_price number  
update_date date

**INVOICE_LINE_ITEM_FUNDS**  p. 13, 22  
copy_id number  
fund_id number  
inv_line_item_id number  
ledger_id number  
amount number  
percentage number  
split_fund_seq number

**INVOICE_NOTE**  p. 22  
invoice_id number  
note character 1900

**INVOICE_STATUS**  p. 22  
invoice_status number  
invoice_status_desc character 25

**INV_LINE_ITEM_NOTES**  p. 22  
inv_line_item_id number  
invoice_id number  
note character 1900

**ISSUES_RECEIVED**  p. 25, 26  
To uniquely identify an issue, you need both issue_id and component_id.
In the opac_suppressed field, 0=suppressed, 1=not suppressed.

component_id number
copy_id number
issue_id number
item_id number
location_id number
collapsed character 1
note character 256
opac_suppressed number
receipt_date date

ISSUES_VW
This view is dropped in V7.0.

This view has a number of quirks and it is not efficient. Consider using the tables directly instead.

component_id number
issue_id number
enumchron character 256
expected_date date
receipt_date date
received number

ITEM  p. 2, 6, 7, 8, 9, 11, 12, 18, 19, 27, 28, 36, 37
The perm_location and temp_location fields can both be used to link to the location_id in the LOCATION table. Remember that there are locations in MFHDs as well as items, and that Voyager lets you change one without the other. The locations in ITEM determine the circulation policies; the location in MFHD_MASTER controls limiting and the call slip queue. If you want to count materials by location, remember that some of your bibs (e-resources, for example) may have MFHDs but not items.

The historical_browses count is incremented whenever an item that is not charged gets discharged. This happens in these situations: when items picked up from desks in the library are discharged (true browses) and when UB items are routed around the consortium (not true browses). So the historical_browses field is not an accurate count of browses (for libraries that use UB), but it still gives a general indication of how frequently an item is used.

By the way, historical_browses is never reset back to zero. Browses are not recorded anywhere else in the system and there’s no date associated with them. Consequently, the only way to get browse statistics for a time period is to take a count at the beginning and end of the period and subtract. CARLI collects historical browse data periodically; documentation is at http://www.carli.illinois.edu/products-services/i-share/reports/secure/histbrowstat

The historical_charges field is incremented when the item is discharged. It includes reserves charges and short loan charges. It may include transaction counts from your previous ILS. All CARLI libraries migrated counts from their previous systems except CSC, IEC, KEN, LAC, NBY, NPU, ONU, RSH, SFM, SIM, SVC, and USF.

The reserve_charges field is incremented by each charge while an item is on reserve. It is zeroed out when the item goes off reserve. Reserve charges are also counted in all the ways that normal charges are counted.

The various enumeration fields are in MHFD_ITEM, not ITEM.

The short_loan_charges field is the historical count of short loans. It is incremented when the item is charged. Short loan charges are also counted in all the ways that normal charges are counted.

It may be that modify_date is not reset for all item changes. It seems that giving an item a temp location and item type may not reset the date.

If a record has not been modified, the modify_date is null.

The copy_number field comes from the MFHD 852$t.

The magnetic_media and sensitize fields are added with V7.0.

create_location_id number
create_operator_id character 10
item_id number
item_type_id number
media_type_id number
modify_location_id number
modify_operator_id character 10
perm_location number
temp_item_type_id number
temp_location number
copy_number number
create_date date
historical_bookings number
historical_browses number
historical_charges number
holds_placed number
item_sequence_number number
magnetic_media character 1
modify_date date
on_reserve character 1
pieces number
price number
recalls_placed_number
reserve_charges number
sensitize character 1
short_loan_charges number
spine_label character 25

ITEM_BARCODE  p. 37
Before you use this table, consider: Are all your items barcoded? What about AV, microforms, or bound serials?
item_id number
barcode_status number
barcode_status_date date
item_barcode character 30

ITEM_BARCODE_STATUS  p. 37
barcode_status_desc character 25
barcode_status_type number

ITEM_NOTE  p. 37
item_id number
item_note character 1000

ITEM_STATS  p. 5, 6, 37
item_id number
item_stat_id number
date_applied date

ITEM_STATUS  p. 6, 11, 12, 27, 37
item_id number
item_status number
item_status_date date

ITEM_STATUS_TYPE  p. 6, 11, 12, 27, 37
item_status_desc character 25
item_status_type number

ITEM_STAT_CODE  p. 5, 6, 27, 37
Data in this table are defined in the SysAdmin client at System, Statistical Categories, Item tab.
item_stat_id number
item_stat_code character 3
item_stat_code_desc character 25

ITEM_TYPE  p. 6, 7, 27, 35, 37
Data in this table are defined in the SysAdmin client at System, Item Types.
item_type_id number
item_type_code character 10
item_type_display character 40
item_type_name character 25

ITEM_TYPE_MAPPING  p. 34
Data in this table are defined in the SysAdmin client at Cataloging, Bulk Import Rules, Mapping tab.
call_no_hierarchy_id number
import_rule_id number
item_type_id number
location_id number
marc_item_type character 50
marc_location character 50

ITEM_TYPE_POLICY  p. 7
Data in this table are defined in the SysAdmin client at Circulation, Policy Definitions, Items tab.
circ_group_id number
item_type_id number
order_quantity_number number
reorder_point number
replace_cost number
short_loan character 1

ITEM_VW
This view has a number of quirks and it is not efficient. Consider using the tables directly instead.

The gov_item_type fields are the item’s current type, in other words, the temp item type, if there is one, otherwise the item type. Similarly, the gov_location fields are the item’s temp location, if there is one, otherwise the perm location.
create_opid character 10  
item_id number  
mfhd_id number  
barcode character 30  
call_no character 300  
call_no_type character 1  
caption character 256  
chronology character 80  
create_date date  
enumeration character 80  
gov_item_type character 25  
gov_item_type_code character 10  
gov_location character 25  
gov_location_code character 10  
historical_bookings number  
historical_browses number  
historical_charges number  
holds_placed number  
media_type character 50  
media_type_code character 10  
normalized_call_no character 300  
perm_item_type character 25  
perm_item_type_code character 10  
perm_location character 25  
perm_location_code character 10  
recalls_placed number  
year character 20

If you are doing statistics by the alphabetic part of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetters in LCClassDetailed and show the Description field.

A set of techniques for producing statistics by more precise call number ranges is given in “Reports with Call Number Ranges: How to Request Then and How to Write Them” at http://www.carli.illinois.edu/products-services/i-share/reports/secure/callnumrange

lcclass_vw
This table parses LC class numbers, which makes statistics by call number ranges really slick. Consider the class number, ML410. Class=ML, classnumber=410, firstletter=M, longclass=MLbb410. Those b's are blanks. Longclass is padded with blanks so that numbers sort nicely. When a class number has a decimal point and digits following, these are not included in longclass.

Since V7.1, this table works correctly for classes that begin with 3 letters, such as LC law numbers.

If you want to sort a report (not a query, a report) by class, you will have to use the Left function to truncate it to less that 255 characters.

If you are doing statistics by the first letter of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetter in LCClassBrief and show the Description field.

lcclass_vw
This table parses LC class numbers, which makes statistics by call number ranges really slick. Consider the class number, ML410. Class=ML, classnumber=410, firstletter=M, longclass=MLbb410. Those b's are blanks. Longclass is padded with blanks so that numbers sort nicely. When a class number has a decimal point and digits following, these are not included in longclass.

Since V7.1, this table works correctly for classes that begin with 3 letters, such as LC law numbers.

If you want to sort a report (not a query, a report) by class, you will have to use the Left function to truncate it to less that 255 characters.

If you are doing statistics by the first letter of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetter in LCClassBrief and show the Description field.

If you are doing statistics by the alphabetic part of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetters in LCClassDetailed and show the Description field.

A set of techniques for producing statistics by more precise call number ranges is given in “Reports with Call Number Ranges: How to Request Then and How to Write Them” at http://www.carli.illinois.edu/products-services/i-share/reports/secure/callnumrange

The fiscal_year_id field in LEDGER can be used to link to fiscal_period_id in FISCAL_PERIOD and ROLLOVER_RULES. This isn’t obvious from the names.

If a record has not been modified, the modify_date is the same as the create_date.

If you are doing statistics by the alphabetic part of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetters in LCClassDetailed and show the Description field.

A set of techniques for producing statistics by more precise call number ranges is given in “Reports with Call Number Ranges: How to Request Then and How to Write Them” at http://www.carli.illinois.edu/products-services/i-share/reports/secure/callnumrange

If you are doing statistics by the alphabetic part of the LC class code, you can easily add a description of the class to your results. Link firstletter in LCCLASS_VW to ClassLetters in LCClassDetailed and show the Description field.
**LEDGER_LOCATIONS**  p. 13
ledger_id number
location_id number

**LEDGER_NOTE**  p. 13
ledger_id number
note character 1900

**LIBRARY**  p. 18, 27
Data in this table are defined in the SysAdmin client at System, Owning Libraries.

The starred field in this table is in UTF-8.

library_id number
library_display_name character 80
library_name character 50
*nuc_code character 15

**LIBRARY_ADDRESS_DEFAULT**
Data in this table are defined in the SysAdmin client at System, Default Address.

address_line1 character 50
address_line2 character 50
address_line3 character 50
address_line4 character 50
address_line5 character 50
city character 30
contact_name character 50
country character 20
email character 50
library_name character 50
san character 10
state_province character 7
zip_postal character 10

**LINE_ITEM... Tables**
These tables are part of purchase orders.

**LINE_ITEM**  p. 17, 22, 25
The standard_num field is added with V7.0.
bib_id number
create_opid character 10
line_item_id number
po_id number
update_opid character 10
cancel_interval number
claim_interval number
create_date date
donor character 50
edi_ref number
line_item_number number
line_item_type number
line_price number
piece_identifier character 50
prepay_amount number
print_std_num character 2
quantity number
requestor character 50
rush character 1
standard_num character 25
unit_price number
update_date date
vendor_ref_num character 35
vendor_ref_qual character 3
vendor_title_num character 25

**LINE_ITEM_BIB_HISTORY**
This table is new with V7.0 because links to bibs and holdings on a PO can now be changed.
audit_id number
bib_id number
create_opid character 10
line_item_id number
create_date date

**LINE_ITEM_COPY**  p. 22, 25
To determine the fund that is being used to purchase this item, link to FUND with use_fund linked to fund_id and use_ledger linked to ledger_id.

line_item_id number
location_id number
copy_count number
ship_to_location number
use_fund number
use_ledger number

**LINE_ITEM_COPY_HISTORY**  p. 17, 22, 25
audit_id number
copy_id number
inv_line_item_id number
line_item_status number
status_date date

**LINE_ITEM_COPY_MFHD_HISTORY**
This table is new with V7.0 because links to bibs and holdings on a PO can now be changed.
audit_id number
copy_id number
create_opid character 10
mfhd_id number
create_date date
LINE_ITEM_COPY_STATUS  p. 25
Both the line_item_status and the invoice_item_status field can be interpreted by linking to the line_item_status field in LINE_ITEM_STATUS.
- copy_id number
- item_id number
- line_item_id number
- location_id number
- mfhd_id number
- invoice_item_status number
- line_item_status number
- status_date date

LINE_ITEM_FUNDS  p. 13, 22
Remember that a fund_id does not uniquely identify a fund. It’s the combination of fund_id and ledger_id that uniquely identifies a fund. Consequently, you need to link by both of these fields when you are linking among the FUND..., PO_FUNDS and LINE_ITEM_FUNDS tables.
- copy_id number
- fund_id number
- ledger_id number
- amount number
- percentage number
- prepay number
- prepay_percentage number
- split_fund_seq number

LINE_ITEM_NOTES  p. 22
- line_item_id number
- po_id number
- note character 1900
- print_note character 60

LINE_ITEM_STATUS  p. 22
- line_item_status number
- line_item_status_desc character 25

LINE_ITEM_TYPE  p. 22
- line_item_type number
- line_item_type_desc character 25

LOCATION  p. 2, 6, 7, 11, 12, 13, 15, 16, 18, 19, 22, 23, 24, 25, 26, 31, 33, 34, 37, 38, 39
Data in this table are defined in the SysAdmin client at System, Locations.

