EMu Documentation

EMu Read-only Modes

Document Version 1

EMu Version 4.0.03

Contents

S ECTION 1

EMu Read Only Modes

Overview

KE EMu 4.0.03 adds two new Registry entries to make all or part of the system read-only. These entries allow system administrators to "turn off" record insertions and updates while still allowing the system to be operational. Prior to EMu 4.0.03 it was possible to make the back-end tables read-only (and it still is), however the EMu client did not reflect the read-only nature of the table in the module interface: it was still possible to select the menu entry to insert a new record, but an error would then be displayed as the table was read-only. The `ReadOnly` Registry entries control the EMu client while still leaving the back-end tables operational. This means that the menu entry to insert a new record is now disabled if the table or system is marked as read-only.

The system wide read-only entry ensures that data is not modified in any modules, including the EMu Registry. Some uses for this entry include:

- Disabling data changes while the system is upgraded.
- Providing read-only access for certain users / groups (e.g. students).
- Allowing data loads to be checked before going live.

The table specific read-only entry allows individual tables, or all tables, to operate in a read-only state. If a table is read-only, data insertions and modifications are not permitted. The only exception is that changes to the EMu Registry are permitted even if the eregistry table is set to read-only. Possible uses for this entry include:

- Disabling data changes to a single table (as it may require maintenance).
- Providing read-only access to a table for certain users/groups (e.g. students).
- Allowing Registry based operations (e.g create a new sort) while the data is read-only.

The addition of the two `ReadOnly` Registry entries provides system administrators with the ability to control access to data stored in EMu at a system wide and table specific basis.

The `ReadOnly` Registry entries work alongside existing EMu Registry entries. For example, if a table is designated as read/write by a `ReadOnly` Registry entry, a user still needs the `daInsert` operational privilege to be able to create records. The `ReadOnly` entries do not override existing Registry entries to provide more privileges than would be the case if the entry was not set.

System ReadOnly Registry setting

The format of the system `ReadOnly` Registry setting is:

```
System|Setting|ReadOnly|value
Group|Default|Setting|ReadOnly|value
Group|group|Setting|ReadOnly|value
User|user|Setting|ReadOnly|value
```

where:

    value   is either `true` (read-only functionality is enabled) or `false` (read-only functionality is not enabled).

The system `ReadOnly` entry provides the following functionality:

- Data in all modules cannot be updated or created.
- Operations involving updates to tables are disabled, namely:
  - Creating groups (uses egroups table).
  - Manipulating the records in groups (uses egroups table).
  - Creating batch exports (uses eschedule table).
  - Running batch exports (uses eexports table).
  - Create task templates (uses etemplates table).
  - Change help contents (uses efieldhelp table).
- Operations that update the Registry are disabled, namely the creation, modification or deletion of any resources:
  - List Settings
  - Default Values
  - Ditto Record
  - Record Recall
  - Record Template
  - Reports
  - Shortcut Settings
  - Sorts
- Operations that update the Registry when performed, but only to record the entry selected are not disabled, namely:
  - List Settings
  - Page View
  - Query Default Values
  - Reports
  - Shortcut Settings

- Sorts
- Ad-hoc Sorts

The system `ReadOnly` entry may be used to limit access on a group or user basis. For example, if users in group `Student` may not update any data, the following entry may be used:

```
Group|Student|Setting|ReadOnly|true
```

Entries may also be used in combination to achieve the desired effect. For example, if all users except those in group `Admin` should have read-only access, the following entries may be used:

```
System|Setting|ReadOnly|true
Group|Admin|Setting|ReadOnly|false
```

Table ReadOnly Registry Setting

The format of the table `ReadOnly` Registry setting is:

Group|Default|Table|Default|ReadOnly|*value*

Group|Default|Table|*table*|ReadOnly|*value*

Group|*group*|Table|Default|ReadOnly|*value*

Group|*group*|Table|*table*|ReadOnly|*value*

User|*user*|Table|Default|ReadOnly|*value*

User|*user*|Table|*table*|ReadOnly|*value*

where:

value   is either `true` (all operations that would result in record creation or updates in the specified table are disabled) or `false` (update operations are allowed, subject to other permissions, e.g. operational permissions).

