Analysis of the issue 939755

[The bug's behavior changes on 75.0.3741.0. Read this bug report for more details]

Way to reproduce the bug:

(0) Assume that two accounts, denoted by account A and the account B, are associated with your chrome device.

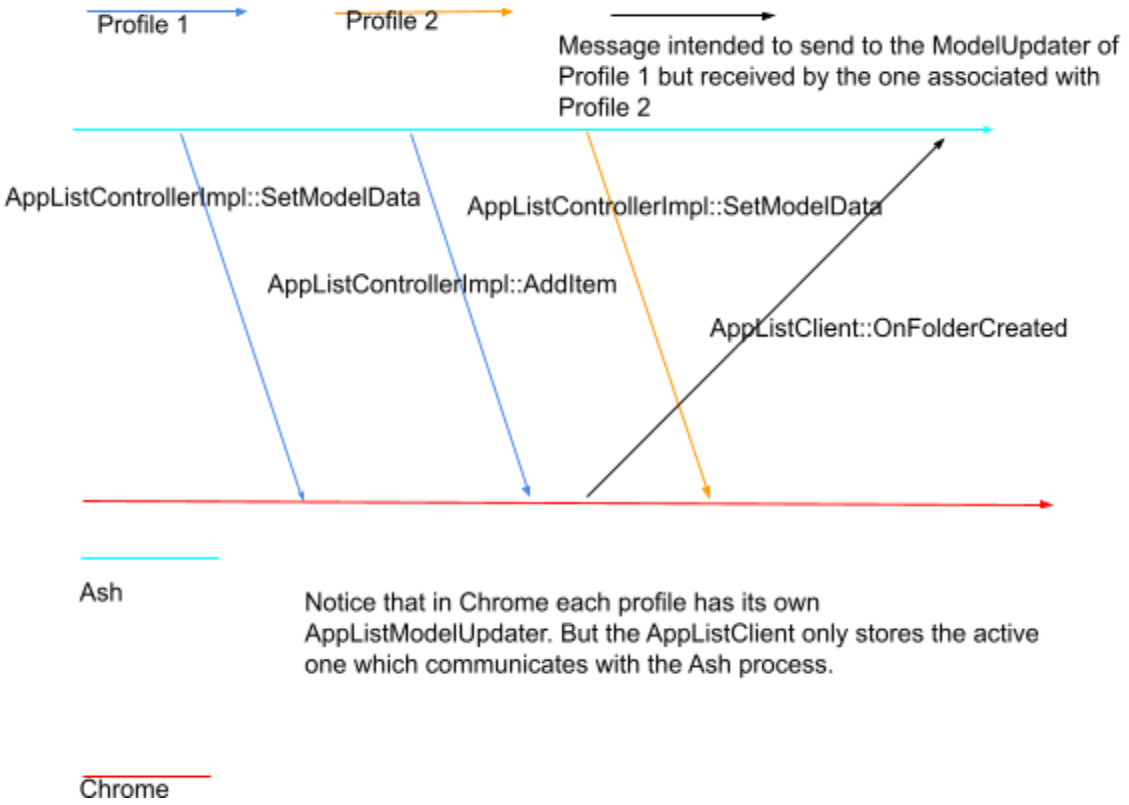
(1) Log in with the account A. In AppListView, choose any two apps, denoted by the app C and the app D, and move them in the same folder.

(2) Switch to account B via clicking the button of "sign in another user". Put the app C and the app D under the root of AppListView.

(3) Execute "Chrome://restart" in the account B. Crash will happen.

Reason:

After the step (3), ChromeOS will sync the account A first. Because the app C and the app D are in folder, ChromeOS will create a folder for them. Notice that the folder is created through mojo. Then ChromeOS will sync the account B. However, the folder creation may finish after the sync of the account B starts. As a result, ChromeAppListModelUpdater::OnFolderCreated is called after AppListClientImpl::SetProfile. And the former function will call ChromeAppListItem::SetMetadata to change the app C and the app D's folder id. Then in ChromeAppListModelUpdater::UpdateAppItemFromSyncItem, ChromeAppListModelUpdater::MoveItemToFolder is called to move the app C and the D from a non-existing folder to the root of the AppListView. So it crashes.



