

EMu Documentation

Release Notes: EMu 4.0.02

Document Version 1

EMu Version 4.0



Contents

Here you will find collected together the Release Notes for EMu 4.0.02, alongside all documents referenced in the notes. These release notes and documents are also available on the [KE EMu website](#).

This PDF document brings together a number of individually published documents: please note that page numbering below refers to this combined PDF document and not to the page numbers printed at the bottom of pages, as each individual document, e.g. Record Recall, has its own internal numbering:

Release Notes: EMu 4.0.02	5
Support for RTL Languages	25
Scheduled Export Facility	44
Appending data using the EMu Import Facility	95
The ArcExplorer Report	128

Release Notes: EMu 4.0.02

Release Date: 28 June 2010

Requirements

- For Windows 2000, XP, 2003, [Microsoft Windows Services for UNIX](#) (version 3.5)
- [Texpress 8.2.008](#) or later
- [TexAPI 6.0.001](#) or later
- [Perl 5.8](#) or later

New Features

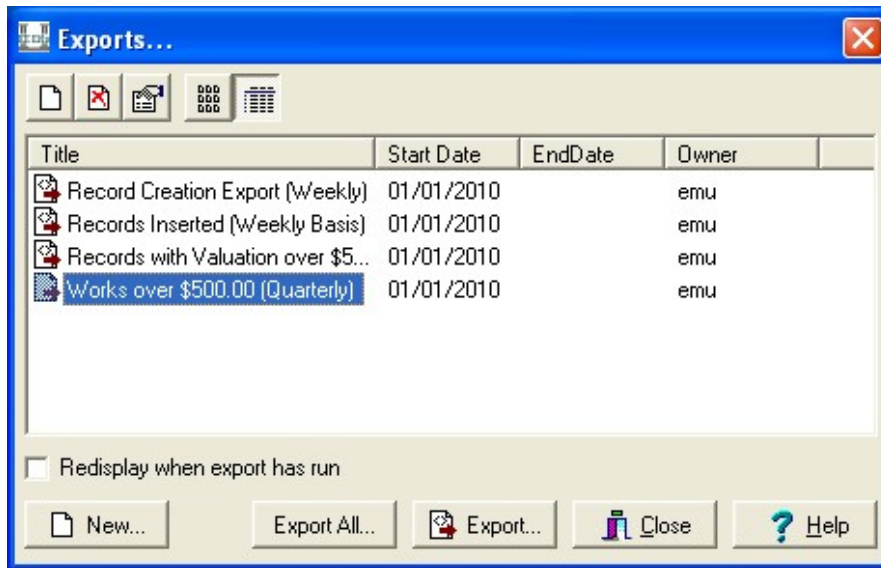
- **Support for RTL languages:** Support for right-to-left (RTL) languages has been added to EMu. Arabic and Hebrew scripts are now supported, as are two additional left-to-right (LTR) languages, Greek and French (Canadian). The user interface has been modified to reverse the display of all controls and images when using RTL languages. Dynamic switching between RTL and LTR languages is supported.



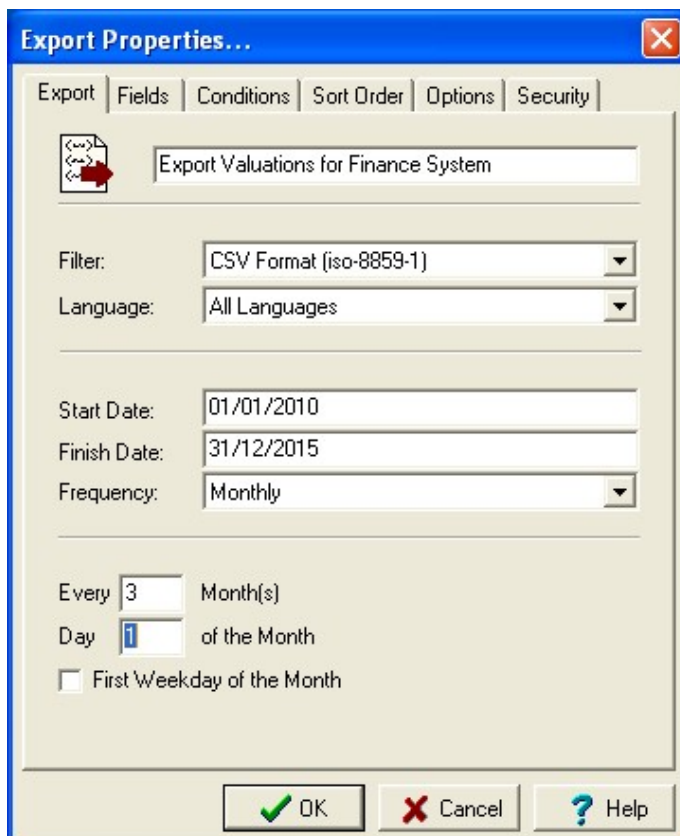
A complete description of the support for RTL languages may be found in the [Right To Left Language Support](#) documentation.

- **Scheduled Export Facility:** The Scheduled Export facility enables data to be exported on a regular basis. Data can be scheduled to be exported on anything from a daily to an annual basis. The export occurs outside working hours (generally early in the morning) and the results are stored on the EMu server. A user may then view the results and save / view the export files produced. The exported data may be processed by a filter to create the output format required (e.g. XML, CSV). The creation of a scheduled export is very similar to

defining a report. A set of fields may be selected and the records may be sorted. A TexQL statement may be defined to limit the records exported.



The Export Properties dialogue box is shown below:



The Scheduled Exports facility provides two new modules:

- **Schedules Module**

The Schedules module holds one record per Scheduled Export containing the definition of the export.

Schedules (1) - Display

File Edit Select View Tools Tabs Multimedia Window Help

Works over \$500.00 (Quarterly), 3 Monthly [01/01/2010 -] 13

Schedule Details

Name: Works over \$500.00 (Quarterly)

Module: ecatalogue

Language: All Languages

Filter:

Schedule Details

Commencement Date: 01/01/2010

Completion Date:

Frequency: Monthly

Schedule

Every 3 Month(s)

Day 1 of the Month

☐ First Weekday of the Month

Schedule Fields Conditions Exports Notes Multimedia Security A

Display Record 9 of 9 emu Admin 20136

- **Exports Module**

The Exports module holds one record per set of export files produced. When a Scheduled Export is run a new exports record is created containing the output of the export process.

Exports (1) - Display

File Edit Select View Tools Exports Tabs Multimedia Window Help

Works over \$500.00 (Quarterly), ecatalogue, Monthly [05/06/2010 - 05/06/2010] 100

Export Details

Schedule: Works over \$500.00 (Quarterly), 3 Monthly [01/01/2010 -]

Frequency: Monthly

Start Date: 05/06/2010

End Date: 05/06/2010

Export Type: ☐ Scheduled ☒ Unscheduled

Module: ecatalogue

Record Count: 1

Export Files

	File Name	File Size
1	/home/ke/emu/artdemo/logs/exports/ecatalogue/Works over \$500.00 (Quarterly)/2010-06-...	269
2	/home/ke/emu/artdemo/logs/exports/ecatalogue/Works over \$500.00 (Quarterly)/2010-06-...	111

Filter Output

Email sent to csvexport@museum.org successfully

Export Notes Multimedia Security Audit Admin

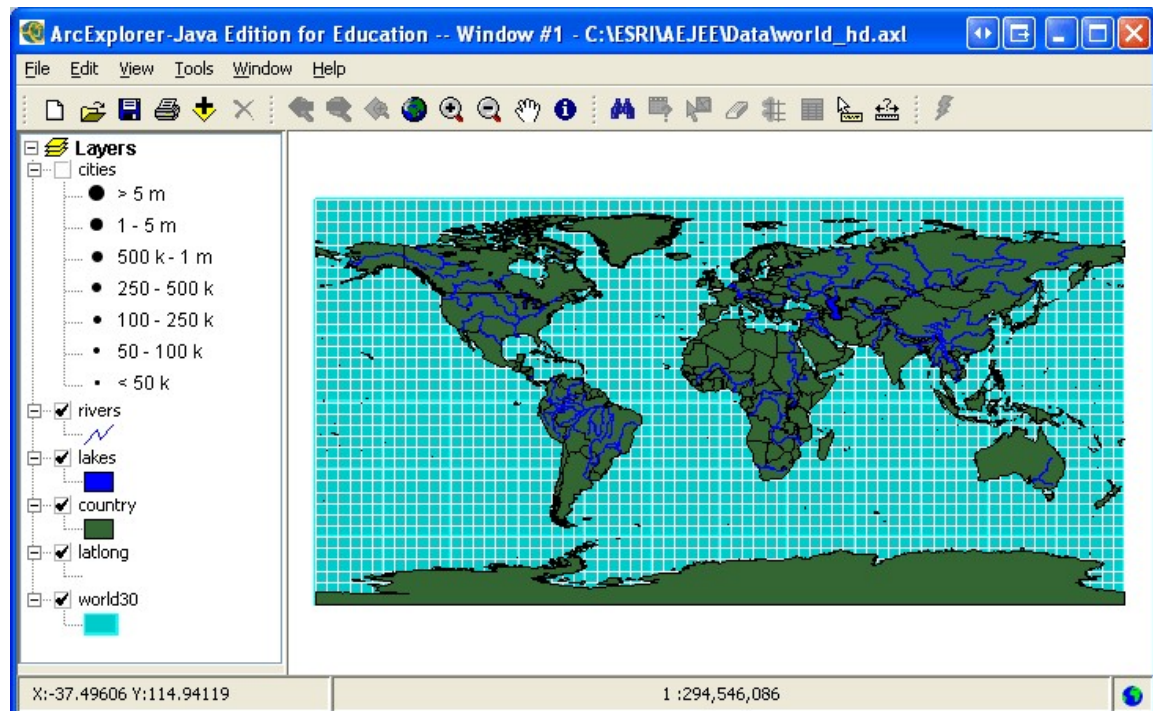
Display Record 1 of 100 emu Admin 20136

A complete description of the Scheduled Export facility can be found in the [Scheduled Exports](#) documentation.