Don’t use mfhd_count in statistics. It’s not at all reliable.

library_id number
location_id number
location_code character 10
location_display_name character 60
location_name character 25
location_opac character 1
location_spine_label character 25
mfhd_count number
suppress_in_opac character 1

LOCATION_ADDRESS
Data in this table are defined in the SysAdmin client at System, Locations, Address tab.

address_id number
location_id number
address_line1 character 50
address_line2 character 50
address_line3 character 50
address_line4 character 50
address_line5 character 50
bill_to_address character 1
campus_address character 1
circ_desk_address character 1
city character 30
contact_name character 50
country character 20
email character 50
other_address character 1
san character 10
ship_to_address character 1
state_province character 7
street_address character 1
zip_postal character 10

LOCATION_LIMIT
Data in this table are defined in the SysAdmin client at System, Location Limit Groups.

This table is used for WV and client searching to control location limiting in searches.

location_limit_id number
limit_code character 10
limit_name character 60
suppress_in_opac character 1

LOCATION_LIMIT_LOCS
Data in this table are defined in the SysAdmin client at System, Location Limit Groups.
This table is used for WV and client searching to control location limiting in searches.

- **location_id** number
- **location_limit_id** number

**LOCATION_PHONE**
Data in this table are defined in the SysAdmin client at System, Locations, Address tab.

- **address_id** number
- **phone_id** number
- **phone_number** character 25
- **phone_type** number

**MAINTENANCE** p. 16
This table is part of the media booking module.

- **create_location_id** number
- **create_opid** character 10
- **equip_id** number
- **maint_id** number
- **update_location_id** number
- **update_opid** character 10
- **create_date** date
- **date_in** date
- **date_out** date
- **update_date** date

**MAINTENANCE_DETAIL** p. 16
This table is part of the media booking module.

- **maint_dtl_id** number
- **maint_id** number
- **maint_type_id** number
- **detail_comment** character 100

**MAINTENANCE_NOTE** p. 16
This table is part of the media booking module.

- **maint_id** number
- **op_id** character 10
- **note** character 2000
- **update_date** date

**MAINTENANCE_QUEUE**
This table is at least a partial record of when indexes were regenerated for this database. The causation_comment field is always set to “upgrade” which isn’t true. Routine index regens should create records with maintenance_code set to K (=Keyword) and T (=TurboBibText). Other values are H=Heading, M=MFHDDIndex, E=BibTextTable, B=BibLeftAnchored, G=Geospatial, F=FacetedBib (part of geospatial), S=StatSampler, X=Bib856Links, Y=Auth856Links, Z=MFHD856Links.

- **causation_comment** character 2000
- **enqueue_date** date
- **maintenance_code** character 1
- **process_date** date
- **release_processed** character 30

**MAINTENANCE_TYPE** p. 16
This table is part of the media booking module.

- **maint_type_id** number
- **type** character 50
- **type_code** character 10

**MAP_INDEX**
The starred fields in this table are in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.

- **bib_id** number
- **map_index_id** number
- ***east_longitude_display** character 1
- **east_longitude_normal** number
- ***north_latitude_display** character 1
- **north_latitude_normal** number
- ***south_latitude_display** character 1
- **south_latitude_normal** number
- ***west_longitude_display** character 1
- **west_longitude_normal** number

**MAP_INDEX_G_RING**
The starred fields in this table are in UTF-8.

This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.

- **map_index_id** number
- ***g_ring_latitude** character 12
- **g_ring_latitude_normal** number
- ***g_ring_longitude** character 12
- **g_ring_longitude_normal** number
- **seqnum** number

**MAP_INDEX_SCALE**
This table is part of Voyager’s Geospatial module. Until we begin using that module, the table will not be useful.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>map_index_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>map_scale</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>scale_type</td>
<td>character 1</td>
<td></td>
</tr>
</tbody>
</table>

**MARC... Tables**

These tables parse out some of the fixed fields from bibliographic records. Remember that some of the fixed fields are also available in BIB_INDEX and BIB_TEXT. Voyager uses the record type and bib level fields to decide which records are included in each view.

For each MARC* table, the Access field name and the label from a WorldCat display are given.

**MARCBOOK_VW**

Includes these record type/bib level pairs: aa, ac, ad, am, ha, hc, hd, hm, ta, tc, td, tm.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>audience</td>
<td>Audn</td>
<td>008 byte 22</td>
</tr>
<tr>
<td>biography</td>
<td>Biog</td>
<td>008 byte 34</td>
</tr>
<tr>
<td>conferencepub</td>
<td>Conf</td>
<td>008 byte 29</td>
</tr>
<tr>
<td>governmentpub</td>
<td>GPub</td>
<td>008 byte 28</td>
</tr>
<tr>
<td>itemform</td>
<td>Form</td>
<td>008 byte 23</td>
</tr>
<tr>
<td>literaryform</td>
<td>LitF</td>
<td>008 byte 33</td>
</tr>
</tbody>
</table>

**MARCGRAPHIC_VW**

Includes these record type/bib level pairs: ea, ec, ed, em, fa, fc, fd, fm

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>cartographictype</td>
<td>CrTp</td>
<td>008 byte 25</td>
</tr>
<tr>
<td>governmentpub</td>
<td>GPub</td>
<td>008 byte 28</td>
</tr>
<tr>
<td>indexed</td>
<td>Indx</td>
<td>008 byte 31</td>
</tr>
<tr>
<td>projection</td>
<td>Proj</td>
<td>008 bytes 22-23</td>
</tr>
</tbody>
</table>

**MARCINDEX_VW**

Includes these record type/bib level pairs: ia, ic, id, im, ja, jc, jd, jm

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>audience</td>
<td>Audn</td>
<td>008 byte 22</td>
</tr>
<tr>
<td>biblevel</td>
<td>Bibl</td>
<td>008 byte 19</td>
</tr>
<tr>
<td>bibtype</td>
<td>Bibt</td>
<td>008 byte 20</td>
</tr>
<tr>
<td>compositionform</td>
<td>Comp</td>
<td>008 bytes 18-19</td>
</tr>
<tr>
<td>itemform</td>
<td>Form</td>
<td>008 byte 23</td>
</tr>
<tr>
<td>musicformat</td>
<td>FMus</td>
<td>008 byte 20</td>
</tr>
</tbody>
</table>

**MARCSpeaking_VW**

Includes these record type/bib level pairs: ab, as, bb, bs, cb, cs, db, ds, eb, es, fb, fs, gb, gs, hb, hs, ib, is, jb, js, kb, ks, mb, ms, nb, ns, ob, os, pb, ps, rb, rs, tb, ts

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>conferencepub</td>
<td>Conf</td>
<td>008 byte 29</td>
</tr>
<tr>
<td>entirenature</td>
<td>EntW</td>
<td>008 byte 24</td>
</tr>
<tr>
<td>frequency</td>
<td>Freq</td>
<td>008 byte 18</td>
</tr>
<tr>
<td>governmentpub</td>
<td>GPub</td>
<td>008 byte 28</td>
</tr>
<tr>
<td>itemform</td>
<td>Form</td>
<td>008 byte 23</td>
</tr>
<tr>
<td>originalform</td>
<td>Orig</td>
<td>008 byte 22</td>
</tr>
<tr>
<td>regularity</td>
<td>Regl</td>
<td>008 byte 19</td>
</tr>
<tr>
<td>type</td>
<td>SrTp</td>
<td>008 byte 21</td>
</tr>
</tbody>
</table>

**MARC население**

Includes these record type/bib level pairs: aa, ac, ad, am, ha, hc, hd, hm, ta, tc, td, tm.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>audience</td>
<td>Audn</td>
<td>008 byte 22</td>
</tr>
<tr>
<td>biblevel</td>
<td>Bibl</td>
<td>008 byte 19</td>
</tr>
<tr>
<td>bibtype</td>
<td>Bibt</td>
<td>008 byte 20</td>
</tr>
<tr>
<td>compositionform</td>
<td>Comp</td>
<td>008 bytes 18-19</td>
</tr>
<tr>
<td>itemform</td>
<td>Form</td>
<td>008 byte 23</td>
</tr>
<tr>
<td>musicformat</td>
<td>FMus</td>
<td>008 byte 20</td>
</tr>
</tbody>
</table>

**MARC SERIAL_VW**

Includes these record type/bib level pairs: ab, as, bb, bs, cb, cs, db, ds, eb, es, fb, fs, gb, gs, hb, hs, ib, is, jb, js, kb, ks, mb, ms, nb, ns, ob, os, pb, ps, rb, rs, tb, ts

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>bib_id</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>biblevel</td>
<td>Bibl</td>
<td>008 byte 19</td>
</tr>
<tr>
<td>bibtype</td>
<td>Bibt</td>
<td>008 byte 20</td>
</tr>
<tr>
<td>conferencepub</td>
<td>Conf</td>
<td>008 byte 29</td>
</tr>
<tr>
<td>entirenature</td>
<td>EntW</td>
<td>008 byte 24</td>
</tr>
<tr>
<td>frequency</td>
<td>Freq</td>
<td>008 byte 18</td>
</tr>
<tr>
<td>governmentpub</td>
<td>GPub</td>
<td>008 byte 28</td>
</tr>
<tr>
<td>itemform</td>
<td>Form</td>
<td>008 byte 23</td>
</tr>
<tr>
<td>originalform</td>
<td>Orig</td>
<td>008 byte 22</td>
</tr>
<tr>
<td>regularity</td>
<td>Regl</td>
<td>008 byte 19</td>
</tr>
<tr>
<td>type</td>
<td>SrTp</td>
<td>008 byte 21</td>
</tr>
</tbody>
</table>
type character 1

**MARCVISUAL_VW**
Includes these record type/bib level pairs: ga, gc, gd, gm, ka, kc, kd, km, na, nc, nd, nm, oa, oc, od, om, ra, rc, rd, rm

- audience: Audn 008 byte 22
- governmentpub: GPub 008 byte 28
- runningtime: Time 008 bytes 18-20
- technique: Tech 008 byte 34
- visualtype: TMat 008 byte 33
- bib_id number
- audience character 1
- biblevel character 1
- bibtype character 1
- governmentpub character 1
- runningtime character 3
- technique character 1
- visualtype character 1

**MARKED_ISSUE** p. 25
- component_id number
- copy_id number
- issue_id number
- location_id number
- marked_id number
- op_id character 10
- subscription_id number
- mark_date date
- mark_reason number
- marked_comment character 250

**MARKED_LINE_ITEM** p. 17
- copy_id number
- line_item_id number
- location_id number
- marked_id number
- op_id character 10
- mark_date date
- mark_reason number
- marked_comment character 250

**MARK_REASON** p. 17, 25
Data in this table are defined in the SysAdmin client at Acquisitions, Mark Reasons.

- mark_reason_id number
- claim_type number
- mark_reason_name character 25

**MASTER_OPERATOR** p. 38
Data in this table are defined in the SysAdmin client at Security, Master Profiles, Operator tab and display at Operator, Current Profiles.

