

Gastzugang und spezielle Anpassungen (serverseitiger Code/Scripts)

Wir wollen unsere Inventarplattform so aufbereiten, dass sie auch für Gäste etwas zum Stöbern bietet und die Funktionen von Teedy als praktikable Open Source Software zeigt. Teedy hat einen Gastmodus, der von Haus aus leider neben der regulären Verwendung auch ein gewisses Spamming per Web-Browser und API erlaubt. Zur Reduktion haben wir Scripts geschrieben, die dies unterbinden.

Wir möchten Kommentare und wild angelegte Tags und Dokumente damit vermeiden, da sie die Struktur zu leicht unlesbar machen. Gäste können Kommentare, Tags und Dokumente von anderen Gastsitzungen generell einfach modifizieren oder löschen. Deshalb eignet sich der Gastmodus nur als reiner Betrachtermodus. Selbiges trifft auch geteilte/generische) ReadOnly-Benutzer zu.

Die Speicher-Quota für den Gast und für sonstige ReadOnly-Benutzer beträgt 0 MB.

Neben Gast-Tags werden Tags von anderen Nutzern, die keine Admins oder Editoren sind, ebenso vom System erkannt und automatisch gelöscht. Wir behalten uns vor unser System für Mitglieder zu kapseln, sodass das Tool nur noch vereinsintern und nicht durch Gäste genutzt werden kann, falls zu viele Schabernackversuche auftreten.

Scripts (diese laufen jeweils als cron job alle 10 Minuten)

- **Kommentare** (erzeugt von "guest" user) werden automatisch gescannt und gelöscht
 - Siehe [Auto-delete guest comments](#).
 - Ein Wiederherstellen ist mit dem Zurücksetzen der **com_deletedate_d** Spalte möglich
 - Der Haupt-Admin erhält eine Mail über alle gelöschten Kommentare ins Webmaster-Postfach
- **Dokumente** (erzeugt von "guest" user oder anderen Benutzern, die nicht den Gruppen "Editoren" oder "Administratoren" angehören) werden automatisch gescannt und ebenso gelöscht
 - Siehe [Auto-delete guest documents](#).
 - Ein Wiederherstellen ist mit dem Zurücksetzen der **doc_deletedate_d** Spalte möglich. Ein Wiederherstellen der Dateien ist allerdings nicht möglich
 - Der Haupt-Admin erhält eine Mail über alle gelöschten Dokumente ins Webmaster-Postfach

- **Tags** (erzeugt von "guest" user oder anderen Benutzern, die nicht den Gruppen "Editoren" oder "Administratoren" angehören) werden automatisch gescannt und gelöscht
 - Siehe [Auto-delete guest tags](#)
 - Ein Wiederherstellen ist mit dem Zurücksetzen der **tag_deletedate_d** Spalte möglich
 - Der Haupt-Admin erhält eine Mail über alle gelöschten Tags ins Webmaster-Postfach

Manuelle Pflege der Tag-Besitzer

Wer Tags sehen und bearbeiten kann, kann vom Admin entweder über die Datenbank oder das User Interface gesteuert werden.

defekt

Name

Farbe

Übergeordnet

Speichern

Berechtigungen für dieses Tag werden auch auf Dokumente angewendet, die mit einem Tag versehen sind **defekt**

Für	Zugriffsberechtigung
Gruppe Editoren	Kann lesen ✕ Kann schreiben ✕
Gruppe Aktivmitglieder	Kann lesen ✕
Gruppe ErweiterteBetrachter	Kann lesen ✕

Aus Gründen der Gesamtübersicht (Grafana ACL Tabelle) entfernen wir Einzelbenutzer und definieren **nur** Gruppen. Leider können in Teedy Tags nicht von ihrem Einzelbenutzer getrennt werden. Zumindest können die Tags nicht gelöscht werden, die vom Haupt-Admin erstellt worden sind. Letzteres ist nur über Datenbank Update SQL Scripts änderbar:

```

/*Finde alle Tags, die nicht dem Haupt-Admin gehören. Gast-Tags tauchen hier generell nicht
auf*/
SELECT
    T.tag_name_c AS "Tag",
    U.use_username_c AS "Ersteller",
    T.tag_createdate_d AS "Erstelldatum"
FROM
    t_tag AS T
  
```

```

JOIN t_user AS U ON U.use_id_c = T.tag_iduser_c
WHERE
    T.tag_deletedate_d IS NULL AND
    T.tag_iduser_c IN (
        SELECT
            use_id_c
        FROM t_user AS U
        JOIN t_user_group AS UG ON UG.ugp_iduser_c = U.use_id_c
        JOIN t_group AS G ON G.grp_id_c = UG.ugp_idgroup_c
        WHERE
            U.use_deletedate_d IS NULL AND
            UG.ugp_deletedate_d IS NULL AND
            G.grp_deletedate_d IS NULL
    ) AND
    U.use_id_c != 'admin'
;

```

/*
Übertrage Ersteller/Eigentümer des Tags von anderen Editor/Administratoren auf den Haupt-Administrator.
Dazu müssen nachträglich ebenso die ACLs angepasst werden. Die Spalte "tag_iduser_c" spielt für die Berechtigung keine direkte Rolle sondern enthält nur die Information, wer den Tag initial erstellt hat.