- **Append values using the Import Facility:** The existing Import facility has been extended to allow data to be prepended / replaced / appended in tables and nested tables of values. Support is provided for both XML and CSV (Comma Separated Values) data formats. A grouping mechanism allows data across multiple fields to be added to the same row, allowing grids with multiple columns to have complete rows appended.

A complete description of the Import facility extensions can be found in the [Appending data using the EMu Import Facility](#) documentation.

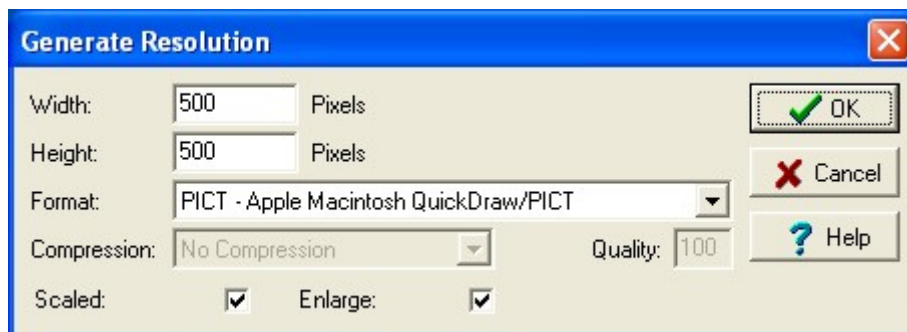
- **Arc Explorer Report Viewer:** A new report type supporting ESRI's [ArcExplorer](#) (a freely distributable GIS viewer) allows latitude / longitude points to be plotted on maps. The ArcExplorer viewer allows layers to be defined and enabled / disabled within the viewer. Data points may have associated metadata which is displayed when a point is selected.



A complete description of the Arc Explorer report type can be found in [The ArcExplorer Report](#) documentation.

Improvements

- **Improved time to save records:** A number of optimisations have been made to the EMu database server (Texpress) leading to significantly reduced record save times. In particular the speed of calculated values and lookup of data in other modules has been enhanced. The improvements affect all operations that alter data, including:
 - Global edits
 - Data imports
 - Bulk update tools (e.g. locations, condition checks, etc.)
 - Single record insertions / edits
- **Alpha channel support in images:** The generation of thumbnail and other resolution images has been enhanced to ensure the alpha channel (or transparency plane) is maintained in generated images. For image formats that do not support transparency (e.g. JPEG), the colour defined in the alpha channel is used as the background colour for derived images.
- **Improved attaching and importing of multimedia:** When adding multimedia, either by attaching an image to a Multimedia module record or by importing a set of images, a link is created to the original image. The link is used to generate the thumbnail and resolutions for the image. Using a link removes the need to copy an image into the EMu cache before saving it to the EMu server, hence saving time and disk space on the local computer.
- **Generate Resolution settings saved:** The settings of the Generate Resolution dialogue box found in the Multimedia module are now saved between invocations and across sessions. The last set of values entered is always displayed when the dialogue box is next displayed.



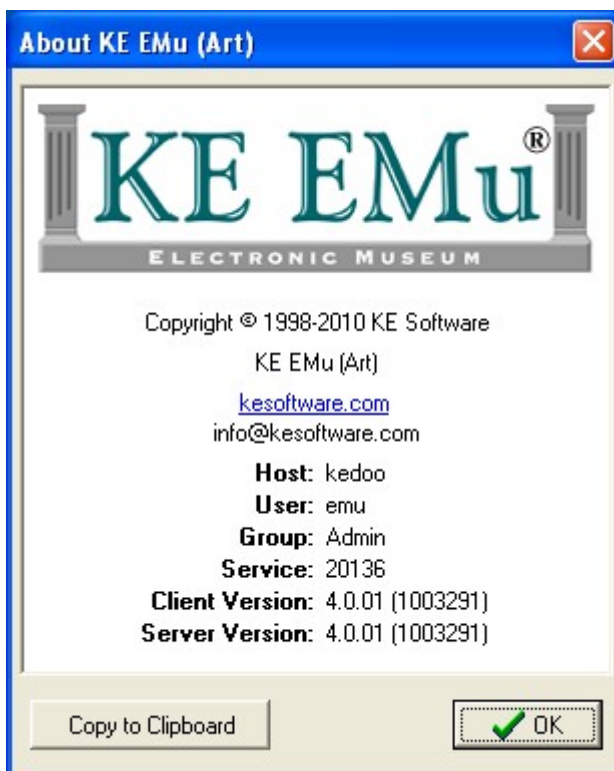
- **Option to suppress module compaction:** The size of the Audit and Statistics modules will grow over time as more records are added to them. Since both modules have a very small / non-existent number of record changes it is not necessary to compact these tables as part of the weekly maintenance schedule. A new option `COMPACT="no"` may be added to a module's `emuoptions` file (stored in the module directory on the EMu server) to skip the compaction of the module when maintenance is performed.
- **Tasks tab added to Locations module:** The standard EMu Tasks tab has been added to the Locations module. The addition allows location based activities to be defined along with the accompanying notifications. For example, a task may be specified requiring a location to be checked after a given time period.
- **User's group added to status bar:** The group to which the current user belongs has been added to the EMu status bar. The status bar now displays the user name, group and service used by the EMu client, along with the current mode and any messages.



- **Abort connection to incorrect client:** The EMu login process has been enhanced to allow the login process to be terminated and the login dialogue redisplayed if an incorrect service is specified. The enhancement is useful for clients who run multiple clients on the one EMu server.



- **Adjust precision of single latitude / longitude centroids:** EMu 4.0.01 saw the introduction of a new set of rules, provided by the EMu Natural History Special Interest Group, regarding the implied precision when calculating centroid values. The rules indicated that for all centroid values calculated an extra digit of precision should be applied to the resultant latitude / longitude point. A number of clients have indicated that implying extra precision where only a single latitude / longitude pair is specified provides a "false" level of accuracy. The centroid calculator has been changed to no longer imply extra precision where a single latitude / longitude pair is specified.
- **About box information:** A *Copy to Clipboard* button has been added to the EMu About dialogue box. The button is used to place a copy of the EMu connection and settings information onto the Windows clipboard. Once on the clipboard the information may be pasted into emails, etc. to help with issue diagnosis.



A sample of the information generated is displayed below:

```
[Client Information]
Client Version: 4.0.01 (1003291)
Application Name: KE EMu (Art)
Application Path: C:\Projects\EMu\Clients\Art\emu.exe
Cache Path: C:\Documents and Settings\bern\Local Settings\Application
Data\KESoftware\Cache
```

```
[Server Information]
Server Version: 4.0.01 (1003291)
Server Path: /home/ke/emu/artdemo
Server Media Path: /home/ke/emu/artdemo/multimedia
```

```
[Texpress Information]
Texpress Version: 8.2.8
TexAPI Version: 3.1.018
Uname: SunOS kedoo 5.10 Generic_118855-36 i86pc i386 i86pc
```

```
[Connection Information]
Host: kedoo
User: emu
Group: Admin
Service: 20136
```

```
[Language Information]
Prompt Language: 0
Data Language: 0
Language List: 0;1
Language Delimiter: ;;;
```

- **Improved auditserver processing:** The EMu auditserver facility handles the processing of all audit trail information. If a large number of records is modified in a small period of time, the server may not be able to process records at the rate at which they are created. In this cast it falls behind and catches up when the system load is lighter. The auditserver has been optimised to allow for faster processing of audit trail records. It is about ten times faster than previous version. To take advantage of this improvement the EMu server must have the `XML::Parser::Expat` and `File::FcntlLock` perl packages installed.
- **Audit operation statistics gathered weekly and monthly:** The audit operations statistics (for every user in every table for every operation and the number of times the user has performed the operation) are now gathered on a weekly and monthly basis as well as the previous daily basis. The new statistics allow weekly and monthly reports to be produced detailing the operations performed by each user on all modules.
- **References tab added to Conservation Module:** A new *References* tab has been added to the Conservation module. The proposed change was discussed on emuusers.org and submitted for implementation. The tab allows users to enter as many references as necessary to support whatever Conservation activity is being recorded. It contains four fields:
 - **Reference:** Attachment field to Bibliography module.
 - **Kind:** Controlled by Lookup List (values to be determined by individual institutions). Would contain values that describe the kind of reference source, such as Background Info, Object Function.
 - **Page/Figure/Plate:** Text field. Would contain citation-level information such as fig. 1, p. 24. This field gives the user the option not to create a Citation record in the Bibliography module.