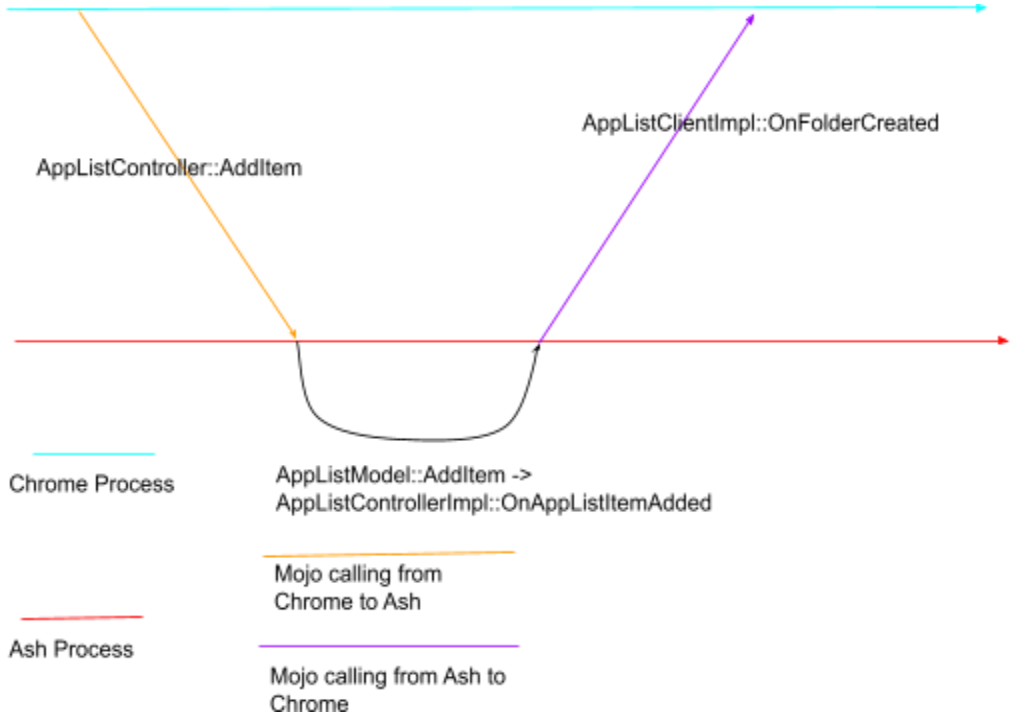
(Illustration of the bug)

How to Fix:

According to @alemate, we should not include the profile information in the mojo calling, although it would be much easier to implement (I tried it and this specific bug can be solved in this way). The better way is to add returned values to the mojo functions of AppListController called from the ChromeAppListModelUpdater. Then the callback of the mojo function is bound with the ChromeAppListModelUpdater instance. Notice that these functions may trigger mojo functions of AppListClient. So do the job of triggered functions in the callback. To summarize, replace the functions provided by the AppListClient mojo interface with Callbacks.