- master_profile_id number
- operator_id character 10

**MASTER_PROFILE** p. 38
Data in this table are defined in the SysAdmin client at Security, Master Profiles, and display at Operator, Current Profiles.

- master_profile_id number
- acq_policies character 1
- cat_policies character 1
- circ_policies character 1
- cluster_create character 1
- cluster_delete character 1
- cluster_edit character 1
- cluster_view character 1
- currency_tables character 1
- master_profile_name character 25
- media_policies character 1
- patron_group_edit character 1
- security character 1
- system_definitions character 1

**MASTER_SECURITY_LOCS** p. 38
Data in this table are defined in the SysAdmin client at Security, Master Profiles, Locations tab.

- location_id number
- master_profile_id number

**MEDIA... Tables**
These tables are part of the Voyager media scheduling module.

**MEDIA_BOOKING_EXCEPTION**
- equip_id number
- equip_sts_type_id number
- item_id number
- location_id number
- media_booking_exception_id number
- media_room_id number
- op_id character 10
- patron_id number
- room_sts_type_id number
- action character 1
- item_status_type number
- update_date date
MEDIA_BOOKING_EXCEPTION_TYPE
media_booking_exception_id number
media_booking_exception character 25

MEDIA_BOOKING_TYPE
media_booking_type_id number
type character 20

MEDIA_OPERATOR
patron_id number
update_location_id number
update_opid character 10
status character 1
update_date date

MEDIA_POLICY_DELIVERYCALENDAR
calendar_id number
media_schedule_policy_id number

MEDIA_POLICY_EQUIPMENT_MATRIX
equip_type_id number
matrix_id number
media_schedule_policy_id number
patron_group_id number
settings_id number

MEDIA_POLICY_EQUIPMENT_TYPE
equip_type_id number
media_schedule_policy_id number
cleanup_time number
replacement_default number
request_equip_using_opac character 1
setup_time number

MEDIA_POLICY_EQUIPSETTINGS
settings_id number
booking_interval character 1
booking_period_max number
booking_renew number
booking_renew_count number
can_deliver character 1
can_pickup character 1
fine_grace_period number
fine_interval character 1
fine_max number
fine_rate_delivery number
fine_rate_pickup number
settings_name character 40
usage_fee number
usage_rate number
usage_rate_interval character 1
usage_rate_period number

MEDIA_POLICY_GROUP
media_schedule_policy_id number
block_interval number
cancel_unclaimed_booking number
delivery_count_closed character 1
delivery_count_closed_fees character 1
overdue_conflict_list_interval number
overdue_first_interval number
overdue_lost_fee character 1
overdue_lost_fee_amt number
overdue_lost_interval number
overdue_lost_max_fine character 1
overdue_notice_count number
overdue_notice_interval_number
overdue_renew character 1
pickup_count_closed character 1
pickup_count_closed_fees character 1
schedule_policy character 40
warning_interval number

MEDIA_POLICY_ITEM_MATRIX
matrix_id number
media_schedule_policy_id number
media_type_id number
patron_group_id number
settings_id number

MEDIA_POLICY_ITEM_SETTINGS
settings_id number
booking_interval character 1
booking_period_max number
booking_renew number
booking_renew_count number
can_deliver character 1
can_pickup character 1
fine_grace_period number
fine_interval character 1
fine_max number
fine_rate_delivery number
fine_rate_pickup number
recall_for_booking character 1
settings_name character 40
usage_fee number
usage_rate number
usage_rate_interval character 1
usage_rate_period number

MEDIA_POLICY_LOCATION
location_id number
media_schedule_policy_id number
print_location_id number
booking character 1
collect_fines character 1
courtesy_discharge_equipment char
courtesy_discharge_item character 1
delivery character 1
delivery_slip_print character 1
delivery_time number
equip_restricted character 1
fly_item_location number
fly_item_suppress character 1
fly_item_type number
item_shelving_interval character 1
item_shelving_period number
item_transit_period number
pickup character 1
pickup_slip_print character 1
print_confirmation character 1
return_time number
MEDIA_POLICY_MEDIA_TYPE
media_schedule_policy_id number
media_type_id number
replacement_default number
request_item_using_opac character 1
MEDIA_POLICY_PATRON_GROUP
media_schedule_policy_id number
patron_group_id number
booking_limit character 1
booking_max number
cancelled_booking_limit character 1
cancelled.booking_max number
early_pickup number
early_pickup_interval character 1
equip_booking_limit character 1
equip_booking_max number
fees_apply character 1
fines_apply character 1
item_booking_limit character 1
item_booking_max number
late_return_limit character 1
late_return_max number
outstanding_balance_limit character 1
outstanding_balance_max number
overdue_notice_apply character 1
overlapping_bookings character 1
request_patron_using_opac character 1
unclaimed_booking_limit character 1
unclaimed_booking_max number
MEDIA_POLICY_PICKUP_CALENDAR
calendar_id number
media_schedule_policy_id number
MEDIA_POLICY_ROOM_CALENDAR
calendar_id number
media_schedule_policy_id number
MEDIA_POLICY_ROOM_MATRIX
matrix_id number
media_room_type_id number
media_schedule_policy_id number
patron_group_id number
settings_id number
MEDIA_POLICY_ROOM_SETTINGS
settings_id number
booking_interval character 1
booking_period_max number
can_book character 1
settings_name character 40
usage_fee number
usage_rate number
usage_rate_interval character 1
usage_rate_period number
MEDIA_POLICY_ROOM_TYPE
media_room_type_id number
media_schedule_policy_id number
can_deliver character 1
room_scheduled character 1
MEDIA_ROOM p. 15, 16
create_location_id number
create_opid character 10
location_id number
media_room_id number
media_room_type_id number
update_location_id number
update_opid character 10
capacity number
create_date date
historical_bookings number
room_name character 40
room_name_normalized character 40
room_no character 15
room_no_normalized character 15
storage character 1
update_date date
MEDIA_ROOM_DETAILS p. 16
media_room_details_id number
media_room_dtl_type_id number
media_room_id number
room_dtl character 100

**MEDIA_ROOM_DETAIL_TYPE** p. 16
media_room_dtl_type_id number
repeatable character 1
type character 50
type_code character 10

**MEDIA_ROOM_KEY**
media_room_id number
media_room_key_id number
key_no character 15

**MEDIA_ROOM_NOTES** p. 16
media_room_id number
media_room_note_type_id number
op_id character 10
note character 2000
update_date date

**MEDIA_ROOM_NOTE_TYPE** p. 16
media_room_note_type_id number
type character 15

**MEDIA_ROOM_STATUS** p. 16
media_room_id number
media_room_sts_type_id number
op_id character 10
note character 100
update_date date

**MEDIA_ROOM_STATUS_TYPE** p.16
media_room_sts_type_id number
block_booking character 1
display_priority number
message character 50
sts_type character 40
warn_on_booking character 1

**MEDIA_ROOM_TYPE** p. 15, 16
media_room_type_id number
equip_storage character 1
type character 50
type_code character 10

**MEDIA_SCHEDULE** p. 15
create_location_id number
create_opid character 10
media_booking_type_id number
media_schedule_id number
media_schedule_policy_id number
patron_group_id number
patron_id number
patron_id_picked_up number
staging_location_id number
update_location_id number
update_opid character 10
admin_booking character 1
booking_cleanup number
booking_cleanup_date date
booking_end date
booking_setup number
booking_setup_date date
booking_start date
confirm_date date
confirm_no character 77
create_date date
operator_delivery number
operator_pickup number
sched_comment character 1000
update_date date
wizard character 1

**MEDIA_SCHEDULE_ARCHIVE**
create_location_id number
create_opid character 10
media_booking_type_id number
media_schedule_id number
media_schedule_policy_id number
patron_group_id number
patron_id number
patron_id_picked_up number
staging_location_id number
update_location_id number
update_opid character 10
admin_booking character 1
booking_cleanup number
booking_end date
booking_result number
booking_setup number
booking_start date
confirm_date date
confirm_no character 77
create_date date
operator_delivery number
operator_pickup number
sched_comment character 1000
update_date date
wizard character 1

**MEDIA_SCHEDULE_EQUIPMENT** p. 15
count_id number
equip_id number
<table>
<thead>
<tr>
<th>Table</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>equip_type_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>op_id</td>
<td>character 10</td>
</tr>
<tr>
<td>fulfill_item</td>
<td>character 1</td>
</tr>
<tr>
<td>in_room</td>
<td>character 1</td>
</tr>
<tr>
<td>update_date</td>
<td>date</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_EQUIP_ARCHIVE</td>
<td></td>
</tr>
<tr>
<td>count_id</td>
<td>number</td>
</tr>
<tr>
<td>equip_id</td>
<td>number</td>
</tr>
<tr>
<td>equip_type_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>op_id</td>
<td>character 10</td>
</tr>
<tr>
<td>fulfill_item</td>
<td>character 1</td>
</tr>
<tr>
<td>in_room</td>
<td>character 1</td>
</tr>
<tr>
<td>update_date</td>
<td>date</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_FINE</td>
<td></td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_FINE_ARCHIVE</td>
<td></td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_ITEM</td>
<td>p. 15</td>
</tr>
<tr>
<td>bib_id</td>
<td>number</td>
</tr>
<tr>
<td>count_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>media_type_id</td>
<td>number</td>
</tr>
<tr>
<td>mfhd_id</td>
<td>number</td>
</tr>
<tr>
<td>op_id</td>
<td>character 10</td>
</tr>
<tr>
<td>update_date</td>
<td>date</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_ITEM_ARCHIVE</td>
<td></td>
</tr>
<tr>
<td>bib_id</td>
<td>number</td>
</tr>
<tr>
<td>count_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>item_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_ROOM</td>
<td>p. 15</td>
</tr>
<tr>
<td>count_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_room_id</td>
<td>number</td>
</tr>
<tr>
<td>media_room_type_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>op_id</td>
<td>character 10</td>
</tr>
<tr>
<td>room_key_id</td>
<td>number</td>
</tr>
<tr>
<td>trans_location_id</td>
<td>number</td>
</tr>
<tr>
<td>capacity_needed</td>
<td>number</td>
</tr>
<tr>
<td>capacity_operator</td>
<td>number</td>
</tr>
<tr>
<td>update_date</td>
<td>date</td>
</tr>
<tr>
<td>MEDIA_SCHEDULE_ROOM_ARCHIVE</td>
<td></td>
</tr>
<tr>
<td>count_id</td>
<td>number</td>
</tr>
<tr>
<td>fine_fee_id</td>
<td>number</td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_room_id</td>
<td>number</td>
</tr>
<tr>
<td>media_room_type_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_id</td>
<td>number</td>
</tr>
<tr>
<td>media_schedule_trans_type_id</td>
<td>number</td>
</tr>
<tr>
<td>MEDIA_SECURITY_LOCATION</td>
<td></td>
</tr>
<tr>
<td>location_id</td>
<td>number</td>
</tr>
<tr>
<td>media_profile_id</td>
<td>number</td>
</tr>
<tr>
<td>MEDIA_SECURITY_OPERATOR</td>
<td></td>
</tr>
<tr>
<td>media_profile_id</td>
<td>number</td>
</tr>
<tr>
<td>operator_id</td>
<td>character 10</td>
</tr>
<tr>
<td>MEDIA_SECURITY_PROFILE</td>
<td></td>
</tr>
<tr>
<td>media_profile_id</td>
<td>number</td>
</tr>
<tr>
<td>booking_add</td>
<td>character 1</td>
</tr>
<tr>
<td>booking_cancel</td>
<td>character 1</td>
</tr>
<tr>
<td>booking_charge</td>
<td>character 1</td>
</tr>
</tbody>
</table>
When you are searching record_segment, it is helpful to know that CHR(31) is the subfield delimiter, CHR(30) is the end of field delimiter, and CHR(29) is the end of record delimiter.

```
mfhd_id number
*record_segment character 300
seqnum number
```

**MFHD_HISTORY** p. 27
There’s an error in some versions of the E-R diagrams. Action_type_id has a value between 1 and 6 and it is interpreted by linking to the ACTION_TYPE table.

