

---

## item-04: "Not in scope"

- **Article Type:** General
  - **Product:** Aleph
  - **Product Version:** 20
  - **Relevant for Installation Type:** Dedicated-Direct; Direct; Local; Total Care
- 

Running the item-04 shelf reading report against a list of barcodes uploaded to the server, it put some of the items onto the "missing" report, and lists everything not missing from the shelves as being "not in scope". Why is that?

Resolution:

For an item to be "in scope" it must have the proper class code (such as, "0", for LC call numbers) and the call number must be in the "From call no" / "To call no" range. Note that the "To call no" is to, but not including, the call number specified.

If all "P" call numbers are desired, then the range should be From Call No "P", To Call No "PZZZ".

In this case the item connected to the first barcode in the input file has z30\_call\_no\_type '0' while the second, third, etc., are for z30\_call\_no\_type '8'. The program is looking for items which are in the Call No. Type specified in the submission, that is, Call No. Type '0'.

SQL shows that there are 10,000 z30\_call\_no\_type '0'; 1189 z30\_call\_no\_type '8's; 672 z30\_call\_no\_type <blank>; etc., in the specified sublibrary/Collection.

Since the input file contains barcodes with a variety of Call No. Type's, the Call No. Type needs to be ignored in this processing.

Since, as things stand, the z30\_call\_no\_key (which is what is used for sorting) also contains the z30\_call\_no\_type, e.g.:

```
0 ac 1 a4 v 511  
8 $$$hac 1 g7
```

This will, first, require generating the z30\_call\_no\_key's without the z30\_call\_no\_type....

1. Remove all the "... add\_call\_no\_type" lines from abc50 tab\_filing\_call\_no
2. Copy the type "0" lines to a type "8", so that the type "8" call numbers are sorted the same as the type "0"
3. Run p\_item\_06 to rebuild the z30\_call\_no\_key's
4. Submit p\_item\_04 with the default value of "NO\_USE" for the Call No. Type.

---

## Additional Information

In another Case tab\_filing\_call\_no had the following:

```
!1 2      3      4
```

```
!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
0 L del_subfield
0 L to_blank      &()-[]<>?,./:
0 L lc_call_no
0 L add_call_no_type
0 L pack_spaces
!*
<snip>
!
# L add_call_no_type
# L to_lower
# L pack_spaces
```

Adding the following lines for type 8 (which it then used instead of the default "#" entry) and re-running the item-06 Service fixed the problem:

```
!*
8 L del_subfield
8 L to_blank      &()-[]<>?,./:
8 L add_call_no_type
8 L pack_spaces
```

**Category:** Cataloging

---

- **Article last edited:** 9/6/2015