- **Caption/Text:** Text field. Allows the users to copy the cited text if appropriate.

Conservation (1) - Display

File Edit Select View Tools Tabs Multimedia Window Help

[C-2007-023] 1

Bibliography

Reference: [Journal] Crawford, Diane. Communications of the ACM.

Kind: Some Kind

Page/Figure/Plate: Page 12

Caption/Text: The text is here.

Bibliography Summary

	Reference	Kind	Page/Figure/Plate	Caption/Text
1	[Journal] Crawford, Diane. Communications of the ...	Some Kind	Page 12	The text is here.
2	[Book] Simpson, Peggy. The Life and Rhymes of D...	Another kind	Page 1	Some more text.
3		Third kind	Page 12	Lots of text.
*				

Analysis Accounting Recommend. Tasks References Notes Multimedia Other

Display Record 1 of 6 emu Admin 20136

Issues Resolved

Issue

If an existing Multimedia record has a new file attached to it with the same name as an existing file but with a different file extension, then both the old and new files will be removed when the record is saved.

The first time the Narrative or Summary tab is visited in the Narratives module the narrative text is displayed correctly. On the second and subsequent visits the narrative text is not displayed and the HTML editor cannot be activated.

The copying and pasting of records from List mode may copy empty virtual fields, even if they contain values, where some of the records selected do not contain attachments.

The **fifoserver** may generate an error where the data returned contains a format specifier (e.g. %n). In this case an empty value is returned rather than the correct value.

The calculation of the earliest and latest date ranges for circa dates where the date is BC and a month and day are specified is incorrect. For example, the date 2000BC/12/1 generates an earliest and latest date of -2000. The correct earliest and latest date is -2000/12/1.

The master version of an image may be fetched when running a report where the multimedia column in the report is in an attached record. For example, if a Loans report contains multimedia for each Catalogue record attached, the images displayed will use the master image, even if a smaller resolution is specified in the report definition.

The keyboard shortcut, CTRL+SHIFT+END does not select all records from the current record to the last record in List and Contact Sheet modes. Only the current and last records are selected.

If a record has multiple attachments on the Multimedia tab and one of the attached records does not have any media associated with it (that is, a Multimedia record without any file, URL or reference), the thumbnail of the previous attachment will be displayed for the empty attachment rather than a blank thumbnail.

Resolution

The old file is now removed and the new file kept in the Multimedia repository.

The narratives text is displayed correctly on all visits to the Narratives and Summary tabs. The HTML editor can be activated.

The contents of all virtual fields are now copied correctly when records are copied from List mode.

The correct data is returned from the **fifoserver** even when the data contains a format specifier.

The correct circa date ranges are generated for BC dates where a month and day are specified.

The "best fit" image based on the resolution specified in the report definition is now used rather than the master image.

All records from the current record to the last record are now selected.

A blank thumbnail is now displayed where an attached Multimedia record does not contain any file, URL or reference.

Issue

The attachment of an image containing invalid EXIF Interop data in the Multimedia module may result in the client hanging.

The Extended data is not displayed in the Groups module when thumbnails are displayed in Details mode.

The attach media drop down button on the Multimedia tab in the Multimedia module may appear as a single button rather than a drop down list containing File, URL and Reference for clients who have modified the standard Multimedia module (that is, sub-classed it).

The contents of a multi-column grid may lose their row associations where the grid contains more than one virtual column for the same attachment. The row associations are affected by the Move, Sort, Delete, Insert, Clear, Paste and Insert Paste operations available when right-clicking the grid. The issue may also occur when using drag and drop to rearrange rows within the grid.

The **auditserver** may produce duplicate audit records if the previous instance of the server was shutdown at an "inopportune" time while processing data. The problem only arises where the **auditserver** was processing records and the offset of the records processed was not saved correctly when terminated.

The dialogue box displayed when generating a crystal report that prompts for a parameter may display broken images for the OK and Cancel buttons. The only means of closing the dialogue box is to click on the X in the top left of the form.

The bibliography information for Bibliography records that have three or more links via *Published In* may not be updated correctly when one of the link records is modified. For example, if a citation record links to an article record which in turn links to a Journal record and the journal information is modified, the article record is updated correctly, however the journal information in the citation record may not be updated.

The error **Column operation performed before row has been accessed** may appear when deleting a Multimedia record from the Multimedia repository.

Resolution

The EXIF parsing has had more checks added to ensure that invalid EXIF tags do not cause the client to hang.

The Extended data is now displayed correctly.

The attach media drop down button now displays correctly.

The row associations are maintained correctly for all row based operations.

Duplicate records are no longer produced.

The OK and Cancel buttons now display correctly.

All bibliography information is updated correctly regardless of the number of links in the record.

The error no longer appears when deleting a record.

Issue

The Conservation tab in both the Loans and Events modules may not display a list of all conservation records associated with the current Loans or Events record. The problem occurs for clients who use the "new" version of the Conservation module.

A **Grid Range Index** error may appear when dragging and dropping rows in a grid where the dragged row is empty and it is dropped on the Insertion row at the bottom of the grid.

The error **Column operation performed before row has been accessed** may appear after sorting records in either List or Contact Sheet modes. The error occurs sporadically.

The last record displayed in List mode may be duplicated in the grid. The records retrieved are correct but not displayed correctly. The issue only arises if a false match is found in the matching records (very rare).

An **Access Violation** error message may appear when selecting a Thesaurus record where Valid is set to *No*. The error appears after the dialogue box displaying **The term *term* is not valid for selection and so is being discarded** is dismissed.

The username and service displayed in the status bar at the bottom of the client form may appear centred in the bar rather than on the right hand side. The issue only occurs if Shortcuts is enabled.

The form window may appear truncated on the right when opening a module that was last viewed maximised and the *Save Last Position* and *Save Last Size* options are enabled.

The Modified time on the Reports Properties dialogue box may be an hour ahead of the current time. The issue only occurs when the client computer does not have daylight savings enabled.

The red background colour used by the Column Colour Registry entry, e.g.:
Group|Default|Table|eparties|Column Colour|NamLast|red
will appear as a white background on screens with 16 bits or less colour depth.

Resolution

All associated conservation records are displayed correctly in Loans and Events records.

The error no longer appears when dragging empty rows in a grid to the Insertion row.

The error no longer appears after sorting records.

The correct data is displayed for all records in List mode, even if false matches occur.

The error message no longer appears when selecting invalid Thesaurus records.

The username and service are always displayed on the right hand side of the status bar.

The module form is now displayed correctly.

The correct Modified time is displayed regardless of the daylight setting on the client computer.

The red colour has been adjusted to still appear red on screens with 16 bits or less colour depth.

Issue

The Caption in a module form may have duplicate mode indicators appended to the module name, e.g.

Parties (1) - Search (1) - Search

The issue arises when switching between Prompt languages where the Caption for the module form has not been translated.

The default Page View displays all prompts in English regardless of the Prompt language selected.

The **auditserver** may produce duplicate audit records if the previous instance was terminated while copying audit XML records. The copying only occurs after all existing records are processed, so the issue arises rarely.

The Security, Audit and Admin tabs may not be displayed in the correct order for clients who use a modified version of the Multimedia module.

If a record containing calculated values (e.g. feet to metres, etc.) is dittoed into a new record and the calculated values are cleared, the Orig flag (which is used to determine which value is the one entered by hand and therefore which values are calculated) is not cleared when the record is saved.

The font used to display data may not be set correctly where the selected font does not include a character for the current code page. In particular single value fields may exhibit the problem.

A deleted record may reappear in the Catalogue and Taxonomy modules if the background loads were not running when the deletion occurred and outstanding updates are required to the deleted record.

The error **Column operation performed before row has been accessed** may occur if the first matching record is discarded while in Details mode. The same error may occur if you choose to delete the record.

The weekly and monthly statistics generated for operation by table by user may produce incorrect values for the number of operations performed.

Resolution

The correct module form Caption is now displayed.

Language specific prompts are now displayed based on the Prompt language selected.

Duplicate audit records are no longer produced if **auditserver** is terminated while copying audit XML records.

The Security, Audit and Admin tabs are now displayed in the correct order.

The Orig flag is now cleared if all values for calculated fields are removed.

Only fonts containing characters for the current code page may be selected.

The background loads have been altered to ignore updates to deleted records.

The error no longer occurs when discarding / deleting the first record in Details mode.

The correct number of operations is now computed and stored.