When a MFHD is deleted, its MFHD_HISTORY records are deleted too.

The encoding_level and suppress_in_opac are the values after the transaction.

```
action_type_id number
location_id number
mfhd_id number
operator_id character 10
action_date date
encoding_level character 1
suppress_in_opac character 1
```

**MFHD_ITEM** p. 2, 18, 27, 30, 37
The chron field comes from serials check-in. It will match the value in enumchron in the SERIAL_ISSUES table.

The item_enum field comes from the MFHD 853/863 interaction.

```
item_id number
mfhd_id number
caption character 256
chron character 80
freetext character 256
item_enum character 80
year character 20
```

**MFHD_MASTER** p. 2, 8, 17, 18, 22, 25, 30, 33, 36, 37
The starred fields in this table are in UTF-8.

Call_no_type is usually the same as the 852 first indicator, but there are exceptions. If there’s no
852$h, call_no_type will be blank. If the indicator shows that the call number should be LC, Dewey, SuDoc, or NLM, but the call number in 852$h cannot be parsed by Voyager according to its rules for the class scheme, then call_no_type will be set to 8. If the 852 first indicator is 7 and subfield $2 is not a classification scheme that Voyager recognizes, then call_no_type will be set to 8. If the 852 first indicator is 7 and subfield $2 is empty, then call_no_type will be set to blank.

For most libraries, the display_call_no includes 852 subfields $k, $h, $i, and $m in that order. The normalized_call_no contains only $h, $i, and $m. Since call number prefixes are in 852$k, you’ll find them only in display_call_no. However, when your library came up on Voyager, a decision may have been made not to include 852 $k and $m. You can see this decision in the GLOBAL_PARM table, but you can’t see it in the SysAdmin client and you can’t change your mind. CARLI libraries IIT and TIU do not have 852 $k and $m in display_call_no and normalized_call_no.

Generally, call numbers are sorted by normalized_call_no. However, in staff client searches with location limiting, the sort is by display_call_no.

The value of display_call_no is NULL when there is no 852$h. The value of normalized_call_no is NULL when 1) there is no 852$h, or 2) there is an 852$h but it’s empty, or 3) 852 Ind1=blank, or 4) 852$h cannot be parsed by Voyager according to the rules for the classification indicated by Ind1.

For Dewey numbers, normalized_call_number begins with the Dewey class (including the decimal), one space, followed by the book number, which may have spaces embedded. For LC numbers, normalized_call_number begins with the class code, followed by the whole number portion of the class code right justified in a 5-character field. If the class code includes decimals, the decimal digits follow (without the decimal point). Then comes some spaces, followed by the rest of the book number, which may include embedded space. Examples using carets to show spaces: KFH1396.2 = KFH^13962, F868=F^^868, DA5=DA^^^^5.

Remember that there are locations in ITEM as well as MFHD_MASTER, and that Voyager lets you change one without the other. The locations in ITEM determine the circulation policies; the location in MFHD_MASTER controls limiting.

If you want to sort by display_call_no or normalized_call_no in a report (not a query, a report) you will need to use the Left function to cut them to fewer than 255 characters.

If a record has not been modified, the modify_date is null.

Remember that there are locations in ITEM as well as MFHD_MASTER, and that Voyager lets you change one without the other. The locations in ITEM determine the circulation policies; the location in MFHD_MASTER controls limiting.

For most libraries, the display_call_no includes 852 subfields $k, $h, $i, and $m in that order. The normalized_call_no contains only $h, $i, and $m. Since call number prefixes are in 852$k, you’ll find them only in display_call_no. However, when your library came up on Voyager, a decision may have been made not to include 852 $k and $m. You can see this decision in the GLOBAL_PARM table, but you can’t see it in the SysAdmin client and you can’t change your mind. CARLI libraries IIT and TIU do not have 852 $k and $m in display_call_no and normalized_call_no.

Generally, call numbers are sorted by normalized_call_no. However, in staff client searches with location limiting, the sort is by display_call_no.

The value of display_call_no is NULL when there is no 852$h. The value of normalized_call_no is NULL when 1) there is no 852$h, or 2) there is an 852$h but it’s empty, or 3) 852 Ind1=blank, or 4) 852$h cannot be parsed by Voyager according to the rules for the classification indicated by Ind1.

For Dewey numbers, normalized_call_number begins with the Dewey class (including the decimal), one space, followed by the book number, which may have spaces embedded. For LC numbers, normalized_call_number begins with the class code, followed by the whole number portion of the class code right justified in a 5-character field. If the class code includes decimals, the decimal digits follow (without the decimal point). Then comes some spaces, followed by the rest of the book number, which may include embedded space. Examples using carets to show spaces: KFH1396.2 = KFH^13962, F868=F^^868, DA5=DA^^^^5.

Remember that there are locations in ITEM as well as MFHD_MASTER, and that Voyager lets you change one without the other. The locations in ITEM determine the circulation policies; the location in MFHD_MASTER controls limiting.

If you want to sort by display_call_no or normalized_call_no in a report (not a query, a report) you will need to use the Left function to cut them to fewer than 255 characters.

If a record has not been modified, the modify_date is null.

export_ok_location_id number
export_ok_opid character 10
location_id number
mfhd_id number
call_no_type character 1
create_date date
*display_call_no character 300
encoding_level character 1
export_date date
export_ok character 1
export_ok_date character 1
*field_007 character 23
*field_008 character 32
*normalized_call_no character 300
record_status character 1
record_type character 1
source_module character 1
suppress_in_opac character 1
update_date date

MISCELLANEOUS
In the SysAdmin client, there’s a Miscellaneous section under Circulation, OPAC Configuration, and System. Some of the data on these screens is stored in MISCELLANEOUS, some is in GLOBAL_PARM. Some data in MISCELLANEOUS doesn’t appear in the Voyager clients at all.

For the auto_retrieve_system field, N=ARS is not available, Y=ARS is available and only the item barcode is exported, C=ARS is available and the item barcode, call#, author, and title are exported.

authreadonly character 1
auto_retrieve_system character 1
bibreadonly character 1
call_slip_item_required character 1
The MONO_CLAIM Tables

A claim is uniquely identified by copy_id, claim_thread, and claim_id.

The claim_count tells you which claim this is (first, second, etc.). For the most recent claim, claim_status=1; otherwise claim_status=0.

The claim_type may be interpreted using the CLAIM_TYPES table.

The claim_date is the date when the order should be claimed. If it has been overridden, the new date is in override_claim_date.

MONO_CLAIM  p. 17

- claim_id number
- copy_id number
- op_id character 10
- vendor_id number
- claim_count number
- claim_date date
- claim_status number
- claim_thread number
- claim_type number
- edi_ref number
- note character 256
- override_claim_date date

MONO_CLAIM_ARCHIVE

- claim_id number
- copy_id number
- op_id character 10
- vendor_id number
- archive_date date
- claim_count number
- claim_date date
- claim_status number

MONO_SUPPLIER_REPORT  p. 17

- audit_id number
- claim_id number
- action_date date
- action_quantity number
- edi_ref number
- note character 512
- report_date date
- report_type number

MY_OPAC_DB

- db_id number
- patron_id number

MY_OPAC PREFERENCES

- patron_id number
- search_preferences character 50

NALCLASS_VW

If you want to sort a report (not a query, a report) by longclass, you will have to use the Left function to truncate it to less than 255 characters.

- mfhd_id number
- class character 3
- longclass character 300

NLMCLASS_VW

This table parses NLM class numbers, which makes statistics by call number ranges really slick. The parsing algorithm is the same one used for LC numbers. Consider the class number, QS110. Class=QS, classnumber=110, firstletter=Q, longclass=QSbb110. Those b’s are blanks. Longclass is padded with blanks so that numbers sort nicely. Be warned that this table does not work correctly for classes that begin with 3 letters, such as the 19th century class schedule.

If you want to sort a report (not a query, a report) by class, you will have to use the Left function to truncate it to less than 255 characters.

A set of techniques for producing statistics by more precise call number ranges is given in “Reports with Call Number Ranges: How to Request Then and How to Write Them” at
http://www.carli.illinois.edu/products-services/i-share/reports/secure/callnumrange

mfhd_id number
class character 300
classnumber number
firstletter character 1
longclass character 7

NOTE_TYPE p. 20
This table is used with PATRON_NOTES.
note_desc character 25
note_type number

NO_FILL_REASON p. 2, 30
Data in this table are defined in the SysAdmin client at Call Slips, No-Fill Reasons.

This table is part of call slip processing.

reason_id number
reason_code character 10
reason_desc character 50
suppress character 1

OPAC_CHANGE_TYPE
opac_change_desc character 25
opac_change_type number

OPAC_CIRC_SETTINGS
Data in this table are defined in the SysAdmin client at OPAC Configuration, Patron Self-registration.

patron_purge_period number
self_reg_dfltt_patron_grp number

OPAC_FORM
Data in this table are defined in the SysAdmin client at OPAC Configuration, Request Forms.

form_id number
blank_form character 1
email character 100
form_code character 10
form_name character 40
form_type character 1
instructions character 1000
login character 15
output_type character 1
password character 15
suppress_in_opac character 1
voucher_last_used number
voucher_prefix character 4
voucher_start number

OPAC_FORM_DATABASES
Data in this table are defined in the SysAdmin client at OPAC Configuration, Request Forms, Select Form tab.

db_id number
form_id number
db_code character 8

OPAC_FORM_FIELDS
Data in this table are defined in the SysAdmin client at OPAC Configuration, Request Forms, Data tab.

