

ExLibris®  
Part of Clarivate

IGOLU  
International Group of Ex Libris Users

# Aleph Working Group Open Meeting and Aleph Product Update

Yoel Kortick – Senior Librarian



# Agenda



Aleph roadmap

---



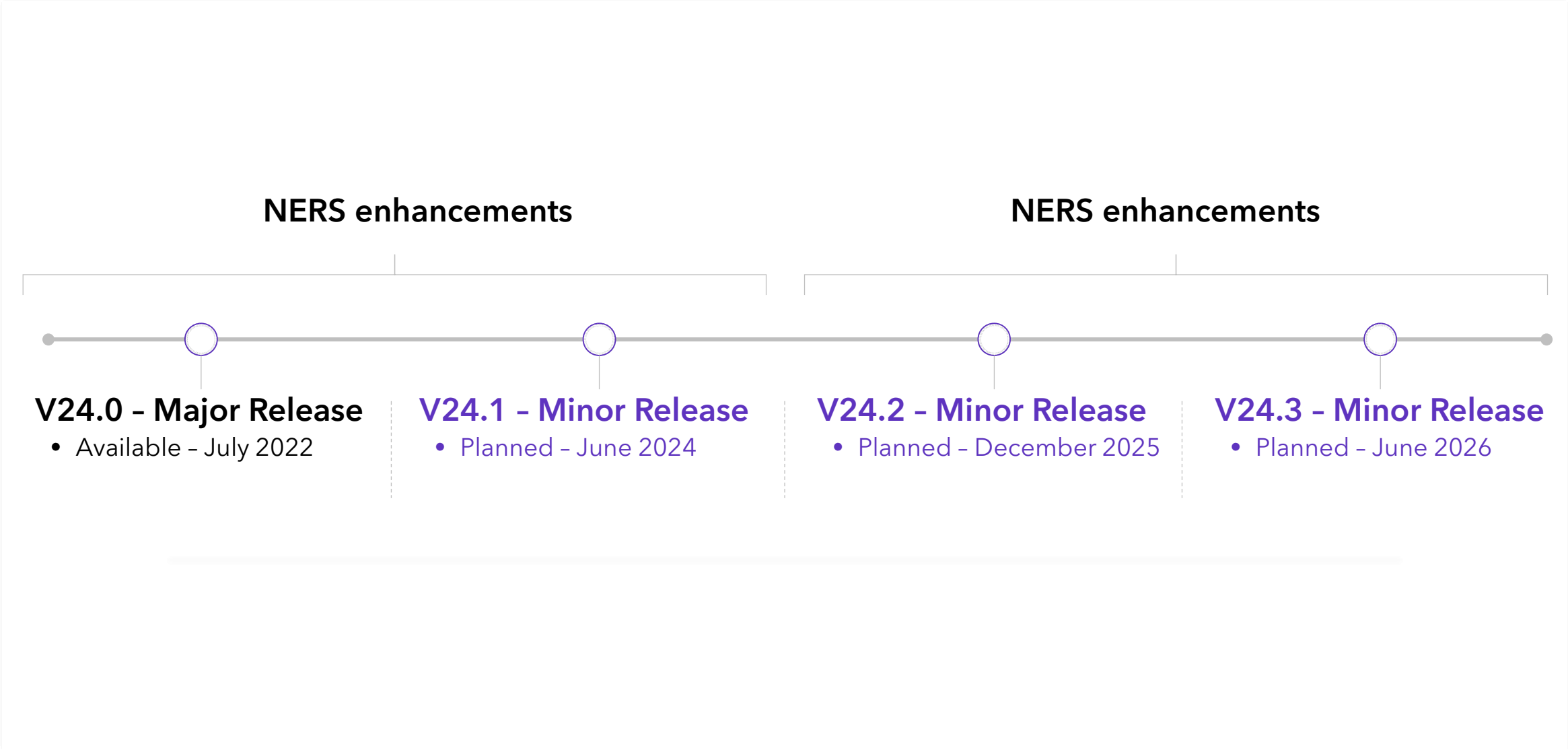
Aleph 24.0 highlights

---



Plans for 24.1 and beyond

# Aleph Roadmap - Updated



# Aleph V24.0



Prompt for confirmation before deleting triggers - V24.0



Validation checks across fields  
- create dependencies for subfields across two or more fields - V24.0