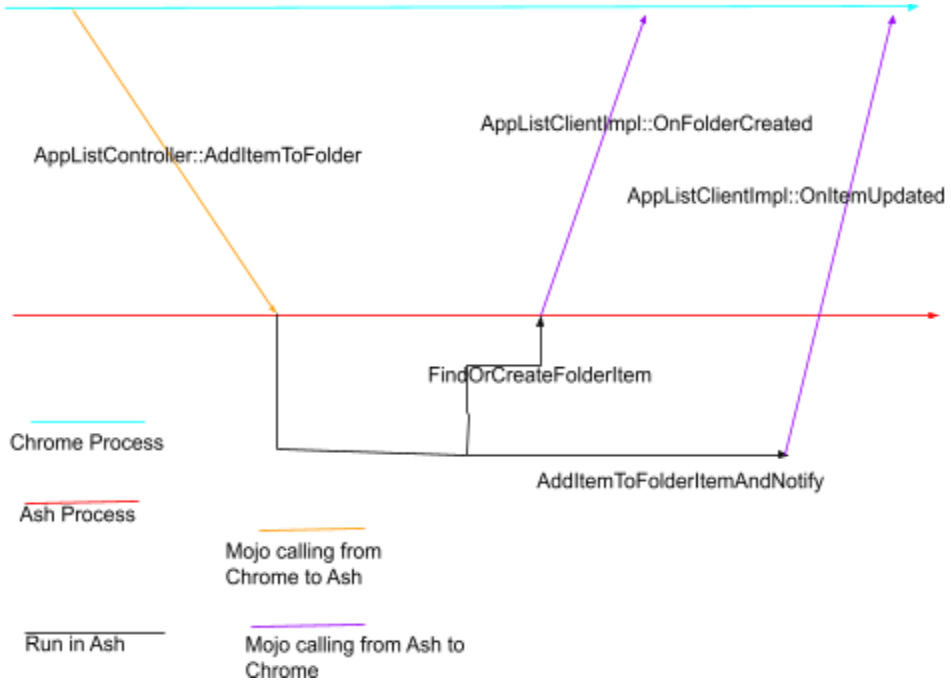
See the following two examples.

(1) Code path 1 in the current code:

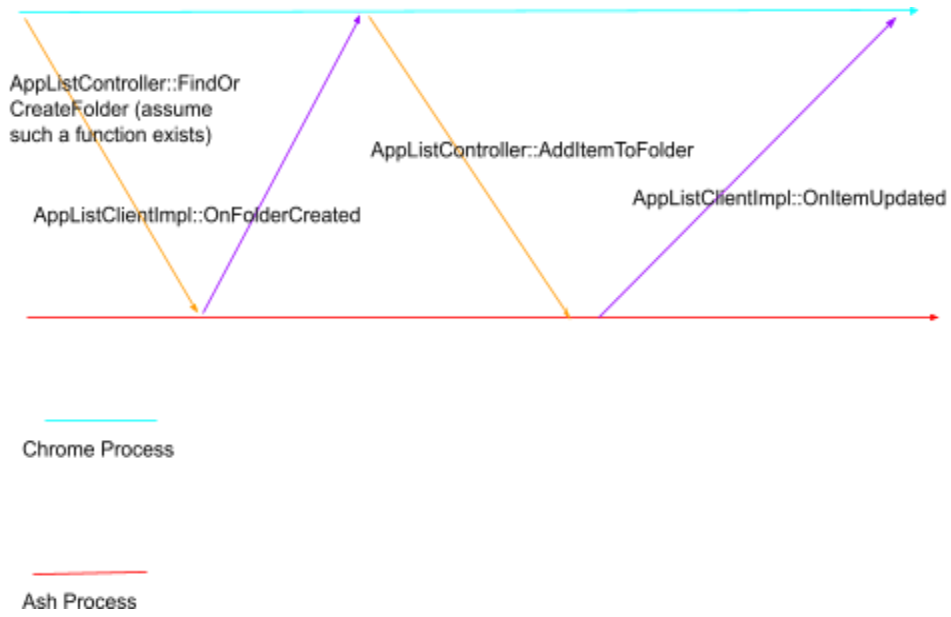


Solution: do the job of AppListClientImpl::OnFolderCreated in the callback of AppListController::AddItem. It ensures that OnFolderCreated is processed by the correct ChromeAppListModelUpdater.

(2) Code path 2 in the current code:



Solution: Different from the code path 1, the code path 2 may trigger two mojo callings: OnFolderCreated and OnItemUpdated. The solution is to refactor the code to make sure that each mojo calling from Chrome will trigger at most one mojo calling from Ash. Then the ideal procedures should be:



Concerns:

Some mojo callings from the AppListController are not triggered from the Chrome process but from the Ash process. For example, the following code path exists:

AppListClient::OnItemUpdated
`AppListControllerImpl::OnAppListItemUpdated`
`AppListModel::AddItemToFolderItemAndNotify`
AppListModel::MergeItems
AppsGridView::MoveItemToFolder
…

The solution mentioned above does not work for those mojo functions. The good news is that it is enough to fix this specific bug.