Issue

The **emuregdelete** utility does not delete entries containing the owner of the Registry entry correctly (e.g. `emu\System\Paths\ServerMediaPath\`). The inclusion of the owner is intended to allow the deletion to be limited to entries owned by a given userid.

The time taken to move between matching records in Details mode may deteriorate the longer a module window is used. The issue occurs if the window is cached and is used often. Each use of the cached form results in slower times moving between records, particularly if the module has a lot of tab switching involved (e.g. Catalogue module).

The View Attachments button displayed next to a grid control may not be enabled where a grid contains a single attachment. The issue only arises if the attachment list is computed by the client, rather than being stored in the record (rare).

The weekly maintenance may fail on modules with a small number of records where the size of the record contents vary greatly (e.g. Groups module). Only the affected module(s) are left off-line, all other modules are operational. The issue was caused by an incorrect configuration being generated.

Collection Events and Site records containing invalid latitude and / or longitude values may display an invalid value error message even after the value has been corrected. The issue only affects natural history clients who upgraded to the common Lat / Long tab where records already contained invalid values (due to the initial data migration).

The Close button on the Summary dialogue box, displayed after sorting records, may not close the form. The X button in the top right hand corner of the form may be used successfully.

The tool bar may not have the correct length when Shortcuts are enabled and the *Save Last Size* option is disabled. The right hand end of the tool bar will be truncated.

If saving a record fails because a mandatory field has not been filled, the error message form may be hidden behind the module form. The issue only arises if the module switched tabs to focus on the field that is not filled. In this case, the module form cannot be selected as the hidden error message has focus. It appears the client has frozen.

Resolution

The correct Registry entries are deleted based on the userid supplied.

The time required to move between records in Details mode no longer deteriorates for cached module forms.

The View Attachment button is now enabled for a single attachment in a grid control.

The weekly maintenance now completes correctly for modules where the record size varies greatly.

The record may now be saved when the invalid value is corrected.

The Close button now functions correctly.

The tool bar now displays correctly.

The mandatory error message is now displayed on top of the module form, even if tabs are switched to display the offending field.

Issue

The conversion of longitude values between DMS (Degrees / Minutes / Seconds), DM (Degrees / Minutes) and decimal may not occur in the Collection Events module when importing data via the Import facility or via **texload**.

Some video and audio file formats may not be playable within the EMu client on Windows Vista and Windows 7. Launching an external viewer will result in the media playing correctly.

Audit Trail records may be duplicated for clients who installed a pre-release version of EMu 4.0.01 and then upgraded to the official release at a later date. The duplicates occur as the pre-release version installed an audit processing plugin and the official release installed another audit processing plugin. As the pre-release plugin was not removed, each audit record was processed twice.

The Summary dialogue box may be hidden behind the sorted module where a sort was performed and a summary requested while in Details mode.

The Reports Properties dialogue box may not display the name of the report file, its size and modified time.

The Import Identifier and System Identifier fields are dittoed when either the ditto all or ditto tab command is selected. As these fields relate to the creation of the record via the Import facility they should not be dittoed.

In very rare circumstances an active form may become hidden behind an inactive form (e.g. the Save form hidden behind the module form). The issue only occurs if EMu is made the active application (via ALT+TAB) while a form was being created.

A user may be able to see records that should be hidden by Record Level Security (RLS). The issue only occurs where the base security system has been augmented with column specific values via the Security Registry entry, e.g.:

Group|Molluscs|

Table|ecollectionevents|Security|Display|SecDepartment=Molluscs

The same issue may cause the Summary of sorted records to be slow.

Resolution

The conversion of longitude values always occurs when using the Import facility or **texload**.

All video and audit file formats registered with Vista and Windows 7 now play correctly.

The pre-release plugin is removed as part of the EMu 4.0.02 installation.

The Summary dialogue box is placed on top of the sorted module when displayed.

The correct Report Properties information is now displayed.

The Import Identifier and System Identifier are no longer dittoed.

The active form is always placed on top of the inactive form.

Users can no longer view hidden records where the base security system has been adjusted.

Issue

The thumbnail image displayed in Details mode when view Thumbnail is enabled may not appear on the far right side of the form. The Issue only arises if **Save Last Size** and **Save Last Position** are enabled and the module form is not the default size.

Multimedia files larger than 2 Gb are not handled correctly. Any file over 2 Gb will result in a zero sized media file being produced.

A **List out of bounds** error may occur when deleting a row from the Latitude / Longitude List grid. The issue only occurs if none of the existing Latitude / Longitude entries has Preferred set to *Yes*.

An **Access Violation** error may be listed in the Import report when importing records into the Taxonomy module. The affected records are not imported.

If a user double clicks on a group name while using the Group Retrieve dialogue box, all records in the module are retrieved, rather than the records in the group. The issue only arises for static groups.

The Import facility may create multiple versions of the same attachment record where the data contains non-ASCII characters. The issue only occurs for clients who have their base EMu language encoding set to UTF-8 (that is `langcode=utf8`).

Resolution

The thumbnail image is always displayed on the right hand side of the form.

Multimedia files over 2 Gb are now handled correctly.

The error message no longer appears.

The error message no longer appears and the records are created correctly.

Double clicking on a static group now retrieves the record in the group only.

A single version of the same attachment record is created where the data contains non-ASCII characters.

Issue

The error message **AlsoSearch column *column* does not exist** may appear when defining an AlsoSearch Registry entry where the columns to search are virtual columns that refer to the initial AlsoSearch column.

For example, the Registry entry:

```
Group|Default|Table|ethesaurus|Also Search|TerTerm|  
RelRelatedTerms
```

indicates the column *RelRelatedTerms* should also be searched when *TerTerm* is searched. The *RelRelatedTerms* column is a virtual column pointing to the *TerTerm* column in the Thesaurus module. So when *RelRelatedTerms* is searched it looks up *TerTerms*, which causes *RelRelatedTerms* to be also searched again. EMu detects the loop and displays the error message.

If a number of records are attached to a control in Query mode (that is, <#nnn#> is displayed in the control), the attached records cannot be cleared by using the DEL key or backspacing over the control contents.

The *Publish on Internet* and *Publish on Intranet* fields have associated flag fields used by some web based searches to check whether a record may be viewed on the web. In some instances the flag fields may be out of sync with the *Publish on* fields (that is, the field has a value of *Yes* and the flag is set to *N*). The field and its associated flag value should always be in sync.

The **emulogger** facility allows the standard configuration file (`etc/logger`) to be replaced with a localised version in `etc/local/logger`. A better mechanism would allow the local configuration to override entries in the base file rather than replace the configuration file completely. Such a set up provides a useful means of adding new services without the need to update the local configuration file.

The Field Help module does not contain records for reference columns, that is column names ending in `Ref`, `Ref_tab` and `Ref_nesttab`. These columns are not accessible from within the EMu client, however the columns may be used for other purposes (e.g. web page design).

Resolution

The error message is no longer displayed when the Also Search loop is detected.

The DEL and backspace key can now be used to clear attached records in Query mode.

Checks have been added so that when a record is saved the *Publish On* fields and their associated flag values are synchronised to ensure they always reflect the same state.

The local configuration file now augments the entries in the base configuration file, rather than replacing them.

Reference fields have been added to the Field Help module.

Issue

The Summary and Extended data calculated for the Groups module does not generate values suitable for multilingual systems.

An error message is displayed when invalid characters are found in EXIF data fields, forcing the EXIF data to be discarded. In most cases it would be better to skip the invalid character(s) and insert the EXIF data.

A very large hierarchy (e.g. the taxonomic hierarchy) may cause Lookup List searches to deteriorate when retrieving values from the hierarchy. The issue only occurs where the hierarchy contains many levels with a large number of terms at each level.

The Image Display Registry entry only looks for images stored in the `etc/images` directory on the EMu server. The `local/etc/images` directory should be checked first to allow sites to provide localised images.

If a number of TWAIN devices are registered on the EMu client machine, moving from record to record in the Multimedia module may be slow. Each twain device is queried as each record is displayed causing a long delay if multiple devices are registered.

The Open and Save buttons on the Reports Properties dialogue box may cause confusion as to their purpose. The buttons should be renamed to better indicate their functionality.

Many modules contain date and date ranges in their Summary and Extended data calculations. The formatting of the date(s) depends on the date order (DMY, MDY, YMD) used by the EMu server, however in some instances partial dates may not format correctly causing "invalid" dates to be displayed.

Resolution

The Summary and Extended data generated is now multilingual.

Invalid characters in EXIF data are converted to a '?' character allowing the data to be extracted rather than discarded.

The Lookup List values for hierarchies are clustered together eliminating the deterioration in search times.

The `local/etc/images` directory is checked before `etc/images` for images.

The Multimedia module now checks for the existence of TWAIN devices when the module is invoked rather than between each record displayed.

The Open button has been renamed to Download and the Save button renamed to Upload. The Download dialogue box title has been changed to *Download Report to Edit* and the Upload dialogue box title changed to *Update Report to EMu*.

Partial dates are not formatted correctly.

Upgrade Notes

The upgrade from EMu Version 4.0.01 to EMu 4.0.02 involves a number of steps. Please follow the instructions below carefully.