Fiscal Year Purge - remove all vendors without any remaining orders or invoices



# Aleph V24.0 - Release July 2022

# Prompt For Confirmation Before Deleting Triggers

---



Current Aleph GUI 'Trigger List' interface doesn't provide an alert when user activate the 'Delete' trigger action

---



Once 'Delete' is pressed, the trigger is immediately deleted without any recovery option

---



Having a confirmation message before deletion will attract the user's attention and avoid redundant deletion

# Prompt For Confirmation Before Deleting Triggers

## Solution Description



The user's flows for trigger deletion will be enhanced with an additional confirmation message before the delete is executed



This change will be implemented for all GUI users without any special customization or setup

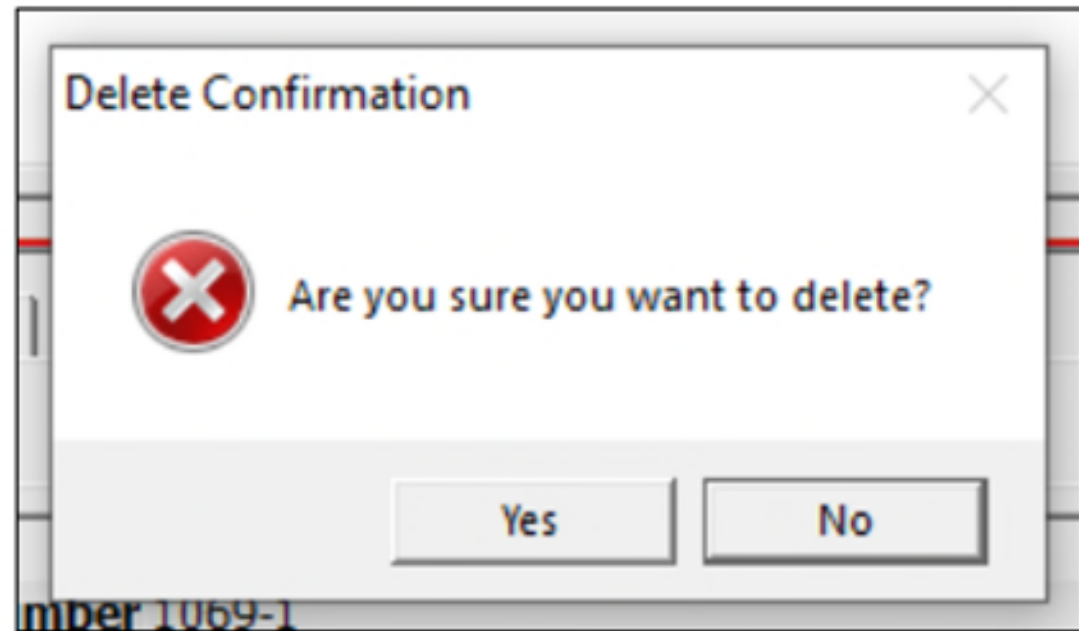


It will cover the three on-line triggers that offer the 'Delete'

# Prompt For Confirmation Before Deleting Triggers



A new pop-up message will appear when the 'Delete' action will be activated by the user via GUI-Triggers List:





# Prompt For Confirmation Before Deleting Triggers

---



The default selected option will be 'No'

---



The user will have to explicitly press the 'Yes' option in order to submit the deletion