In opac_change, 1=No edit/no display, 2=No edit/display, 3=Editable.

form_id number
clio_tag character 20
field_label character 20
field_required character 1
field_sequence number
mapping character 10
opac_change number

OPAC_FORM_PATRON_GROUP
Data in this table are defined in the SysAdmin client at OPAC Configuration, Request Forms, Patron Group tab.

form_id number
patron_group_id number

OPAC_FORM_REQUEST_FILE

bib_id number
form_id number
item_id number
mfhd_id number
patron_group_id number
patron_id number
request_id number
date_requested date
email_text character 2000
expire_date date
free_text1 character 100
free_text2 character 100
free_text3 character 100
free_text4 character 100
free_text5 character 100
free_text6 character 100
note character 100
voucher_number number

**OPAC_FORM_TYPE**
Data in this table are defined in the SysAdmin client at OPAC Configuration, Request Forms.

form_type character 1
form_type_desc character 25

**OPAC_MESSAGES**
line_number number
opac_line character 70
tab_number number

**OPAC_SEARCH_LOG**
When OPAC logging is turned on in Voyager, Web Voyage searches done against your database are logged in this table and in BIB_USAGE_LOG. The table is documented in the Voyager Technical Users Guide. The values are not completely consistent, so try to find what you want in several ways. Here are some additional comments:

- The client_ip field is the IP address of one of CARLI's Web Voyage servers, not the user's workstation, so it's not very useful. However, at other Voyager sites it might be the address of the workstation.
- The index_type field has values of A=Authority search, B=Browse, K=Keyword, and L=Left-anchored, but the field is often blank or null at times when search_string or search_type indicate an authority, browse, keyword, or left-anchored search.
- The limit_flag field is set to Y or N. Usually, if limit_flag is N, then limit_string is null, but there are exceptions, so think twice about which field you trust.
- For browse searches, hits=-1.
- The limit_string field lists the limits were in effect for a search: DATE, LANG, LOCA, MEDI (medium), PLAC (place), STAT (status), and TYPE. The values following TYPE are the record type and bib level of the bibs: am=books, as=magazines/journals, gm=films/videos, c?=scores, j?=music recordings, i?=spoken word recordings, e?=maps, m?=computer files, o?=kits, rm=3D objects.
- The search_string field has the search argument (e.g. GONE WITH THE WIND), which is often preceded by a search code. You can look up a search code in the SEARCHPARM table.
- The search_tab field has values of 1=quick search, 2=guided keyword, 3=course reserves.
- The values of the client_type field should be W=Web Voyage, G=Web Voyage, Z=Z39.50, and A=ASCII OPAC (which is no longer supported by Ex Libris.) However, bug 88568 in Voyager 2001.2 says that other values may be found here that oughtn't.

session_id character 16
client_ip character 15
client_type character 1
dbkey character 100
hits number
hyperlink character 1
index_type character 1
limit_flag character 1
limit_string character 250
redirect_flag character 1
relevance character 1
search_date date
search_string character 250
search_tab character 1
search_type character 25
stat_string character 15

**OPERATOR** p. 23, 24
CARLI has not allowed access to this table by library staff because operator passwords are stored here. Remember that operator_id is a text field, not a number. If you choose operator_id’s that are reasonably mnemonic, you can probably get by without the operator’s name, which is the most useful field in this table. File a work request if you need other information from this table.

Data in this table are defined in the SysAdmin client at Security, Operator Profiles, Operator tab.

If a record has not been modified, the modify_date is the same as the create_date.
create_opid character 10
modify_opid character 10
operator_id character 10
create_date date
first_name character 25
last_name character 25
middle_initial character 1
modify_date date

ORDER_TYPES
The values in this table are set by Ex Libris and
cannot be changed by the customer. In the
SysAdmin client, they figure in Acquisitions,
Policy Definitions, Vendor Policies tab and
Cataloging, Bulk Import Rules.
order_type number
order_type_desc character 25

PATCH_REGISTRY
This table can tell you when your Voyager
upgrades were done.
patch_opid character 30
patch_date date
patch_file character 30
patch_status character 30
release_processed character 30

PATRON
The items_recalled field is the number of items
currently charged to this patron which have been
recalled for another patron.

For privacy reasons, avoid including SSN in
reports. If you print reports with the SSN, be sure
to dispose of them properly.

The suspension_date is the date on which a
patron’s suspension ends. Patrons with nothing
in this field or with dates before today’s date are
not suspended.

If a record has not been modified, the
modify_date is the same as the create_date.

In name_type, 1=personal name, 2=institutional
name.

The rest of the comments about this table are
relevant only for sites that use UB.

Patrons affiliated with your library have db_id=0.
Stub patrons have a db_id greater than 1. Link
db_id to VOYAGER_DATABASES to find the
patron’s affiliation.

Records for patrons affiliated with another library
are either stub records or child records. Stub
records have modify_operator_id='SYS-UB'; they
are deleted nightly by circjob29 when they are no
longer needed. Child records have been modified
by staff at your library, so the modify_operator_id
is the ID of one of your staff; they are deleted
nightly by circjob29 when they are no longer
needed AND their purge_date has passed.
CARLI runs a script right before circjob29 runs to
changes children back to stubs so that they can be
deleted if they are no longer needed.

If you requested an SSN Purge from CARLI, the
following fields may have been reset to zero at
the time of the purge: claims_return_ub,
current_charges_ub, historical_charges_ub,
historical_requests_ub, lost_items_ub,
requests_ub, self_shelved_ub, total_fees_due_ub.

The current_charges_ub and requests_ub
counters are reliable since V7.1.

The counters, historical*, lost_items,
claims_return, self_shelved, etc., may have been
initialized when your library migrated from your
previous system to Voyager. The CARLI libraries
that migrated in 2002 initialized
historical_requests, claims_return, self_shelved,
and lost_items, although lost_items has since
been reset. The CARLI libraries that migrated in
2012 initialized historical_charges.
The values of the address_type field are:
1=permanent address, 2=temporary address,
3=email address.

Ex Libris says that type 2 addresses are not copied
into stub patron records, but this seems to be true
only sometimes.

The value of address_status is H=hold, N=no-
hold.

When a patron record is updated by a batch
patron load, all its addresses are deleted and re-
added. Consequently, the values in address_id
grow faster than the _id fields in other patron
tables.

address_id number
modify_operator_id character 10
patron_id number
address_line1 character 50
address_line2 character 40
address_line3 character 40
address_line4 character 40
address_line5 character 40
address_status character 1
address_type number
city character 30
country character 20
effect_date date
expire_date date
modify_date date
protect_address character 1
state_province character 7
zip_postal character 10

PATRON_BARCODE  p. 2, 6, 11, 15, 20, 21, 28,
29, 30

Records for patrons affiliated with another library
are either stub records or child records. Stub
records have modify_operator_id='SYS-UB'; they
are deleted nightly by circjob29 when they are no
longer needed. Child record have been modified
by staff at your library, so the modify_operator_id
is the ID of one of your staff; they are deleted
nightly by circjob29 when they are no longer
needed AND their purge_date has passed.

The home_barcode_id and
home_patron_group_id are filled in for UB stub
patron records. They are values from the patron's

The values of the address_type field are:
1=permanent address, 2=temporary address,
3=email address.

Ex Libris says that type 2 addresses are not copied
into stub patron records, but this seems to be true
only sometimes.

The value of address_status is H=hold, N=no-
hold.

When a patron record is updated by a batch
patron load, all its addresses are deleted and re-
added. Consequently, the values in address_id
grow faster than the _id fields in other patron
tables.

address_id number
modify_operator_id character 10
patron_id number
address_line1 character 50
address_line2 character 40
address_line3 character 40
address_line4 character 40
address_line5 character 40
address_status character 1
address_type number
city character 30
country character 20
effect_date date
expire_date date
modify_date date
protect_address character 1
state_province character 7
zip_postal character 10

PATRON_BARCODE  p. 2, 6, 11, 15, 20, 21, 28,
29, 30

Records for patrons affiliated with another library
are either stub records or child records. Stub
records have modify_operator_id='SYS-UB'; they
are deleted nightly by circjob29 when they are no
longer needed. Child record have been modified
by staff at your library, so the modify_operator_id
is the ID of one of your staff; they are deleted
nightly by circjob29 when they are no longer
needed AND their purge_date has passed.

The home_barcode_id and
home_patron_group_id are filled in for UB stub
patron records. They are values from the patron's
home database, so you can’t use them to link in other databases.

When an item is charged to a UB patron, the patron_group_id in PATRON_BARCODE is ignored. Instead, the patron group is looked up anew in UB_PATRON_GROUP_MAP using the db_id from the stub and home_patron_group_id from PATRON_BARCODE. So, if you change the UB patron group mapping, it’s possible to have some items charged under the old patron group and some under the new one. And a renewal of an item charged under the old patron group can’t be done in WV because an override is required.

Data in this table are defined in the SysAdmin client at Circulation, Patron Groups.

The demerits_applies and max_demerits fields do not appear to be used for anything.

Data in this table are defined in the SysAdmin client at Circulation, Patron Groups, Global Borrowed Item Limits tab.

If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

PATRON_GROUP_ITEM_TYPE

In this table are defined in the SysAdmin client at Circulation, Policy Definitions, Patrons tab.

The place_hold_outside_lib, place_interlib_loan_req, place_purchase_req, and place_recall_outside_lib fields are obsolete.
overdue_recall_limit number
place_call_slips character 1
place_hold_inside_lib character 1
place_recall_inside_lib character 1
place_short_loan_in_lib character 1
recall_limit number
self_shelve_limit number
title_short_loan number
total_short_loan number

PATRON_NAME_TYPE p. 20
patron_name_desc character 25
patron_name_type number

PATRON_NOTES p. 20
address_id number
modify_operator_id character 10
patron_id number
patron_note_id number
modify_date date
note character 1900
note_type number

PATRON_PHONE p. 20
When a patron record is updated by a batch patron load, all its addresses and phone numbers are deleted and re-added. Consequently, the values in address_id and patron_phone_id grow faster than the _id fields in other patron tables.

address_id number
modify_operator_id character 10
patron_phone_id number
modify_date date
phone_number character 25
phone_type number

PATRON_STATS p. 6, 20
patron_id number
patron_stat_id number
date_applied date

PATRON_STAT_CODE p. 5, 6, 20
Data in this table are defined in the SysAdmin client at System, Statistical Categories, Patron tab. If an undefined stat code comes in on a batch patron load, the code will be defined here automatically. In this case, the patron_stat_code and patron_stat_desc will be the same.

To count circulation by patron stat code for current charges, link from CIRC_TRANSACTIONS via patron_id to PATRON_STATS. To count circulation by patron stat code for completed circ transactions, link from CIRC_TRANS_ARCHIVE via circ_transaction_id to CIRC_TRANSACTION_STATS. Either way, you can then use PATRON_STAT_CODE to translate patron_stat_id.

If any of your patrons have multiple patron stat codes, be aware that their circ transactions will be counted multiple times.

patron_stat_id number
patron_stat_code character 3
patron_stat_desc character 25

PATTERN p. 25
If a record has not been modified, the modify_date is null.
create_location_id number
create_opid character 10
pattern_id number
update_location_id number
update_opid character 10
alt_chron1 number
alt_lvl1 character 20
alt_lvl1_scheme character 2
alt_lvl2 character 20
alt_lvl2_max number
alt_lvl2_num_cont number
alt_lvl2_scheme character 2
chron1 number
chron2 number
chron3 number
chron4 number
create_date date
frequency_code character 1
lvl1 character 20
lvl1_scheme character 2
lvl2 character 20
lvl2_max number
lvl2_num_cont number
lvl2_scheme character 2
lvl3 character 20
lvl3_max number
lvl3_num_cont number
lvl3_scheme character 2
lvl4 character 20
lvl4_max number
lvl4_num_cont number
lvl4_scheme character 2
This table may be used for both patron and vendor phone numbers, but see the note on VENDOR_PHONE before you use it for vendors.

PO_FUNDS p. 13
This table contains fund info for POs that have not yet been invoiced or have rolled over as an open order.

Remember that a fund_id does not uniquely identify a fund. It's the combination of fund_id and ledger_id that uniquely identifies a fund. Consequently, you need to link by both of these fields when you are linking among the FUND..., PO_FUNDS and LINE_ITEM_FUNDS tables.

PO_NOTES p. 22
The print_note field is called “Instructions to vendor” in the acq client.

PO_STATUS p. 22
po_status number
po_status_desc character 25

PO_TYPE p. 22
Data in this table are defined in the SysAdmin client at Acquisitions, PO Types.

