

# A few notes keep appearing in the root at start-up

I have five notes which keep showing up under the root note. These notes are already under the hierarchy of other notes.

```
Consistency issue fixed: Branch '8yc5SDQRrhss_4aKhk3oLjfnx' was set to root parent since it was referencing missing parent note '8yc5SDQRrhss'
Consistency issue fixed: Branch 'g0UdmDhnHNiH_16FHE0bZt3pX' was set to root parent since it was referencing missing parent note 'g0UdmDhnHNiH'
Consistency issue fixed: Branch 'g0UdmDhnHNiH_1PhkrImWTcA' was set to root parent since it was referencing missing parent note 'g0UdmDhnHNiH'
Consistency issue fixed: Branch 'g0UdmDhnHNiH_6QnjJ4hwGqRo' was set to root parent since it was referencing missing parent note 'g0UdmDhnHNiH'
Consistency issue fixed: Branch 'sLHEyPuAseQD_vw4MhGZe1phE' was set to root parent since it was referencing missing parent note 'sLHEyPuAseQD'
```

Seems that there is a server error right after these logs:

```
TypeError: childToParents[noteId] is not iterable
    at checkTreeCycle (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:65:54)
    at checkTreeCycle (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:83:41)
    at checkTreeCycle (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:83:41)
    at ConsistencyChecks.checkTreeCycles (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:97:33)
    at ConsistencyChecks.checkAndRepairTreeCycles (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:110:21)
    at ConsistencyChecks.runAllChecksAndFixers (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:678:18)
    at ConsistencyChecks.runChecksInner (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:724:14)
    at /nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:704:34
    at Object.doExclusively (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/sync_mutex.js:13:22)
    at async ConsistencyChecks.runChecks (/nix/store/8kx0ii608f10q51l769z2y7x3g7nscg4-trilium-desktop-0.58.7/share/trilium/resources/app.asar/src/services/consistency_checks.js:703:9)
```

Was able to reproduce the issue with a fully anonymised database.

- Let's start with the `8yc5SDQRrhss_4aKhk3oLjfnx` branch.
  - Note ID: `4aKhk3oLjfnx` (exists).
  - Parent Note ID: `8yc5SDQRrhss` (does not actually exist).
  - There are a total of three branches pointing to `4aKhk3oLjfnx`:
    - `Qm1p9UbgJqeu_4aKhk3oLjfnx` → `Qm1p9UbgJqeu` (this one is good).
    - `8yc5SDQRrhss_4aKhk3oLjfnx` → `8yc5SDQRrhss` (the problematic one, notice how it did not get removed).
    - `root_4aKhk3oLjfnx` → `root` (probably created as part of the consistency fixer).

The problem with the old branch not being removed (and thus the recovery process being done at each start-up seems to be related to the ID):

```
### getBranch(8yc5SDQRrhss_4aKhk3oLjfnx)
upsert(branches, branchId,
{"branchId":"root_4aKhk3oLjfnx","noteId":"4aKhk3oLjfnx","parentNoteId":"root","prefix":null,"notePosition":10,"isExpanded":false,"isDeleted":false,"utcDateModified":"2023-02-04 22:49:27.361Z"})
Consistency issue fixed: Branch '8yc5SDQRrhss_4aKhk3oLjfnx' was set to root parent since it was referencing missing parent note '8yc5SDQRrhss'
```

The old branch ID is `8yc5SDQRrhss_4aKhk3oLjfnx`, but then it becomes `root_4aKhk3oLjfnx`.

This seems to be caused by `branch.js#beforeSaving`, where the ID is reconstructed before saving:

```
this.branchId = `${this.parentNoteId}_${this.noteId}`;
```

(introduced a month ago in [0758c82983](#)).

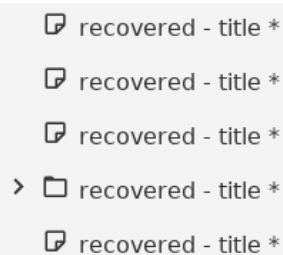
A possible solution to fix this without breaking the recent modifications would be to delete the old branch before recreating the new one.

After introducing this fix, the logs look better now:

```
Marking branches 8yc5SDQRrhss_4aKhk3oLjfnx as deleted
Consistency issue fixed: Branch '8yc5SDQRrhss_4aKhk3oLjfnx' was was missing parent note '8yc5SDQRrhss', so it was deleted.
root_4aKhk3oLjfnx was created in the root instead.
Marking branches g0UdmDhnHnIH_16FHE0bZt3pX as deleted
Consistency issue fixed: Branch 'g0UdmDhnHnIH_16FHE0bZt3pX' was was missing parent note 'g0UdmDhnHnIH', so it was deleted.
root_16FHE0bZt3pX was created in the root instead.
Marking branches g0UdmDhnHnIH_1PhkrImWtTcA as deleted
Consistency issue fixed: Branch 'g0UdmDhnHnIH_1PhkrImWtTcA' was was missing parent note 'g0UdmDhnHnIH', so it was deleted.
root_1PhkrImWtTcA was created in the root instead.
Marking branches g0UdmDhnHnIH_6QnjJ4hWGqRo as deleted
Consistency issue fixed: Branch 'g0UdmDhnHnIH_6QnjJ4hWGqRo' was was missing parent note 'g0UdmDhnHnIH', so it was deleted.
root_6QnjJ4hWGqRo was created in the root instead.
Marking branches sLHEyPuAseQD_vw4MhGZe1phE as deleted
Consistency issue fixed: Branch 'sLHEyPuAseQD_vw4MhGZe1phE' was was missing parent note 'sLHEyPuAseQD', so it was deleted.
root_vw4MhGZe1phE was created in the root instead.
```

The previous exceptions from the cyclic check no longer appear either.

As an added benefit, the notes which are added to the root note are also marked with the “recovered” prefix, indicating that they are “out of place”.



A screenshot of a list of notes. Each note is preceded by a folder icon and the text 'recovered - title \*'. The list contains five such entries, with a right-pointing chevron icon before the second entry.

The only downside with this consistency check (`findBrokenReferenceIssues`) is that it does not seem to treat the situation where it finds a branch with a bad parent, but that note has some other correct branches.

This is what happened here, the parent was removed by the sync at some point, but the branch (which was local to one of the devices) got stuck. In this situation, removing the branch without recreating it at all would have worked, since the note was already placed in the right location.

But there's no reason to bother for the moment, as it's OK to indicate to the user that a branch had to be fixed. They can easily remove it and it won't reappear.