Do not skip any steps under any circumstances.

Before proceeding with the update please ensure that a complete backup of the EMu server exists and is restorable.

The upgrade may take some time, particularly on systems with a large number of records.

1. Install Texpress 8.2.008 or greater.
2. Install TexAPI 6.0.001 or greater.
3. Log in as **emu**.
4. The following steps need to be repeated for each client installed on the EMu server.
Please ensure that no one is using the system while the upgrade is underway.
5. Enter `client client`
6. Enter `script /tmp/output-4-0-02`
A new shell will start and all output recorded until the shell is terminated.
7. Enter `upgrade-4-0-02`
8. Enter `vi ~/luts/default/System.txt`
Check each System entry has its correct translation then save the file.
9. Enter `upgrade-4-0-02.geoorig` (*clears **Orig** flag for empty calculated fields*)
10. Enter `upgrade-4-0-02.georef` (*adjusts centroid latitude / longitude precisions*)
11. Enter `upgrade-4-0-02.statistics` (*adjusts incorrect weekly / monthly statistics*)
12. Enter `upgrade-4-0-02.webflags` (*corrects **Publish In Internet / Intranet** flags*)
13. Enter `emureindex`
14. Enter `emulutsrebuild`
15. Enter `exit`
The script session will terminate.
16. Enter `EDITOR=vi crontab -e`
Add the following entry:

```
#
# Run Scheduled Exports
#
0 6 * * * ~emu/clientname/bin/emurun emuexport 2>&1 |
~emu/clientname/bin/emurun emulogger -t "KE EMu Scheduled
Exports Report" -z exports
```

All cron commands must be entered on one line (even if they appear wrapped on the screen)! The start times of command may need to be varied to fit in with existing maintenance jobs.

17. Enter `vi ~/data/eaudit/emuoptions`
Add the following line to the file then save the file:

```
COMPACT="no"
```

18. Enter `vi ~/data/estatistics/emuoptions`
Add the following line to the file then save the file:

```
COMPACT="no"
```

19. Go back to step 5 to commence upgrading the next client, until all clients are complete.

EMu 4.0.02 does not require the new Windows client to be installed on every machine for network installations. Updating the network server is sufficient. For standalone installations a new client is required on each machine.



EMu Documentation

Right-to-Left Language Support in EMu

Document Version 1.0

EMu Version 4.0



Contents

SECTION 1	Overview	1
SECTION 2	Switching Languages	5
SECTION 3	HTML Editor Extensions	7
SECTION 4	Unicode Data Controls	9
SECTION 5	Installation of RTL Languages	11
	Windows 2000	11
	Windows XP	11
	Windows 2003 (Server)	11
	Windows Vista and Windows 7	11
SECTION 6	Data Entry for RTL Languages	13
	Windows 2000	13
	Windows XP	13
	Windows 2003 (Server)	14
	Windows Vista and Windows 7	14
SECTION 7	Index	15

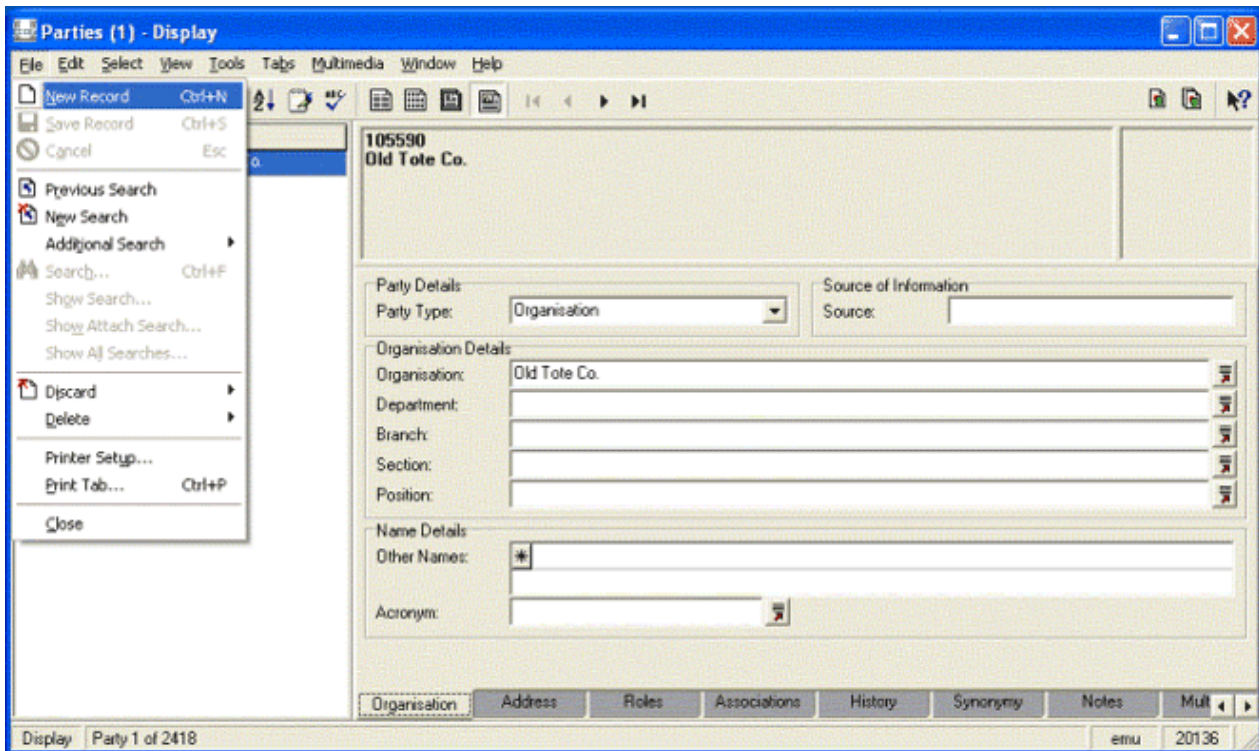
SECTION 1

Overview

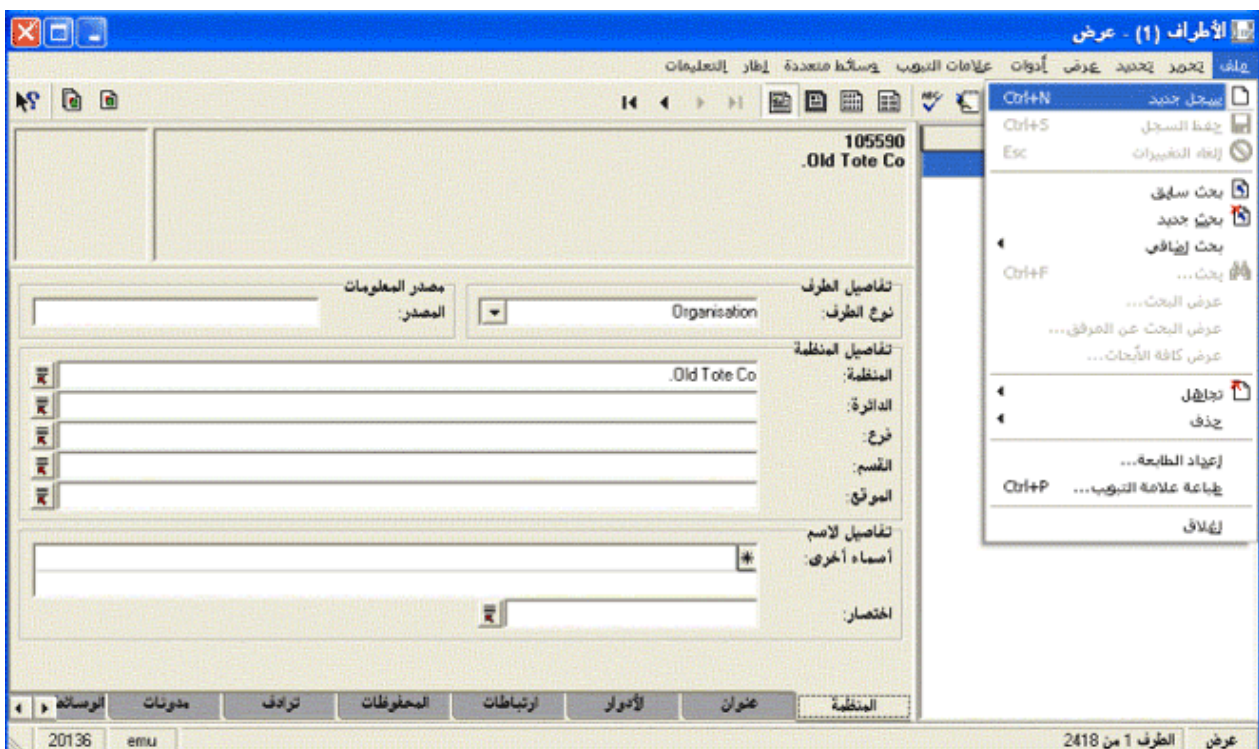
EMu 4.0.02 introduces enhanced support for languages, including right-to-left (RTL) languages. Arabic and Hebrew scripts are now supported, as are two additional left-to-right (LTR) languages, Greek and French (Canadian). Thirteen languages are now supported in EMu:

- English & English (US)
- Arabic
- Danish
- Dutch
- French & French (CA)
- German
- Greek
- Hebrew
- Italian
- Norwegian
- Polish
- Spanish
- Swedish

Switching between LTR and RTL languages is dynamic and does not require EMu to be restarted - the user interface automatically adjusts to suit the language selected. For example, selecting Arabic will result in the Menu bar moving to the right side of the module window and all controls being mirrored. The image below shows the Person tab of the Parties module with Shortcuts enabled:



This image shows the same tab with Arabic selected:



As you can see, the layout of the various controls is mirrored, even down to the module Title bar. Many of the images used in EMu (e.g. on buttons, menus, etc.) have been altered to suit RTL languages. In some cases a new image has been constructed to ensure an authentic user experience. For example, the *What's this help?* image contains a reversed question mark in the Arabic version as this is how question marks are displayed in Arabic.

