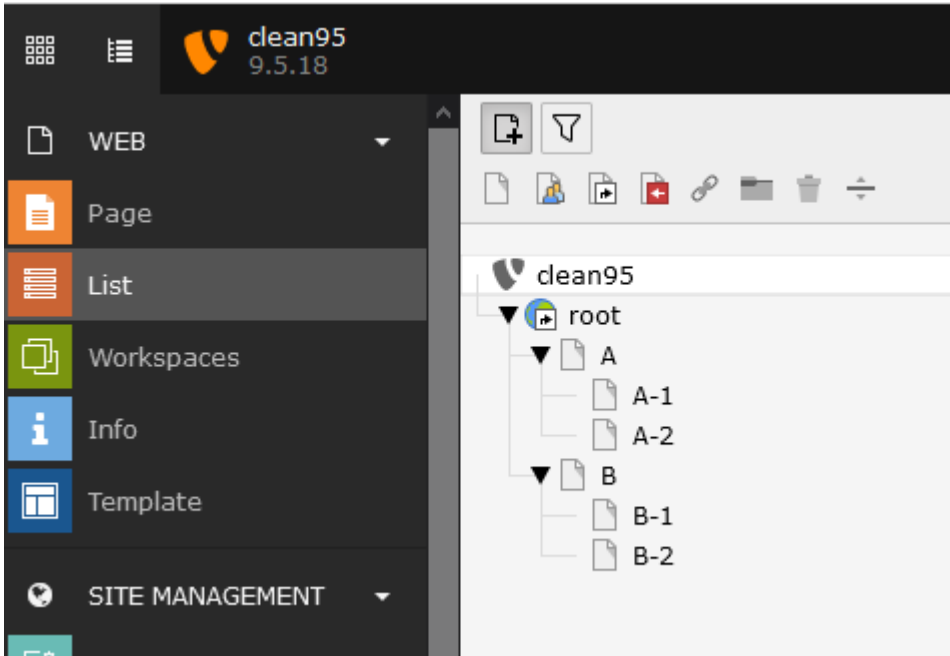


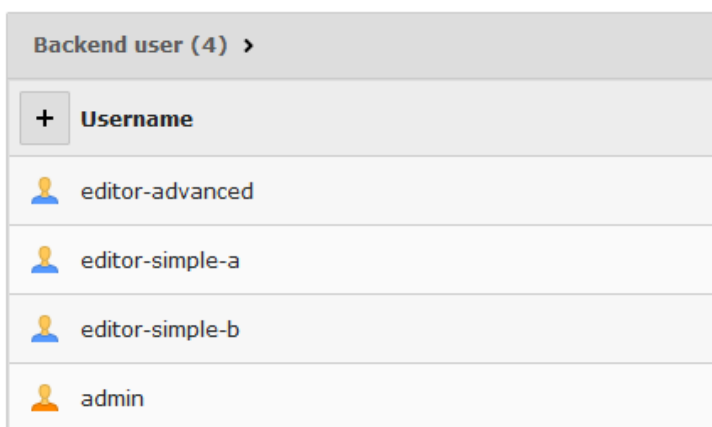
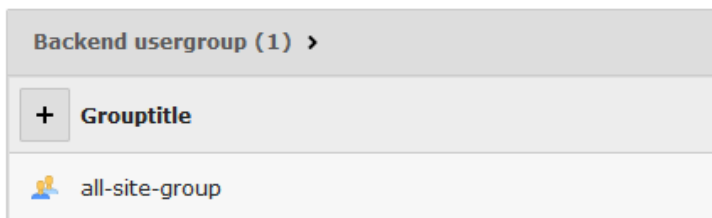
Double page tree when be user editor access a single branch of the workspace

Take the case that there are advanced editors who can modify all the pages of a workspace and publish live, while there are limited operators who can modify only single parts of the workspace tree and cannot publish.