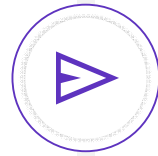
---



The delete action will be executed AFTER the user's confirmation

# Prompt For Confirmation Before Deleting Triggers

## Triggers to be covered



GUI Cataloging/Acquisitions-Serials/Circulation - Trigger List (for a specific record)

---



GUI Cataloging - Trigger List (for all records)

---



GUI Cataloging - Edit Actions menu - Record's Triggers - Trigger List (for a specific record)

# Validation Checks Across Fields



Current Aleph check\_doc validation mechanism, provides the functionality of checking dependencies for subfields within the same field



There is no way to define dependencies across fields check\_doc\_line and check\_doc\_doc



Improving the check\_doc functionality to include validation checks across fields, will assist the catalogers when editing a record and improve the quality of the records

# Validation Checks Across Fields



A new check\_doc program will be added to create dependencies across fields:  
check\_doc\_across\_fields



This will allow checking a BIB/AUTH record based on data cataloged in subfields of different fields (within the same record)



Library will configure the relevant dependencies



New messages (errors) will be added to reflect the detected inconsistencies



Will impact cataloging and related services that use 'check doc'





# Validation Checks Across Fields

## Setting-up the new table: check\_doc\_across\_fields



A new \$data\_tab/check\_doc\_across\_fields table will define the rules of checking content of subfields across fields



The table will enable to define the rules for checking content dependencies among subfields of different fields



Following is a sample table in the demo BIB library:  
./usm01/tab/check\_doc\_across



# Validation Checks Across Fields



The rules may be defined per record format (col.1). For example, books (BK) might have different logic than serials (SE)

The code XX can be set for all formats



Col. 2 thru 4 set the 1st tag+indicator+subfield+content that should be checked against the 2nd set



Col. 6 thru 8 set the 2nd tag+indicator+subfield+content that should be checked against the 1st set



Col. 5 will set the type of dependency:

- Y = the defined content should be presented in the other set
- N = the defined content cannot be presented in the other set



# Validation Checks Across Fields

## The New Block/Warning Message



A new error text will be added to `./error_<lng>/check_doc` file (file already exists)



**Error number:** 0232 (hard-code; ' cannot be changed by the library)



**Error text:** Error found upon dependency check between the content of the subfields "\$1" and "\$2"



\$1 - The place holder for the tag: (field+indicator+subfield) of the 1st field (col. 2 and 3 of `check_doc_across_fields` table)



\$2 - The place holder for the tag: (field+indicator+subfield) of the 2nd field (col. 6 and 7 of `check_doc_across_fields` table)

# Validation Checks Across Fields

```
1009 L 901 - Missing system counter for library $1.
1010 L 901 - The counter $2 for library $1 is already attributed to doc number $3. Check system
1011 L 901 - You are not allowed to add more than 10 inventory numbers at a time.
!
! check_doc_across_fields
0232 L Error found upon dependency check between the content of the subfields "$1" and "$2"
!
! check_doc_call_no_090 )The new error #0232 in ./error_<lng>/check_doc
1100 L ($1$a): Call number prefix $2 is a new entry.
1101 L ($1$a): Call number prefix $2 is a new entry, with manual input of the sequence=$3 !
1102 L ($1$a): Call number prefix $2 is a old entry, with manual input of the sequence=$3 !
1105 L ($1$a): Error, Call number prefix $2 is an old entry, but maximum for sequence reached;
ling !
```

# Validation Checks Across Fields

## Setting the New Check as a Block or Warning



As in any other record checks, the library should define the new check as a Warning (T) or Mandatory (M)



T = error which writes a cataloging trigger record but allows database update



M = error which does not allow database update



This can be done by adding the new entry to \$data\_tab/check\_doc\_mandatory (table already exists)

# Validation Checks Across Fields

Clipboard image tools snapshots Colors

SE System No. 500 Waseda hōgaku = () Year: 1922

Barcode

[R] Edit Records  
USM01-500 (NEW010.MRC)  
[I] Import Records  
[T] Triggers

USM01 - 500  
OBJECTS (No record available)  
USM60  
HOL - 50061 STA () [50061]  
USM50 - 238710  
USM51 (No record available)