All the data controls in EMu 4.0.02 have been replaced with Unicode versions, which allows EMu to determine the character set used to store information and adjust the display to show characters correctly. Whether you use UTF-8 or ISO-8859-6 (Arabic) as your data storage encoding, the display will adjust to show the characters correctly.



It may be necessary to install additional fonts on your computer to ensure that all characters can be displayed (e.g. Windows 2000 and Windows XP require the installation of right-to-left language files in order for Arabic and Hebrew characters to be shown).

A number of controls have been adjusted to show RTL data correctly. In particular the HTML editor provided with EMu (in the Field Help and Narratives modules) has been extended to support RTL orientation.

SECTION 2

Switching Languages

Two language settings can be altered in EMu:

- The first is the language used for Prompts, e.g. prompts, menus, forms, etc. Prompts in any EMu client can be switched to one of the thirteen supported languages.
- The second is the language used for Data.
Data can be added in one or more languages, and it is possible to select which language or languages are used to display the data. For systems with only one language it is obviously not possible to alter the language in which Data is displayed.

Which languages are used is set in the EMu Registry via:

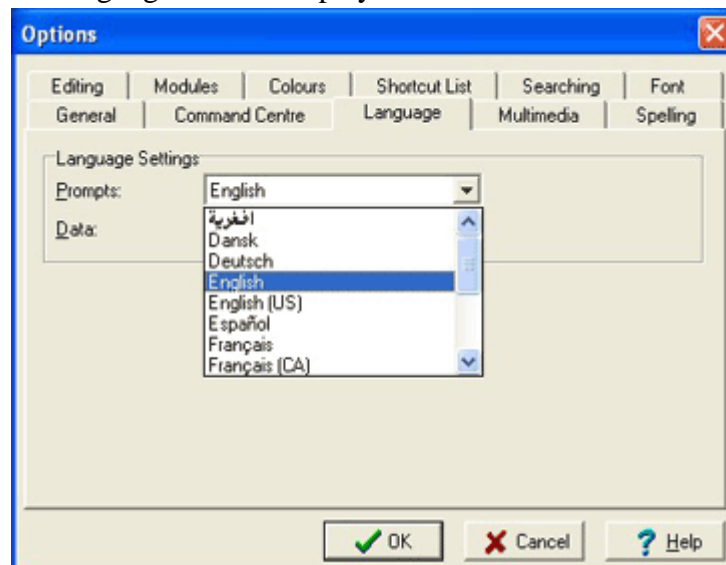
System\Setting\Language\Supported


For clients with multiple languages supported, it is possible to select any one language or all languages to display data.

EMu provides two mechanisms for switching the languages displayed in the user interface:

- The first mechanism allows the language used for Prompts and Data to be set independently.

This is achieved using the Language tab in the Options dialogue box (select **Tools>Options** from the Menu bar). As the names suggest, the *Prompts* drop list specifies the language used for prompts, and the *Data* drop list specifies the language used to display data:



When changing the language of Prompts, EMu adjusts the user interface to the language selected as soon as the **OK**  button is clicked. When switching between LTR and RTL languages a small delay may occur while the user interface is adjusted.

When switching the language of Data, only the data displayed in controls is affected. The layout of the user interface and language of Prompts is not adjusted.

- The second mechanism for switching the languages displayed in the user interface allows both the language of Prompts and Data to be changed at the same time.

This is achieved by selecting **Tools>Language** from the Menu bar. For systems configured to allow more than one language for data, a **Tools>Language** sub-menu lists each language and an entry for **All Languages**.

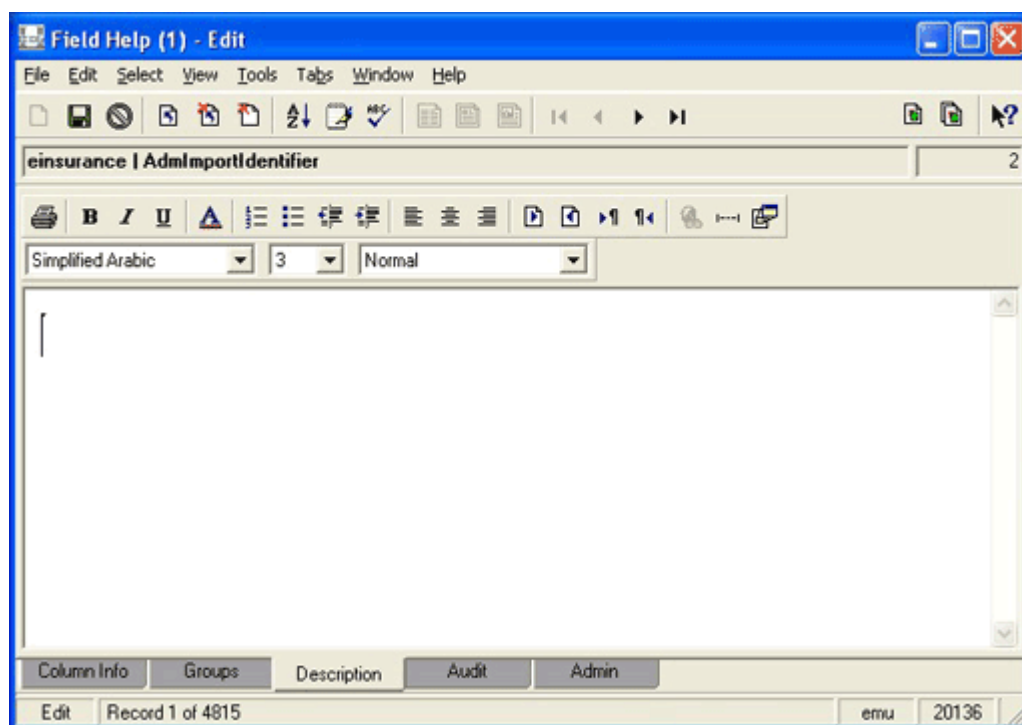
Selecting one of the menu options changes both the language of Prompts and Data at the same time.

If **All Languages** is selected, the language used for Prompts is not changed, but all languages used for Data are displayed separated by the language delimiter (normally ; : ;):





SECTION 3

HTML Editor Extensions

In order to provide full support for RTL languages, the standard EMu HTML editor has been extended to allow the page and paragraph orientation to be set:



The new buttons are:

Button	Description
	The document direction is left-to-right (same as <code><html dir="ltr"></code>). If a direction is not specified, the default direction is <code>ltr</code> .
	The document direction is right-to-left (same as <code><html dir="rtl"></code>). An <code>rtl</code> document has the scroll bar located on the left side of the window with all text right aligned by default. The standard Unicode <i>bidi</i> algorithm is used to display text. That is, English text is displayed in a left-to-right order while Arabic / Hebrew is displayed right-to-left. The setting affects the default direction of all text in the document.
	The text direction for the current paragraph is set to <code>ltr</code> (same as <code><p dir="ltr"></code>). If the document direction is <code>rtl</code> , this command will reset the direction to <code>ltr</code> for the current paragraph only. Note that Arabic text will still be displayed right-to-left but left alignment will be used. This is useful for embedding an English quote in Arabic text.
	The text direction for the current paragraph is set to <code>rtl</code> (same as <code><p dir="rtl"></code>). This is useful for embedding an Arabic quote in English text.

The RTL language features added to the HTML editor are available for all HTML based fields in EMu.

SECTION 4

Unicode Data Controls

To support RTL languages in EMu, all data controls (e.g. RichEdit, Grid and Combobox controls) have been upgraded to support Unicode. All other user interface controls (e.g. menus, tabs, etc.) have not changed and still use single byte based character sets. The change to data controls means any Unicode character may be entered into a data field and it will be displayed and saved correctly. If the EMu server encoding is set to `utf8`, the data will also appear correctly in reports and on the Internet.

For clients who do not have the EMu server encoding set to `utf8`, a mapping takes place where all data characters are translated from the server encoding that has been used into UTF-8 before being displayed, thus causing the correct characters to be shown. A reverse mapping takes place for all Unicode data entered into controls when saved to the EMu server.