Example page tree:



Workspace configuration:



Edit Workspace "DRAFT WS" on root level

General Internal Stages Custom Stages Mountpoints Publishing / Access Notes

Title [title] **Freeze Editing** [freeze]

DRAFT WS [0]

Notification settings during state change [stagechg_notification]

Disable notifications [0]

Users

Owners [adminusers]

Find records

Backend user Backend usergroup

Members [members]

Find records

all-site-group [be_groups_1]

Backend user Backend usergroup

all-site-group [1]

Edit Workspace "DRAFT WS" on root level

General Internal Stages Custom Stages Mountpoints

DB Mounts [db_mountpoints]

Find records

root [pages_1]
A [pages_2]
B [pages_3]

Page

root [1]

A [2]

B [3]

Backend user configuration:

Edit Backend user "editor-advanced" on ro

General Access Rights Mounts and Workspaces Options Ac

Enabled [disable]

[0]

Admin (!) [admin]

[0]

Username [username]

editor-advanced

Password [password]

Avatar [avatar]

Allowed file extensions

Group [usergroup]

Selected Items

all-site-group [1]

Edit Backend user "editor-advanced" on root level

General Access Rights **Mounts and Workspaces** Options Access Notes


Workspace permissions [workspace_perms]


Edit Live (Online) [1]

DB Mounts [db_mountpoints]

Q Find records

root [pages_1]

 Page

 root [1]

Edit Backend user "editor-simple-a" on root level

General | Access Rights | Mounts and Workspaces | Options | Access | Note

Enabled [disable]
 [0]

Admin (!) [admin]
 [0]

Username [username]

Password [password]

Avatar [avatar]

Allowed file extensions

Group [usergroup]
Selected Items

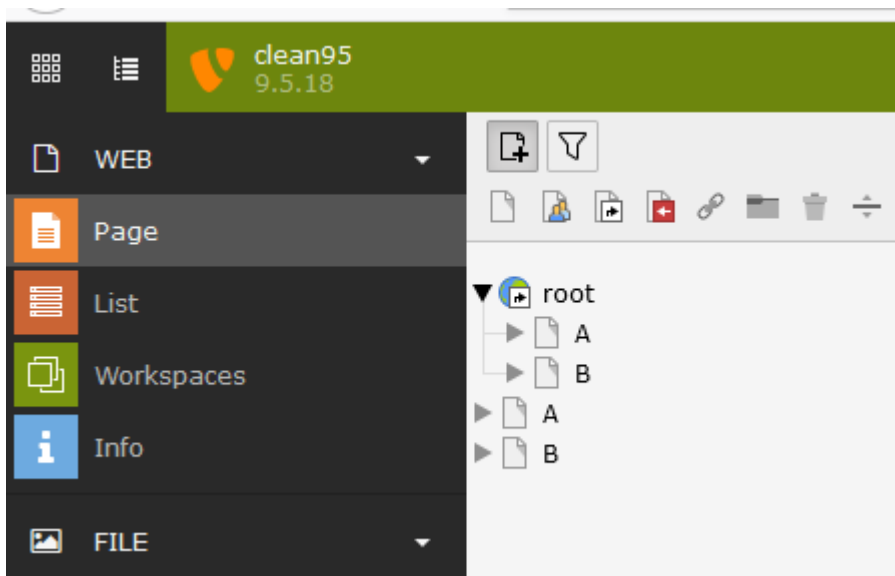
Edit Backend user "editor-simple-a"

The screenshot shows the 'Mounts and Workspaces' tab of a user configuration interface. At the top, there are four tabs: 'General', 'Access Rights', 'Mounts and Workspaces', and 'Options'. The 'Mounts and Workspaces' tab is active. Below the tabs, there is a section titled 'Workspace permissions [workspace_perms]' with a toggle switch for 'Edit Live (Online) [0]' which is currently turned off. Below this is a section titled 'DB Mounts [db_mountpoints]' which contains a search bar with the text 'Find records' and a list of records. The first record is 'A [pages_2]'. Below the list, there are two buttons: 'Page' and 'A [2]'. The 'Page' button is highlighted.

In the "DRAFT WS", "editor-simple-a" have correctly accesso to page tree (same for "editor-simple-b"):

The screenshot shows a user interface for a system named 'clean95 9.5.18'. On the left, there is a navigation menu with the following items: 'WEB', 'Page', 'List', 'Workspaces', 'Info', 'FILE', and 'Filelist'. The 'Page' item is selected. On the right, there is a toolbar with icons for adding, filtering, and deleting. Below the toolbar, there is a page tree structure showing a root node 'A' with two child nodes 'A-1' and 'A-2'.