PO_TYPE_RULES
po_type_id number
rule_id number
apl_increase number
approval character 1
blanket_order character 1
bo_increase number
mem_increase number
membership character 1
mp_increase number
multi_part character 1
single_part character 1
so_increase number
sp_increase number
standing_order character 1
sub_increase number
subscription character 1

PO_VENDOR_HISTORY
account_id number
audit_id number
po_id number
replace_opid character 10
vendor_id number
replace_date date
replace_location number

PRICE_ADJUSTMENT p. 22
The reason_id is interpreted by the ADJUST_REASON table.
The values of method are 1=Amount (Line item or PO total), 2=Per Copy, 3=Percentage.
If object_type=A, then object_id is a po_id.
If object_type=B, then object_id is a line_item_id.
If object_type=C, then object_id is an invoice_id.
If object_type=D, then object_id is an invoice_line_id.
object_id number
payment_id number
reason_id number
adjust_amount number
method number
object_type character 1
sequence number

PRIMO_AVAIL
This table is added with Voyager V6.5.4 for use with the Primo pac.

bib_id number
deleted_YN character 1
avail_hash number

**PRINT_LOCATION**
Data in this table are defined in the SysAdmin client at System, Print Locations.

print_location_id number
acq_global_printing character 1
cat_global_printing character 1
circ_global_printing character 1
default_printing character 1
media_global_printing character 1
print_location_code character 10
print_location_name character 25

**PROXY_PATRON** p. 6, 21
create_opid character 10
patron_barcode_id number
patron_barcode_id_proxy number
create_date date
create_location number
expiration_date date

**PURCHASE_ORDER** p. 13, 17, 22, 25
account_id number
approve_location_id number
approve_opid character 10
create_location_id number
create_opid character 10
po_id number
update_location_id number
update_opid character 10
vendor_id number
adjustments_subtotal number
bill_location number
cancel_interval number
claim_interval number
conversion_rate number
currency_code character 3
edi_ref number
line_item_count number
line_item_subtotal number
normal_po_number character 25
not_needed_after date
order_location number
po_approve_date date
po_create_date date
po_number character 25
po_status number
po_status_date date
po_type number
po_update_date date
prepay_conversion_rate number
rush character 1
ship_location number
ship_via character 20
total number

**RECORDCOUNT_VW**
To get an easy count of various types of records in your database, select all fields and all records from this view.

For most types of records, it’s a simple count; but for patrons, only those whose expire date has not passed are counted.

count number
recordtype character 16

**REFERENCE_TYPE** p. 10, 32
display_constant character 80
reference_type character 1
reference_type_desc character 20

**REMOTE_STORAGE_QUEUE**
This table is related to Voyager’s ARS product.

item_id number
location_id number
patron_id number
pickup_location_id number
queue_id number
item_barcode character 30
message_type character 4
sent character 1

**RENEW_TRANS... Tables**
There is a renewal_count field in CIRC_TRANSACTIONS and CIRC_TRANS_ARCHIVE. If you just need counts, use it. If you need to know when or how the renewal occurred, you need the RENEW_TRANS* tables.

RENEW_TRANSACTIONS has a record for each time an item is renewed. When the item is discharged, all of the renewal records get copied to RENEW_TRANS_ARCHIVE. If you want to count renewals, you probably want to include
records from both tables. A technique for doing this is given in the CARLI shared SQL space, http://www.carli.illinois.edu/products-services/i-share/reports/secure/sql-loc-circ#08-01b

For non-UB transactions, renew_location and renew_oper_id should contain a circ happening location and a circ operator. However, if renew_location is not a circ happening location, check the renew_date. The renewal may have come over in the conversion from your previous ILS.

For UB transactions, the values of renew_location and renew_oper_id vary depending on the Voyager version on which the renewal was done.

For UB transactions in V2001.2, if renew_location is zero, the renewal was done at the circ desk of another library. If renew_location is not zero and renew_oper_id is blank, then the renewal was done in your Web Voyager. If renew_location is not zero and renew_oper_id is SYS-UB, then the renewal was done in another library’s Web Voyager.

For UB transactions in V6.1, if renew_oper_id is OPAC or SYS-UB or null, then the renewal was done someplace other than your circ desk. In this case, renew_location is set to the item’s location.

RENEW_TRANSACTIONS
  circ_transaction_id number
  renew_oper_id character 10
  renew_date date
  renew_due_date date
  renew_location number
  renew_type character 1

RENEW_TRANS_ARCHIVE  p. 5
  circ_transaction_id number
  renew_oper_id character 10
  renew_date date
  renew_due_date date
  renew_location number
  renew_type character 1

REPORT_TYPES  p. 17, 25
Data in this table are defined in the SysAdmin client at Acquisitions, Vendor Reports.

  edi_code character 11

  report_type number
  report_type_desc character 70

REQUEST_GROUP  p. 11
Data in this table are defined in the SysAdmin client at Circulation, Request Groups.

  group_id number
  group_code character 10
  group_name character 25

REQUEST_GROUP_LOCATION  p. 11
Data in this table are defined in the SysAdmin client at Circulation, Request Groups.

  group_id number
  location_id number

REQUEST_HISTORY
For call slips that started in this database (i.e. have not been promoted), sometimes there is no REQUEST_HISTORY record and sometimes there is a REQUEST_HISTORY record that shows that it started here. I don’t know the difference between these cases, but count both if you want to count requests without counting promoted requests multiple times.

For UB requests that have been promoted to your library, this table shows you where they’ve been previously, ordered by the sequence field. There is also a row for your library.

If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

  call_slip_id number
  circ_cluster_id number
  db_key character 100
  promote_date date
  sequence number

RESERVE... Tables
Reserves circ is much like regular circ. At the time of charge, a CIRC_TRANSACTIONS record is created. At discharge, the record is moved to CIRC_TRANS_ARCHIVE and the historical_charges counter in the ITEM record is incremented. The RESERVE... tables keep track of what is on your reserve lists. There are also some
counters that allow you to see how your reserves are used, but it gets complicated.

There are 2 ways to count reserve circulation. One is by using the CIRC_TRANS_ARCHIVE table. You can do this if an item that is put on reserve is given a distinctive temp item type, or a distinctive temp location, or if your reserve desk has its own circ happening location. To do this, link from CIRC_TRANS_ARCHIVE to CIRC_POLICY_MATRIX and from there to either ITEM_TYPE, LOCATION, or CIRC_POLICY_GROUP. The advantage of this approach is that you can count reserves circulation even when items are no longer on reserve. The disadvantage is that you cannot tell what reserve list the item was on.

The second way of counting reserve circulation works only while an item is still on reserve, but you can tell which reserve list or lists the item is on. If you use this strategy, you might want to collect your statistics before you disperse your reserves at the end of the term. To do this, link from RESERVE_LIST to RESERVE_LIST_ITEMS to ITEM and use reserve_charges in ITEM. Be aware that the circulation for items on multiple reserve lists will be counted for all the lists that they are on.

More detail: When an item is on a reserve list, there's a record for it in RESERVE_LIST_ITEMS. When an item is turned on on a reserve list, the on_reserve field in ITEM is set to "Y" and an open-ended record is written in RESERVE_ITEM_HISTORY. While an item is turned on on a reserve list, the circulation count is collected in the reserve_charges field in ITEM. When an item is turned off of a reserve list, that value is copied to the reserve_charges field in RESERVE_ITEM_HISTORY and the field is zeroed out in ITEM. At the same time, the expire_date in RESERVE_ITEM_HISTORY is set. Reserve circulation is also recorded in CIRC_TRANSACTIONS and CIRC_TRANS_ARCHIVE in the same way that non-reserve circ transactions are. So, if you want to count reserve circ separately from non-reserve circ, you need the RESERVE_ITEM_HISTORY table to tell you when the item was on reserve.

Now, consider the case of an item on multiple reserve lists. It has multiple records in RESERVE_LIST_ITEMS. Because an item is turned on in the ITEM table, an item on multiple reserve lists is turned on for all reserve lists or none of them. So, if an item is on multiple reserve lists, you cannot distinguish the charges for list.

RESERVE_ITEM_HISTORY p. 8
item_id number
effect_date date
expire_date date
reserve_charges number

RESERVE_LIST p. 8
If a record has not been modified, the modify_date is null.
create_location_id number
create_opid character 10
reserve_list_id number
update_location_id number
update_opid character 10
create_date date
effect_date date
expire_date date
list_title character 40
normal_list_title character 40
reserve_item_type number
reserve_location number
reserve_location_number
update_date date

RESERVE_LIST_COURSES p. 8
course_id number
department_id number
instructor_id number
reserve_list_id number
section_id number

RESERVE_LIST_ITEMS p. 8
eitem_id number
reserve_list_id number

RESERVE_LIST_ITEMS p. 8
The RESERVE_LIST_ITEMS table tells you which items are on which reserve lists.
item_id number
reserve_list_id number

ROLLOVER_AUDIT
audit_id number
parent_id number
record_id number
run_id number
other_info character 50
record_type number
result_code number
timestamp date

**ROLLOVER_RESULT_CODES**
description character 256
result_code number

**ROLLOVER_RULES**
The fiscal_period_id in ROLLOVER_RULES can be used to link to fiscal_year_id field in LEDGER. This isn’t obvious from the names.
action_indicator character 1
create_op_id character 10
fiscal_period_id number
new_fiscal_period_id number
rule_id number
update_op_id character 10
create_date date
initialize_type character 1
normal_rule_name character 25
rule_name character 25
update_date date

**ROUTING_LIST** p. 26
create_location_id number
create_opid character 10
routing_list_id number
update_location_id number
update_opid character 10
create_date date
name character 45
normal_name character 45
note character 256
print_note character 1
update_date date

**ROUTING_LIST_MEMBERS** p. 26
member_id number
routing_list_id number
add_date date
member_type character 1
rank number

**SAVED_RECORDS_RESULTS**
CARLI has not allowed access to this table by library staff because of patron confidentiality concerns.
bib_id number
db_id number

**SAVED_SEARCHES**
The starred field in this table is in UTF-8.

CARLI has not allowed access to this table by library staff because of patron confidentiality concerns.

**SDI_INTERVALS**
sdi_interval_id number
sdi_interval_code character 10
sdi_interval_days number

**SEARCHFIELDS**
Data in this table are defined in the SysAdmin client at Search, Indexes - Holding Keyword Definitions and Indexes - Holding Keyword Definitions.

This table holds the definition of keyword search keys. The name of the search key is in searchcode. The MARC fields and subfields that are indexed with this search key are in fieldcode. The searchcode field is further defined in the SEARCHPARAM table.
fieldcode character 4
searchcode character 4

**SEARCHPARAM** p. 14
Data in this table are defined in the SysAdmin client at Search, Indexes - Composite Definitions and in Indexes- Headings and Left-Anchored Definitions and in Indexes - Holding Keyword Definitions and in Indexes - Holding Keyword Definitions.
This table is cryptic, but it stores many of the indexing decisions that govern searching in Voyager.

The searchcode field can be used to link to index_code in the BIB_INDEX and AUTH_INDEX tables.

The indexrules field, if you can figure it out, tells how each index in constructed. We don’t completely understand this field, but here’s what we do know: IX=A for authority indexes, B for bib indexes, K for keyword indexes, Q for MFHD indexes, S for special subject indexes, T for special title indexes, U for call number indexes. AL= is a MARC field. SR= is used with a single, repeatable field and indicates that each occurrence should generate an index entry; S+= lists subfields that should be included in the index. S= lists subfields that should be excluded from the index. NM= is the normalization rules. NF tells the location of a non-filing indicator. HL indicates a hard limit, i.e., a limit that is always in effect for this search key.