Aber da wir in den ACLs den User gegen admin tauschen, machen wir es somit konsistent!

```

*/
/*UPDATE t_tag SET tag_iduser_c = 'admin';*/ /*dieses Statement ist überflüssig und sollte nicht ausgeführt werden*/

```

/*
Finde alles, was nicht "Editoren", "ErweiterteBetrachter" oder "Aktivmitglieder" ist.

```

*/
SELECT
    A.acl_id_c,
    T.tag_name_c "Tag",
    A.acl_perm_c AS "Permission",
    U.use_username_c
FROM t_tag AS T
JOIN t_acl AS A ON A.acl_sourceid_c = T.tag_id_c
LEFT JOIN t_user AS U ON U.use_id_c = A.acl_targetid_c

```

```

WHERE
    T.tag_deletedate_d IS NULL AND
    --A.acl_deletedate_d IS NULL AND
    U.use_disableddate_d IS NULL AND
    U.use_id_c = 'admin'
ORDER BY T.tag_name_c
;

/*
Wir wollen die ACLs nur auf Gruppen festlegen. Wir nehmen auch dem Haupt-Admin die
Benutzerberechtigungen, da dieser Einzelbenutzer die Übersichtlichkeit (in Grafana) künstlich
aufbläht.
Berechtigungen für Einzelnutzer soll es generell nicht geben (Ausnahme Tag "public" für den
User "guest")
*/
UPDATE t_acl SET acl_deletedate_d = NOW() WHERE acl_id_c IN (
    SELECT
        A.acl_id_c
    FROM t_tag AS T
    JOIN t_acl AS A ON A.acl_sourceid_c = T.tag_id_c
    LEFT JOIN t_user AS U ON U.use_id_c = A.acl_targetid_c
    WHERE
        T.tag_deletedate_d IS NULL AND
        U.use_disableddate_d IS NULL AND
        U.use_id_c = 'admin'
)
;

```

Manuelle Pflege der Dokumenten-Besitzer

Wer Dokumente sehen und bearbeiten kann, kann vom Admin entweder über die Datenbank oder das User Interface gesteuert werden:

[Teilen](#)[Handwerkzeug](#) [Metallwerkstatt](#) [Schrauben](#)[Inhalt](#) [Workflow](#) [Berechtigungen](#) [Aktivitäten](#)

Die Berechtigungen können direkt auf dieses Dokument angewendet werden, oder können von [tags](#) vorgegeben werden.

Berechtigungen auf diesem Dokument

Für	Zugriffsberechtigung
Benutzer MarioVoigt	Kann lesen ✕ Kann schreiben ✕

Zugriffsberechtigung hinzufügen

Für

Zugriffsberechtigung

[+ Hinzufügen](#)

/*Finde alle Dokumente, die nicht dem Haupt-Admin gehören. Gast-Tags tauchen hier generell nicht auf*/

SELECT

D.doc_title_c AS "Gegenstand",
U.use_username_c AS "Ersteller",
D.doc_createdate_d AS "Erstelldatum"

FROM

t_document AS D

JOIN t_user AS U ON U.use_id_c = D.doc_iduser_c

WHERE

D.doc_deletedate_d IS NULL AND

D.doc_iduser_c IN (

SELECT

use_id_c

FROM t_user AS U

JOIN t_user_group AS UG ON UG.ugp_iduser_c = U.use_id_c

JOIN t_group AS G ON G.grp_id_c = UG.ugp_idgroup_c

WHERE

U.use_deletedate_d IS NULL AND

UG.ugp_deletedate_d IS NULL AND

G.grp_deletedate_d IS NULL

) AND

U.use_id_c != 'admin'

;

```

/*
Übertrage Ersteller/Eigentümer des Tags von anderen Editor/Administratoren auf den Haupt-
Administrator.

Dazu müssen nachträglich ebenso die ACLs angepasst werden. Die Spalte "doc_iduser_c" spielt
für die Berechtigung keine direkte Rolle sondern enthält nur die Information, wer das Dokument
initial erstellt hat.

Aber da wir in den ACLs den User gegen admin tauschen, machen wir es somit konsistent!
*/

/*UPDATE t_document SET doc_iduser_c = 'admin';*/ /*dieses Statement ist überflüssig und
sollte nicht ausgeführt werden*/

/*
Finde alles, was nicht "Editoren", "ErweiterteBetrachter" oder "Aktivmitglieder" ist.
*/
SELECT
    A.acl_id_c,
    D.doc_title_c "Gegenstand",
    A.acl_perm_c AS "Permission",
    U.use_username_c
FROM t_document AS D
JOIN t_acl AS A ON A.acl_sourceid_c = D.doc_id_c
LEFT JOIN t_user AS U ON U.use_id_c = A.acl_targetid_c
WHERE
    D.doc_deletedate_d IS NULL AND
    --A.acl_deletedate_d IS NULL AND
    U.use_disabledate_d IS NULL AND
    U.use_id_c = 'admin'
ORDER BY D.doc_title_c
;

```

Version #2

Erstellt: 14 Mai 2025 13:55:09 von Mario Voigt

Zuletzt aktualisiert: 15 Mai 2025 10:17:34 von Mario Voigt