
Rush recall not given priority to existing regular recall

- **Product:** Aleph
 - **Product Version:** 20, 21, 22, 23
 - **Relevant for Installation Type:** Dedicated-Direct,Direct,Local,Total Care
-

Description:

When a regular recall has been placed on an item (and processed by cir-13) and then a rush recall (z37_recall_type 02, z37_priority 00) is placed on the item, the rush recall is *not* given priority over the regular recall. Why is that?

Resolution:

The effect of priority 00 is to place the request at the top of the queue, but when the first recall is already in process -- and the z36_recall_xxxx fields have been updated -- that recall is, in a sense, no longer in the queue. If there are two regular recalls ahead of it, the priority 00 request would be placed ahead of the second regular recall, but would not supplant the first (already in-process) request.

The following lines in the b_cir_13_a program confirm this:

```
IF Z36-RECALL-DATE NOT = ZERO
THEN
*
*   IN THE FUTURE:
*   In case new rush recall was entered, then update z36.
*
GO TO B-CIR-13-1
END-IF.
```

This says that, if we are processing a new (active) recall request, but the Z36-RECALL-xxxx fields have already been populated (by a previous recall), then don't do anything. BUT, "in the future", add some code to check if it's a rush recall and treat it specially.

Thus, currently, the operator needs to delete an existing, in-process recall in order to allow a rush recall to immediately take effect.