The ordering field does not appear to be used for anything.

The acqcount, catcount, circcount, and mediacount fields are incremented by Voyager during searches in the staff clients. They never display in the clients, but they make the most frequently used searches appear first in the search window. Similarly, the opaccount field makes the most frequently used keyword searches appear first on the WebVoyage advanced search screen.

SERIALS_VW
This view has a number of quirks and it is not efficient. Consider using the tables directly instead.

If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

The SERIAL_CLAIM Tables
To uniquely identify a serial issue, you need both issue_id and component_id. To identify a copy, you need copy_id as well. To identify a claim, you need claim_thread and claim_id too.

The claim_count tells you which claim this is (first, second, etc.). For the most recent claim, claim_status=1; otherwise claim_status=0.

The claim_type may be interpreted using the CLAIM_TYPES table.

The claim_date is the date when the order should be claimed. If it has been overridden, the new date is in override_claim_date.

SERIALCLAIM p. 25
claim_id number
component_id number
copy_id number
issue_id number
location_id number
op_id character 10
vendor_id number
claim_count number
claim_date date
claim_status number
claim_thread number
claim_type number
edi_ref number
note character 256
override_claim_date date

SERIAL_CLAIM_ARCHIVE
claim_id number
component_id number
copy_id number
issue_id number
location_id number
op_id character 10
vendor_id number
archive_date date
claim_count number
claim_date date
claim_status number
claim_thread number
claim_type number
edi_ref number
note character 256
override_claim_date date

SERIAL_ISSUES
To uniquely identify a serial issue, you need both issue_id and component_id.

With V7.0, data that used to be in the UNPREDICTABLE_ISSUES table is moved here.

SERIAL_SUPPLIER_REPORT p. 25
audit_id number
claim_id number
action_date date
action_quantity number
edi_ref number
note character 512
report_date date
report_type number

SHORT_LOAN... Tables
When an item is scheduled for a short loan, a SHORT_LOAN record is created and an ITEM_STATUS record is created. When the item is charged to the patron, the SHORT_LOAN record is archived, short_loan_charges in ITEM is incremented, and all circ activity is recorded as it is for any other charge.

SHORT_LOAN p. 28
If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

SHORT_LOAN_ARCHIVE
If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.

When a short loan is charged, the record is archived, the status changes, but the status_date does not change.
bib_id number
create_opid character 10
item_id number
mfhd_id number
patron_group_id number
patron_id number
short_loan_id number
short_loan_status_id number
update_opid character 10
create_date date
create_location number
end_time date
note character 100
pickup_location number
start_time date
status_date date
update_date date
update_location number

SHORT_LOAN_STATS
If you get the message, “Type mismatch in expression”, when you use this table, see Appendix A for a solution.
  patron_stat_id number
  short_loan_id number

SHORT_LOAN_STATUS  p. 28
  short_loan_status_id number
  short_loan_status_desc character 4

SIMUL_MERGE_PROFILE
Data in this table are defined in the SysAdmin client at Search, Simultaneous Search De-Duplicating.

The pseudo_relevance field does not appear to be used for anything.

bib_field1 character 30
bib_field2 character 30
bib_field3 character 30
citation_field1 character 30
citation_field2 character 30
citation_field3 character 30
pseudo_relevance character 1

SNV_FIELDS
This table is new with V7.0. It is related to the validation of standard numbers in the bib 020, 022, and 024 fields.
  snv_field_id number
  field character 3
  ind1 character 1
  ind2 character 1
  norm_rules character 10
  record_type character 1
  snv_validation_type character 10
  subfield character 1

SORT_GROUP
Data in this table are defined in the SysAdmin client at OPAC Configuration, Holding Sort Groups, Sort Groups tab.

 sequence_number number
  sort_group_code character 8
  sort_group_default character 1
  sort_group_id number
  sort_group_name character 40

SORT_GROUP_LOCATION
Data in this table are defined in the SysAdmin client at OPAC Configuration, Holding Sort Groups, Locations tab.

  location_id number
  sort_group_id number
  sequence_number number

STANDARD_NUMBER_VALIDATION
This table is new with V7.0. It is related to the validation of standard numbers in the bib 020, 022, and 024 fields.

  snv_field_id number
  snv_code character 7
  snv_name character 20

SUBDIVISION  p. 10, 32
The starred fields in this table are in UTF-8.

If a record has not been modified, the modify_date is null.

subdiv_id number
authorized character 1
create_date date
*display_subdiv character 300
heading_type character 1
*normal_subdiv character 300
subdiv_type character 1
update_date date

SUBDIVISION_TYPE  p. 10, 32
The starred fields in this table are in UTF-8.

subdiv_type character 1
*subdiv_type_desc character 50

**SUBSCRIPTION** p. 25
line_item_id number
subscription_id number
auto_renewal character 1
length_type number
normal_sici character 45
normal_upc character 30
note character 256
renewal_date date
sici character 45
start_date date
subscription_length number
upc character 30

**SUDOCCLASS_VW**
If you want to sort a report (not a query, a report) by class, you will have to use the Left function to truncate it to less than 255 characters.

mfhd_id number
class character 300
longclass character 12

**SUPPRESS_SYSTEMCLAIMS** p. 25
component_id number
issue_id number

circ_transaction_id number
db_id number
patron_id number
discharge_date date
due_date date
recall_date date

**UB_CHARGE_ARCHIVE**
circ_transaction_id number
db_id number
patron_id number
discharge_date date
due_date date
recall_date date

**UB_FINE_FEE** p. 19, 29
The value of fine_fee_total is incorrect about 10% of the time, so take this table with a grain of salt. Here’s how it’s supposed to work: When one of your patrons owes a fine to another CARLI I-Share library, the fine amount is recorded in this table. When your patron pays the fine, the fine_fee_total goes to zero, but the record is not deleted. To find out which library is owed the fine, use db_id to link to VOYAGER_DATABASES.

Patron blocks are implemented using total_fees_due_ub in PATRON, not the values in this table.

db_id number
patron_id number
demerits_total number
fine_fee_total number
update_date date

**UB_HOLD** p. 12, 29
When an item is lent through UB, while it is on the hold shelf at another library, there’s a UB_HOLD record in the item’s home database. Not sure what this tells you, but there it is.

There may be multiple records with the same pickup_db_id and hold_recall_id if there are multiple items as part of the hold.

hold_recall_id number
item_id number
patron_id number
pickup_db_id number
**UB_PATRON_GROUP_MAP** p. 29
Data in this table are defined in the SysAdmin client at Circulation, Patron Group Mapping.

This table is not very useful for Access reports because it requires data from the databases of other libraries. Patron_group_id_mapped is the patron_group_id from the database of the library indicated by db_id. Patron_group_id is the patron group in your database.

When patron_group_id_mapped=0, patron_group_id is the default mapping for patrons from the library specified in the db_id.

- circ_cluster_id number
- db_id number
- patron_group_id number
- patron_group_id_mapped number
- remote_circ_cluster_id number
- manual_map character 1

**UB_PATRON_RECORD**
This table indicates when one of your patrons has a stub record in another database.

- circ_cluster_id number
- db_id number
- patron_id number
- patron_stub_id number
- create_date date
- update_date date

**UB_PG_HOME_POLICY** p. 29
Data in this table are defined in the SysAdmin client at Circulation, UB Policy Definitions.

There is a record in this table if UB eligible is checked now or if it was checked in the past.

The *lclblock fields were added with V6.5.1 in support of aggregate blocking functionality.

- patron_group_id number
- claim_return_limit number
- claim_return_limit_lclblock character 1
- demerits_limit number
- demerits_limit_lclblock character 1
- fees_applies character 1
- item_limit number
- item_limit_lclblock character 1
- lost_limit number
- lost_limit_lclblock character 1
- max_claim_return_limit character 1
- max_demerits_limit character 1
- max_item_limit character 1
- max_lost_limit character 1
- max_outstanding_balance number
- max_overdue_limit character 1
- max_overdue_recall_limit character 1
- max_self_shelve_limit character 1
- max_ub_requests character 1
- outstanding_balance_lclblock character 1
- overdue_limit number
- overdue_limit_lclblock character 1
- overdue_recall_limit number
- overdue_recall_lclblock character 1
- self_shelve_limit number
- self_shelve_limit_lclblock character 1
- ub_eligible character 1
- ub_request_limit number
- ub_requests_limit_lclblock character 1

**UB_REQUEST... Tables**
When one of your patrons places a UB request, a record is written in the UB_REQUEST table in your database. Also, when one of your patrons using the Universal Catalog’s Web Voyage made a request of your library (this capability was lost with V6.1), a record was written in this table. In these records, pickup_db_id and holding_db_id both equal -1. However, if your patron places a request in your database, and you no-fill it, and the request is then promoted, there will be no UB_REQUEST record in your database.

Generally, a UB_REQUEST record corresponds to a CALL_SLIP record for one of your patrons in another database, but they are archived at different times. Each time the request is promoted to another library, the old UB_REQUEST is archived and a new UB_REQUEST record is added. When the item is finally charged to the patron, the final UB_REQUEST record will be moved to the UB_REQUEST_ARCHIVE in your database.

If you want to count UB requests made by your patrons, you’ll be pretty close if you count all but the ones with request_status=8 (promoted). Circjob 43 sometimes creates a second UB_REQUEST record for the same call slip, so your counts will be about 10% high. If you want to be really accurate, count the distinct call_slip_id’s.
The db_id field is the database whose item your patron is requesting. The pickup_db_id is where your patron wants to pick up the item. Both can be interpreted using the VOYAGER_DATABASES table, except that your own database has a db_id of zero.

The request_status field can be interpreted using the UB_REQUEST_STATUS table, but note that circjob43 changes the status from 2=In Transit to 5=Available for Pickup before the item arrives at the pickup library.

For promoted requests, date_requested is the promote date, not the request date.

**UB_REQUEST** p. 29, 30
- call_slip_id number
- db_id number
- holding_item_id number
- patron_id number
- pickup_db_id number
- date_requested date
- not_needed_after number
- request_status character 25
- status_date date

**UB_REQUEST_ARCHIVE**
- call_slip_id number
- db_id number
- holding_item_id number
- patron_id number
- pickup_db_id number
- date_requested date
- not_needed_after number
- request_status character 25
- status_date date

**UB_REQUEST_STATUS** p. 30
- status_desc_ub character 25
- status_type_ub number

**UB_ROUTING and UB_ROUTING_ARCHIVE**
As UB items are routed from location to location, records are written in these tables at the “from” and “to” libraries on each leg of the journey. The pairs of records have the same value in ub_routing_id; the value seems to be the max of the next ub_routing_id in the 2 databases. While an item is en route, there is a record in UB_ROUTING. When an item is received at its destination, db_id_received is filled in and the record is moved to UB_ROUTING_ARCHIVE.

All of the db_id* fields in these tables can be translated using the VOYAGER_DATABASES table, except the value 0 (zero) indicating your local database.

Normally, if db_id_patron=0, this is your patron, and patron_id_ub links to patron_id in your patron table. And if db_id_patron is not zero, this is not your patron, but you have a stub patron record which you can locate by linking patron_id_ub to patron_id_ub in your patron table. However, this field is incorrect on occasion, so flexible.