**Leader** LDR 01195cas^^2200361^a^4500  
**Control No.** 001 000000500-2  
**Date and Time** 005 20031024151229.0  
**Fixed Data** 008 830520c19229999ja^qx^p^^^g^^^0uu1^0jp  
**LC Control No.** 010 a sn^89026301^/V/r93  
**ISSN**  
**System No.**  
**Catal. Source**  
**Language Code**  
**Authent. Code**  
**Geog. Area Code**  
**Abbrev. Title**  
**Key Title**  
**Main Title**  
**Varying Title**

Record Check Warnings/Errors

Doc Validation Error		Override
?	008/30: obsolete value u.	
?	008/31: obsolete value u.	
?	008/32: obsolete value 1.	
?	First indicator in tag "035" is not valid.	
?	First indicator in tag "222" is not valid.	
?	Error found upon dependency check between the content of the subfields "949 a" and "960 d"	

1. Messages | 2. Tag Information

Cancel  
Help

The new error in GUI-Catalog, when a record is saved



# Infrastructure Changes in V24.0



# RedHat8



# Aleph V24.1 - Planned for June 2024

# Aleph V24.1

---



Fiscal Year Purge - remove all vendors without any remaining orders or invoices



# Plans for V25 and Beyond

# Plans for V25 and Future Versions



## Items for consideration:



User group enhancements (NERS)

---

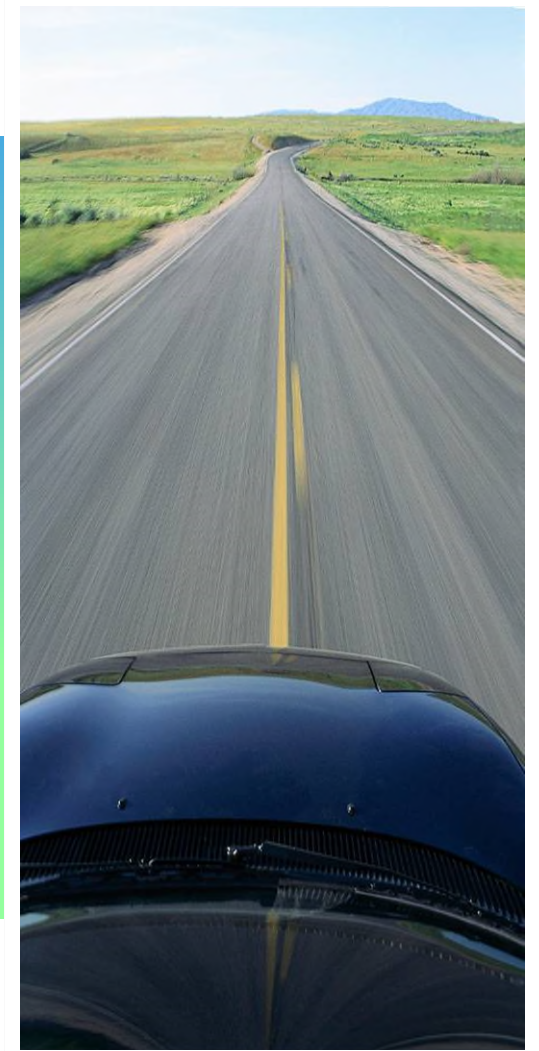


Extending support for protocols and standards such as NCIP, SIP2, RFID, EDI

---



Infrastructure SW/HW certifications







# Thank You!

[Yoel.Kortick@clarivate.com](mailto:Yoel.Kortick@clarivate.com)

© 2023 Clarivate

Clarivate and its logo, as well as all other trademarks used herein are trademarks of their respective owners and used under license.

© 2023 Clarivate