The one exception to the rule is the Registry table (eregistry). If the Registry is made read-only, only explicit changes, that is changes made from the Registry module, are disabled. All implicit changes (e.g. adding a new sort definition) are still permitted. Hence the creation, modification or deletion of any resources is permitted, namely:

- List Settings
- Default Values
- Ditto Record
- Record Recall
- Record Template
- Reports
- Shortcut Settings
- Sorts

Using the first form of the table `ReadOnly` Registry entry:

```
Group|Default|Table|Default|ReadOnly|true
```

disables data creation and updates in all tables while still permitting most operational commands (e.g. report creation). When compared with the system `ReadOnly` setting, the table setting restricts only data modifications, rather than data and operational functionality.

User/group based variants of the table Registry entry may be used to restrict access to a select number of individuals. For example, if users in group `Student` are not allowed to create or modify records in the Catalog module, the following Registry entry may be used:

```
Group|Student|Table|ecatalogue|ReadOnly|true
```

As with the system `ReadOnly` entry, table based entries may be combined to produce the desired effect. For example, if users in group `Student` were only allowed to update the egroups, eschedule and eexports tables (allowing them to create groups and batch exports), then the following Registry entries may be used:

```
Group|Student|Table|Default|ReadOnly|true
Group|Student|Table|egroups|ReadOnly|false
Group|Student|Table|eschedule|ReadOnly|false
Group|Student|Table|eexports|ReadOnly|false
```

Combining system and table ReadOnly Registry entries

In general, it is not wise to mix system based and table based `ReadOnly` Registry entries. Consider the following settings:

```
Group|Student|Setting|ReadOnly|true
Group|Student|Table|eparties|ReadOnly|false
```

Which entry takes precedence? Can users in group `Student` write to the eparties table? The current implementation gives priority to the system entries over table based entries. So for the example above, users in group `Student` would not be able to write to the Parties module.

Examples

Example 1

Your EMu installation is to be upgraded to the next release of the software. While the upgrade is proceeding you would like users to be able to view data in the live system, but not make any changes as they will be lost when the upgraded system replaces the live system. The following Registry entry may be used:

```
System|Setting|ReadOnly|true
```

The above setting will disable all EMu functions that would result in any data changes. Users may still produce reports, sort records, etc., however data changes are not allowed.

Example 2

You have received an initial data load in EMu ready for checking. You do not want users to modify any data as they are only determining the integrity of the load. In order to make the reporting of issues easier, you would like users to be able to create groups so they can group records where the data does not look correct. The following Registry entries may be used:

```
Group|Default|Table|Default|ReadOnly|true
Group|Default|Table|egroups|ReadOnly|false
```

The disabling of the read-only status on egroups allows users to create and manipulate groups.

Example 3

You have created a duplicate version of your live system for access via the web. You do not want users to be able to alter any data in the web copy, however all other functionality should be available. The following Registry entries may be used:

```
Group|Default|Table|Default|ReadOnly|true
Group|Default|Table|egroups|ReadOnly|false
Group|Default|Table|eschedule|ReadOnly|false
Group|Default|Table|eexports|ReadOnly|false
Group|Default|Table|etemplates|ReadOnly|false
Group|Default|Table|efieldhelp|ReadOnly|false
```

The enabling of read/write access to the egroups, eschedule, eexports, etempaltes and efieldhelp tables will allow full command functionality on a read-only system. You may want to exclude some of these tables if you want to further limit functionality. For example, removing efieldhelp will disable the updating of field level help.

Example 4

You are undertaking a clean up of the Lookup List table (eluts) so you would like to restrict updates to lookup lists to users in group `Admin`. The following Registry entries may be used:

```
Group|Default|Table|eluts|ReadOnly|true
Group|Admin|Table|eluts|ReadOnly|false
```

When the eluts table is read-only, users may not insert new values into the table. This means **all** lookup lists are treated as read-only. Users may find this annoying as it may make it difficult to save records.

Example 5

You have just received an allocation of staff to add field level help for all modules in the system. You do not want the staff to alter any other data apart from field level help. You have created a group called `Volunteers` and placed the allocated staff in the group. The following Registry entries may be used:

```
Group|Volunteers|Table|Default|ReadOnly|true
Group|Volunteers|Table|efieldhelp|ReadOnly|false
```

In order for the volunteers to be able to create the field level help for all modules, you would also need to allocate the `daEditHelp` operational privilege for group `Volunteers`.

Index