The user "editor-advanced" show double page tree, this is not correct!:



To fix this problem we have a patch into the file BackendUserAuthentication:

```
protected function initializeDbMountpointsInWorkspace()
{
    $dbMountpoints = trim($this->workspaceRec['db_mountpoints'] ?? '');
    if ($this->workspace > 0 && $dbMountpoints != '') {
        $filteredDbMountpoints = [];
        // Notice: We cannot call $this->getPagePermsClause(1);
        // as usual because the group-list is not available at this point.
        // But bypassing is fine because all we want here is check if the
        // workspace mounts are inside the current webmounts rootline.
        // The actual permission checking on page level is done elsewhere
        // as usual anyway before the page tree is rendered.
        $readPerms = '1=1';
        // Traverse mount points of the

        // PATCH begin
        $wsWebmounts = implode(',', GeneralUtility::intExplode(',', $this->workspaceRec['db_mountpoints']));

        $userMounts = $this->dataLists['webmount_list'];

        $wsArr = GeneralUtility::intExplode(',', $wsWebmounts);
        $userArrTemp = GeneralUtility::intExplode(',', $userMounts);
        $userArr = array();
        if (is_array($userArrTemp) && count($userArrTemp) > 0) {
            foreach ($userArrTemp as $userPage) {
                $rL = BackendUtility::BEgetRootLine($userPage);
                if (is_array($rL) && count($rL)) {
                    $rPages = array();
                    foreach ($rL as $rPage)
                        $rPages[] = $rPage['uid'];
                    if (is_array($wsArr) && count($wsArr) > 0) {

                        $valore = array(
                            $wsArr[0]
                        );
                    }
                }
            }
        }
    }
}
```

```

        if (is_array($rPages) && in_array($rPages, $valore)) {
            $userArr[] = $userPage;
        }
    }
}
}
$resultArr = array_unique(array_merge($userArr, array_intersect($wsArr,
$userArrTemp)));
$webmountUidList = implode(',', $resultArr);
}

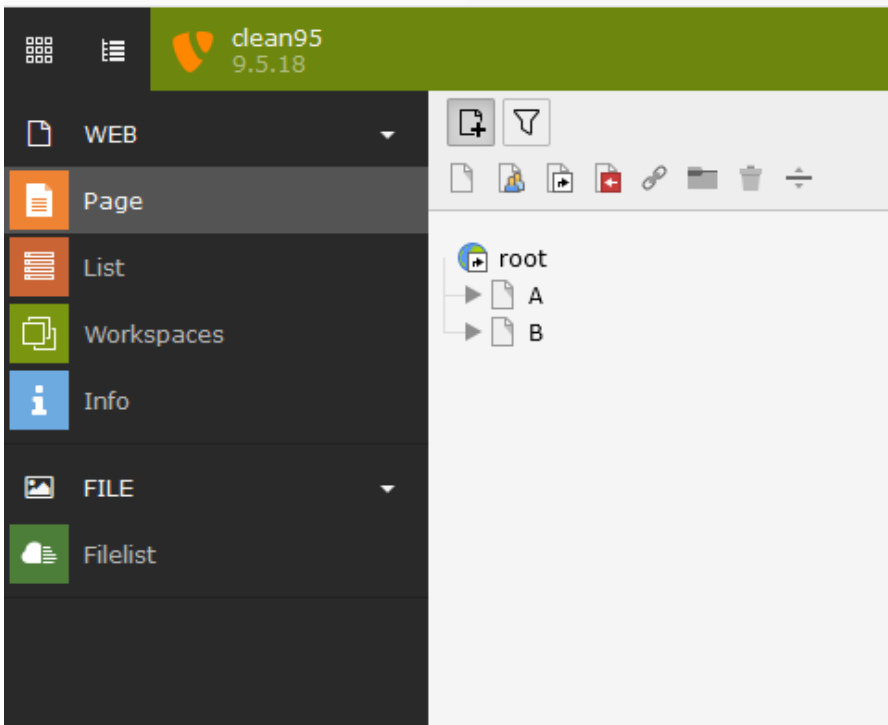
$dbMountpoints = GeneralUtility::intExplode(',', $webmountUidList);

foreach ($dbMountpoints as $mpId) {
    if ($mpId > 0 && $this->isInWebMount($mpId, $readPerms)) {
        $filteredDbMountpoints[] = $mpId;
    }
}
// PATCH end

// Re-insert webmounts:
$filteredDbMountpoints = array_unique($filteredDbMountpoints);
$this->groupData['webmounts'] = implode(',', $filteredDbMountpoints);
}
}
}

```

After patching the core, editor-advanced view correctly page tree:



A limit of this patch is if you use more workspace, the admin user must have a db mount to switch to another workspace and I show all the db mount of the selected workspace.