When the EMu client is invoked it queries the EMu server to determine the encoding used. Once the client knows the encoding, it performs all the necessary translations to ensure the data is displayed and saved correctly.

As non-data controls have not been upgraded to their Unicode equivalent they will display text (in most cases prompts) using the character set defined by the Prompts language setting. In rare instances this may result in the incorrect display of individual characters (for example, selecting French as the language for Prompts and trying to show an Arabic character). In general this should not occur as all prompt based strings are translated using the character set for the language specified for Prompts.

SECTION 5

Installation of RTL Languages

In order to display RTL based characters correctly and to install the *bidi* algorithm used for data entry and display, it may be necessary to add RTL language files to your Windows installation. Details on how to install the RTL language files are given below for each version of Windows:

Windows 2000

1. Select **Start>Settings>Control Panel>Regional Options**.
2. On the General tab, select the language to install (e.g. **Arabic**) in the *Language settings for the system* section.
3. Click **Apply**.
4. If prompted, insert the Windows 2000 CD-ROM and click **OK**.
5. Restart your computer when prompted. If you are not prompted, select **Start>Shut Down>Restart**.

Windows XP

1. Select **Start>Control Panel>Regional and Language Options**.
2. On the Languages tab, click the check box beside *Install files for complex script and right-to-left languages (including Thai)*.
3. Click **Apply** and then **OK**.
4. Restart your computer when prompted. If you are not prompted, select **Start>Shut Down>Restart**.

Windows 2003 (Server)

1. Select **Start>Control Panel>Regional and Language Options**.
2. On the Languages tab, click the check box beside *Install files for complex script and right-to-left languages (including Thai)*.
3. Click **Apply** and then **OK**.
4. Restart your computer when prompted. If you are not prompted, select **Start>Shut Down>Restart**.

Windows Vista and Windows 7

Windows Vista and Windows 7 have RTL language fonts installed by default and there is no need to add any language files.

SECTION 6

Data Entry for RTL Languages

Once the correct language files are installed, it may be necessary to configure Windows to allow data entry in multiple languages. This involves adding a keyboard layout for the language in which you wish to enter data.

For example, to type Arabic characters it is necessary to configure the keyboard to mimic an Arabic keyboard. As each version of Windows provides a different mechanism for adding keyboard layouts, details are provided for each release:

Windows 2000

1. Select **Start>Settings>Control Panel>Regional Options**.
2. Select the **Input Locales** tab.
3. Click **Add** if the language in which you wish to type is not listed under *Installed input locales*.
4. In the Add Input Locale dialogue box, select the language in which you wish to type from the Input Locale list. This automatically selects the Keyboard Layout/IME option.
5. Click **OK**.
6. In the Regional Options dialogue box, click **Apply** and then **OK**.

Windows XP

1. Select **Start>Control Panel>Regional and Language Options**.
2. Select the **Languages** tab.
3. Click **Details** in the *Text services and input languages* section. The Text services and input languages dialogue box appears.
4. If the language in which you wish to type is not listed under *Installed Services*, or if the language that you want is listed but the keyboard is not listed with it, click **Add**.
5. In the Input Language list, select the language in which you wish to type. This automatically populates the Keyboard layout/IME field.
6. Click **OK**. The selected language with a keyboard installed displays under *Installed Services*.
7. Click **OK** to close the Text Services and Input Languages dialogue box, and then click **OK** to close the Regional and Language Options dialogue box.

Windows 2003 (Server)

1. Select **Start>Control Panel>Regional and Language Options**.
2. On the Languages tab, click **Details**.
3. If the language in which you wish to type is not listed under *Installed Services*, or if the language that you want is listed but the keyboard is not listed with it, click **Add**.
4. In the Input Language list, select the language in which you wish to type. The Keyboard layout/IME field is automatically populated for you.
5. Click **OK**. The selected language with a keyboard installed displays under *Installed Services*.
6. Click **OK** to close the Text Services and Input Languages dialogue box, and then click **OK** to close the Regional and Language Options dialogue box.

Windows Vista and Windows 7

1. Select **Start>Control Panel>Regional and Language Options**.
2. On the **Keyboards and Languages** tab, click **Change keyboards**.
3. Under *Installed services*, click **Add**.
4. Double-click the language you want to add.
5. Double-click the text services you want to add.
6. Select the text services options you want to add.
7. Click **OK**.

After keyboard layouts have been added, it is possible to select the input language from the Language bar, which is situated in the System Tray (it is a button with the two letter name for the current input language displayed). Click the **Language** bar for a list of all possible data entry languages. Select the input language required.

Index

D

Data Entry for RTL Languages • 13

H

HTML Editor Extensions • 7

I

Installation of RTL Languages • 11

O

Overview • 1

S

Switching Languages • 5

U

Unicode Data Controls • 9

W

Windows 2000 • 11, 13

Windows 2003 (Server) • 11, 14

Windows Vista and Windows 7 • 11, 14

Windows XP • 11, 13



EMu Documentation

Scheduled Exports

Document Version 1.1

EMu Version 4.0.02



Contents

SECTION 1	Overview	1
SECTION 2	Creating a Scheduled Export	3
	Set up the Export schedule: the Export tab	4
	Select the fields to include in the export: the Fields tab	8
	Determine which records to export: the Conditions tab	10
	Hint - Determining the TexQL Conditions statement	12
	Define sort order: the Sort Order tab	14
	Set sort options: the Options tab	15
	Set security options: the Security tab	16
	Examples	18
	Scenario 1	18
	Scenario 2	20
	Scenario 3	22
	Scenario 4	24
SECTION 3	Unscheduled Exports	27
	Running Unscheduled Exports	28
SECTION 4	Viewing Export Results	29
	View Export Results	30
	View Export Data	32
	Save all Export Data files	32
	Save an Export Data file	32
SECTION 5	The Schedules and Exports Modules	33
SECTION 6	Creating Filters	37
	Where are filters stored?	37
	How is a filter invoked?	38
	Filter Output and Errors	38
	Filtering Data	39
	An example filter	40
SECTION 7	Cron setup	43
SECTION 8	Using emuexport	45
	Index	47

SECTION 1

Overview



To be able to use the Export Schedules facility, a user must have (or be a member of a group that has) the `daExport` operations permission.

EMu 4.0.02 introduces a new facility that enables data to be exported on a regular basis. Users can define:

- A set of fields for export.
- A sort order for the exported records.
- A TexQL statement to determine which records to export.
- Dates on which the export is to occur.

Data can be scheduled to be exported on anything from a daily to an annual basis. The export occurs outside working hours (generally early in the morning) and the results are stored on the EMu server. A user may then view the results and save / view the export files produced.

The Scheduled Exports facility provides two new modules:

- **Schedules**



Holds one record per Scheduled Export containing the definition of the export (as defined above).

- **Exports**



Holds one record per set of export files produced. When a Scheduled Export is run a new exports record is created containing the output of the export process.

See The Schedules and Exports Modules (page 33) for more details.

An audit of all Scheduled Exports is produced and these records are not removed from the system, so it is possible to search / view all exports performed by all users (provided that you have sufficient privileges to do so).

When a Scheduled Export is executed, the default output is XML, which is the same format used by the EMu reporting facility. A filter may be set to transform the XML produced into another format (CSV for example). System Administrators may define their own filters to produce custom output formats. A custom filter may also be used to:

- Email export files to users.
- Print export files.
- Copy export files to a known location (for collection).
- Transmit export files to another machine.

See Creating Filters (page 37) for more details on how to create custom filters.

SECTION 2

Creating a Scheduled Export

Configuring a Scheduled Export is very similar to configuring a report. In fact the Properties dialogue box used to select the fields for export is the same as used by the reporting process. In this section we step through the creation of a new Scheduled Export.

Set up the Export schedule: the Export tab

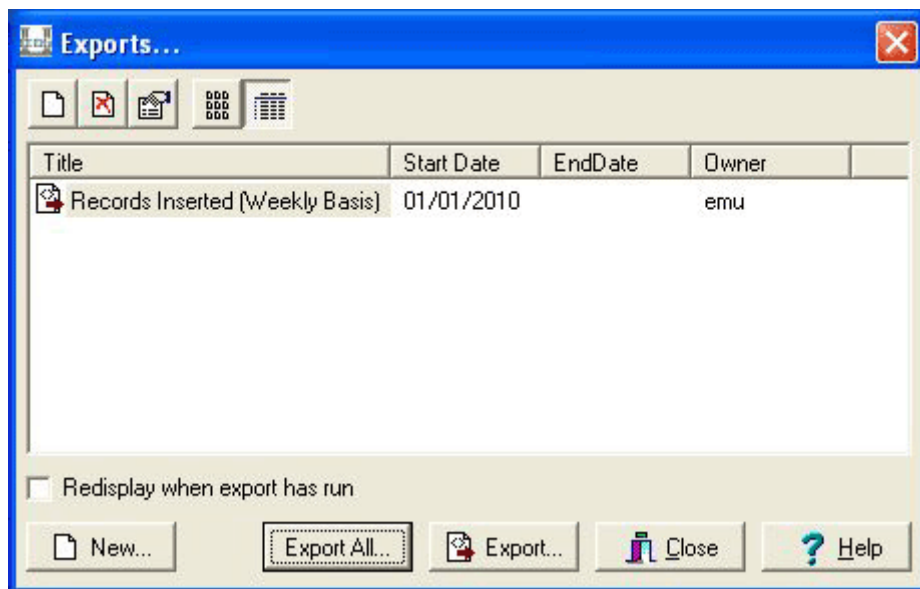
In the module from which the records are to be exported:

1. Search for or otherwise list a group of records.
2. Select **Tools>Exports** in the Menu bar.

-OR-

Use the keyboard shortcut, ALT+T+X.