**UB_ROUTING** p. 12, 29, 30
The db_id_received and received_date fields are never filled in in this table because, once the item is received, the UB_ROUTING record moves to UB_ROUTING_ARCHIVE.
- db_id_from number
- db_id_item number
- db_id_patron number
- db_id_received number
- db_id_to number
- item_id_ub number
- location_id_to number
- patron_id_ub number
- ub_routing_id number
- received_date date
- shipped_date date

**UB_ROUTING_ARCHIVE**
The db_id_received field is wrong about 1/3 of the time. If it says that the item was received back at the library that sent it, it was probably received at the right place.
- db_id_from number
- db_id_item number
- db_id_patron number
- db_id_received number
- db_id_to number
- item_id_ub number
- location_id_to number
- patron_id_ub number
- ub_routing_id number
- received_date date
- shipped_date date

**UDCCLASS_VW**
mfhd_id number
class character 6

UNPREDICTABLE_ISSUES p. 25, 26
This table was dropped with Voyager V7.0 and
the data are moved to SERIAL_ISSUES.

component_id number
issue_id number
eumchron character 256
expected_date date
receipt_date date
received number

VENDOR p. 9, 17, 22, 25, 39
create_opid character 10
federal_tax_id character 10
institution_id character 25
update_opid character 10
vendor_id number
cancel_interval number
claim_count number
claim_interval number
create_date date
default_currency character 3
normal_vendor_code character 10
normal_vendor_name character 60
normal_vendor_type character 2
ship_via character 20
update_date date
vendor_code character 10
vendor_name character 60
vendor_type character 2

VENDORINVOICE_VW
institution_fund_id character 50
institution_id character 25
invoice_id number
bill_to_location character 25
bill_to_location_code character 10
currency_code character 3
currency_name character 35
expend_pending number
expenditures number
fiscal_period_end date
fiscal_period_name character 25
fiscal_period_start date
fund_name character 25
invoice_date date
invoice_number character 25
invoice_status character 25
invoice_status_date date
ledger_name character 40

policy_name character 40
vendor_code character 10
vendor_name character 60
vendor_type character 40
voucher_number character 25

VENDORORDER_VW
institution_id character 25
mfhd_id number
currency_name character 25
invoice_status character 25
line_price number
line_status_date date
order_location character 25
order_location_code character 10
po_line_status character 25
po_number character 25
po_status character 25
po_status_date date
po_type character 25
quantity number
total number
unit_price number
vendor_code character 10
vendor_name character 60
vendor_type character 40

VENDOR_ACCOUNT p. 22, 39
account_id number
vendor_id number
account_name character 25
account_number character 25
account_status number
default_discount number
default_po_type number
deposit character 1
status_date date

VENDOR_ADDRESS p. 39
address_id number
modify_operator_id character 10
vendor_id number
address_line1 character 50
address_line2 character 40
address_line3 character 40
address_line4 character 40
address_line5 character 40
city character 30
claim_address character 1
contact_name character 40
contact_title character 40
country character 20
email_address character 1
modify_date date
order_address character 1
other_address character 1
payment_address character 1
return_address character 1
state_province character 7
std_address_number character 8
zip_postal character 10

VENDOR_BANK_INFO p. 39
modify_operator_id character 10
vendor_id number
account_number character 25
address_line1 character 50
address_line2 character 40
address_line3 character 40
address_line4 character 40
address_line5 character 40
bank_name character 60
city character 30
country character 20
city character 25
fax character 25
modify_date date
phone character 25
state_province character 7
tax_id_number character 11
tax_id_number character 11
transit_number character 25
zip_postal character 10

VENDOR_NOTE p. 39
vendor_id number
note character 1900

VENDOR_PHONE p. 39
The values of phone_type are 0=primary,
1=mobile, 2=fax, 3=other. These are one less than
the values in the PHONE_TYPE table. But you
can effect a link between VENDOR_PHONE and
PHONE_TYPE by this devious means:

SELECT VENDOR_PHONE.PHONE_NUMBER,
PHONE_TYPEPHONE_DESC
FROM VENDOR_PHONE, PHONE_TYPE
WHERE (((Val([phone_type].[phone_type]))=
Val([vendor_phone].[phone_type])+1));

database_name character 50

VENDOR_TYPES p. 39
Data in this table are defined in the SysAdmin
client at Acquisitions, Vendor Types.

vendor_type character 2
vendor_type_desc character 40

VENDOR_TYPE_DEFAULTS p. 39
acq_policy_id number
cancel_interval number
claim_count number
claim_interval number
discount number
order_type number
ship_via character 20
vendor_type character 2

VERSIONS
This tables lists the version of each Voyager
module that is in place. It provides a way to
determine which patch sets have been applied.

module character 20
syncpoint number
version character 30

VOYAGER_DATABASES p. 3, 30
Data in this table are defined in the SysAdmin
client at Search, Database Definitions, Definitions
tab.

This table can be used to interpret database IDs
that occur in all the other tables, with one
exception: In VOYAGER_DATABASES, your
local database has db_id=1; in all the other tables,
your local database is indicated by a value of zero
(or sometimes a mix of zero and null).

The implementor, opacsuppress, retrievaltimeout,
and searchtimeout fields do not appear to be used
for anything.

The public_highwater and staff_highwater fields
are filled in by Voyager as it runs and are not
displayed anywhere in the clients.

db_id number
dup_profile_id number
action character 10
char_set_id number
connecttimeout number
database_name character 50
WOPAC_PID_PATRON_KEYS
This table is used for Voyager’s External Patron Authentication functionality.

patron_key character 30
pid character 80

Z3950_ATTRIBUTES
Data in this table are defined in the SysAdmin client at Search, Database Definitions, Attributes tab.

db_id number
attrib_desc character 50
attributes character 40
boolean_enabled character 1
db_code character 8
lh_truncation character 1
rh_truncation character 1
searchcode character 4
Appendix A: The “Type mismatch in expression” message

The “Type mismatch in expression” message comes up occasionally when you run a new Access query or a query that you have just changed. It means that the fields that you use in one of your links are of different types. In other words, one field is a number and the other is a text string. It’s not your fault. This Data Dictionary leads you to believe that they are both numbers, but that’s not quite true.

Detail for techies: The two fields are indeed defined as numbers in Voyager’s Oracle database. There is a slight difference in the way that they are defined, however, that makes Access treat them differently. It’s such a small difference that it doesn’t matter to Voyager. One field is explicitly defined as an integer and the other is implicitly an integer. Most numerical fields in Voyager are explicitly defined as integers. Access knows that Oracle can handle larger integers than Access can. To protect itself from an integer value that might be too large for it to handle, Access treats the field as if it were a text string. But if Oracle defines a numerical string as an integer implicitly, Access treats it as a number.

Most fields that look like numbers are treated by Access as if they were text strings. But there are a few exceptions, and these are the ones that trigger the “Type mismatch” error message. Here is a list of the fields that Access treats as numbers:

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALL_SLIP_STATS</td>
<td>CALL_SLIP_ID</td>
</tr>
<tr>
<td>CALL_SLIP_STATS</td>
<td>PATRON_STAT_ID</td>
</tr>
<tr>
<td>CIRC_ALERT_CONDITIONS</td>
<td>LOC_ID</td>
</tr>
<tr>
<td>ENUM CHRON_TYPES_VW</td>
<td>CHRON_TYPE_ID</td>
</tr>
<tr>
<td>ENUM CHRON_TYPES_VW</td>
<td>ENUMERATION_TYPE_ID</td>
</tr>
<tr>
<td>HOLD_RECALL</td>
<td>PATRON_GROUP_ID</td>
</tr>
<tr>
<td>HOLD_RECALL_ARCHIVE</td>
<td>PATRON_GROUP_ID</td>
</tr>
<tr>
<td>HOLD_RECALL_ARCHIVE</td>
<td>PATRON_ID</td>
</tr>
<tr>
<td>PATRON_GROUP_ITEM_TYPE</td>
<td>ITEM_TYPE_ID</td>
</tr>
<tr>
<td>PATRON_GROUP_ITEM_TYPE</td>
<td>PATRON_GROUP_ID</td>
</tr>
<tr>
<td>REQUEST_HISTORY</td>
<td>CALL_SLIP_ID</td>
</tr>
<tr>
<td>REQUEST_HISTORY</td>
<td>CIRC_CLUSTER_ID</td>
</tr>
<tr>
<td>SERIALS_VW</td>
<td>ISSUE_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>BIB_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>CREATE_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>ITEM_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>MFHD_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>PATRON_GROUP_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>PATRON_ID</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>PICKUP_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN</td>
<td>UPDATE_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>BIB_ID</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>CREATE_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>ITEM_ID</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>MFHD_ID</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>PATRON_GROUP_ID</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>PATRON_ID</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>PICKUP_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN_ARCHIVE</td>
<td>UPDATE_LOCATION</td>
</tr>
<tr>
<td>SHORT_LOAN_STATS</td>
<td>PATRON_STAT_ID</td>
</tr>
</tbody>
</table>
So the problem that you need to solve involves a link between one of these fields and a like-named field in another table. Here’s what to do:

1) In the design pane, delete the link in question. Do this by right-clicking near the middle of the link and clicking on Delete.

2) Is the numeric field (i.e., the one listed above) in your list of fields? No? Then add it. If you don’t want it in your query results, un-check the Show box.

3) Now you need to add a criterion underneath the numeric field. The criterion will use the table name and field name of the text field. This is the field that was part of the link. The syntax for the criterion is:

   \[ \text{Val ( \{tablename\}.\{fieldname\} )} \]

For example, here is a query that counts hold and recalls by patron group:

But when I try to run it...
So I right-click near the middle of the link and select Delete:

The numeric field (i.e., the one list in the table above) patron_group_id in the hold_recall_archive table. So I add that field to my list of fields:
It was linked to patron_group_id in the patron_group table, so I’ll put that field and table name in the criterion:

I don’t want the patron_group_id to display, so I’ll un-check the Show box. Since this is a Totals query, I’m also going to change the Group By to Where on this field.

Voila!
Appendix B: What's in the LINK and LINK_TEXT fields of ELINK_INDEX?

The values in the LINK and LINK_TEXT fields in the ELINK_INDEX table are derived from the 856 field of the corresponding bib, MFHD, or authority.

The value of the LINK field:

If there is a $u
then if $u$ has a valid prefix (e.g. http, telnet, ftp, file, etc)
    then LINK = $u
else LINK = blank
else if there is a $g
then LINK = $g
else if $d$ and $f$ and $o$ are all present
    then if there is a $2$
        then LINK = blank
    else if $o$ = 'dos'
        then LINK = $d$, a backslash character, $f$
    else LINK = $d$, a slash character, $f$
else LINK = blank

The value of the LINK_TEXT field:
There are 4 cases, depending on whether $3$ and $z$ are present.

1. If there is a $z$ and no $3$
   then LINK_TEXT = $z$

2. If there is a $z$ and $3$
   then if there is $u$ or $g$
       then LINK_TEXT = $3$ followed by $z$
   else LINK_TEXT = $z$

3. If there is a $3$ and no $z$
   then if there is $u$ or $g$
       then LINK_TEXT = $3$
   else if $d$ and $f$ and $o$ are all present
       then LINK_TEXT = $3$
   else if $d$ and $f$ are both present
       then LINK_TEXT = blank
   else LINK_TEXT = all subfields of the 856 with subfield codes

4. If neither $3$ nor $z$ is present and there’s something in the LINK field
   then LINK_TEXT = LINK
   else LINK_TEXT = all subfields of the 856 with subfield codes