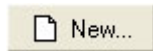
The Exports dialogue box displays with a list of all Scheduled Exports for the current module:



3. Select **New**

-OR-

Select **New**



The Export Properties dialogue box displays:

4. Enter a descriptive name for the Export Schedule in the top text field.
The title will display in the Exports dialogue box, from where it can be selected by authorised users to run the export on an adhoc basis.
5. If you require the exported data to be transformed from XML format to another format (e.g. CSV), select a filter from the *Filter (optional)* drop list. Filters are defined by the System Administrator. If a filter is not selected, the exported data is saved in EMu XML format. See Creating Filters (page 37) for more details.
6. Select the language of the data to be exported from the *Language* drop list. A language may only be selected if your EMu system supports more than one data language. Any of the available languages in which data is recorded may be selected or *All Languages* may be used to export the complete contents of the exported fields.
7. Enter the first date on which the export will be produced in the *Start Date* field.
This date is also used to calculate the next time an export should occur when the report frequency is specified as a number of days / months / years.
The format of the date (e.g. dd/mm/yyyy) is as specified for your EMu system.
A start date must be provided.
8. Enter an end date in the *Finish Date* field.



An end date is not required, however if specified, data will not be exported after that date has been reached. An empty end date implies no final date for exporting data.

9. Select an export frequency from the *Frequency* drop list.
There are four frequency values to choose from:

Adhoc Use **Adhoc** when defining an export that will not be run at a regular interval, for example if you are contacted irregularly to produce the export data.

As the export is not scheduled, it must be run manually to produce the export data. See *Unscheduled Exports* (page 27) for more details.

Daily Use **Daily** if the data is exported on anything less than a monthly basis. The following options are displayed when **Daily** is selected:

There are three ways to configure daily exports:

1. The simplest is to select the days of the week on which the export is to occur. The export is run early in the morning of the day(s) selected.
If no Finish Date is specified, the export will run every week on the specified day(s).
Typically an export would be run once per week or every day.
If there is an interval between exports (the export is scheduled for Sunday and Thursday, for instance), the records exported on Sunday would be from the previous Thursday, and the records exported on the Thursday would be from the previous Sunday.
2. Specify the number of days between exports (starting from the Start Date) in the *Every nnn Day(s)* field.
For example, a value of 14 indicates that the export is to be run fortnightly from the Start Date.
3. Specify the day on which an export will first be run and the interval between exports.
For example, a value of *Every 14 Day(s)*, with *Thursday* selected indicates that an export will commence on the first Thursday after or on the Start Date and will run every 14 days.

Monthly Use **Monthly** when data exports occur on a monthly basis but less than annually. The following options are displayed when Monthly is selected:

Frequency: Monthly

Every Month(s)

Day of the Month

☐ First Weekday of the Month

Specify how many months must elapse between each report using the *Every nnn Month(s)* field. If a value is not given, the export will occur monthly. For example, a value of 3 indicates that the export will occur quarterly.

Use the *Day nnn of the Month* field to specify on what day in the month the export is to occur. A value of 1 indicates that the export will run on the first day of the month. A value of 31 implies the last day of the month, regardless of how many days are in the month.

The *First Weekday of the Month* checkbox may be selected instead of specifying a day of the month. In this case, the export is run early in the morning on the first Monday of the month.

A value must be supplied for either *Every nnn Month(s)* or the *First Weekday of the Month*.

Annually Use **Annually** for any data export required on a yearly basis. The following options are displayed when Annually is selected:

Frequency: Annually

Day of the Year

Day and Month of Each Year

☐ First Weekday of the Year


There are three ways to configure annual exports:

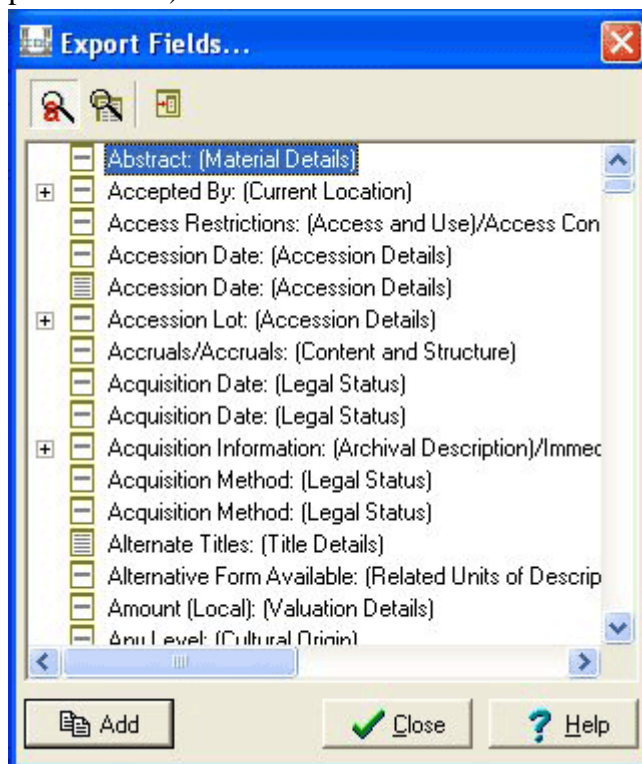
1. Specify the day of the year on which the export is to take place using the *Day nnn of the Year* field. A value between 1 and 365 can be used.
2. Specify a specific day in a given month using the *Day nnn and Month mmm of Each Year* field.
For example, a setting of *Day 1 and Month July of Each Year* would allow data up to and including 30 June to be exported.
3. The *First Weekday of the Year* checkbox may be selected instead of specifying a day within a month. In this case, the export is executed early in the morning of the first Monday of the Year.

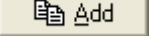
Select the fields to include in the export: the Fields tab

1. In the Export Properties dialogue box, select the **Fields** tab.
Here we add the fields to be exported.

2. Select **Add** .

The Export Fields box displays with a list of all fields in the current module, as well as fields in attached modules (an attachment field is indicated by the plus  icon):




3. In the Export Fields dialogue box, select a field and select **Add** .

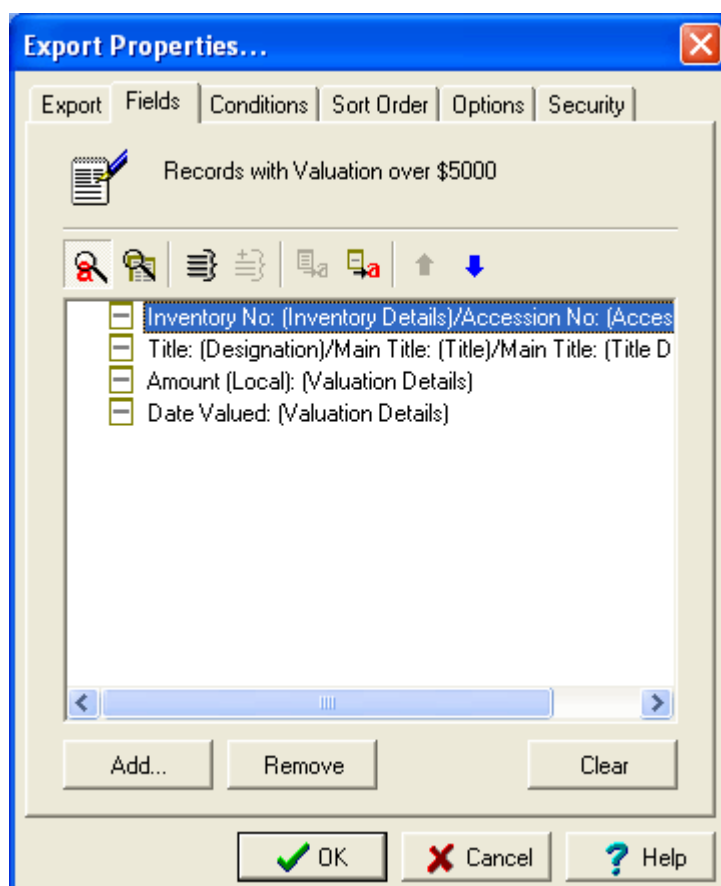
-OR-

Double-click a field.

The field is added to the Fields list in the Export Properties box.

4. When all required fields are added, select **Close**  in the Export Fields dialogue box.

The Export Properties dialogue box now lists all the fields that will be included in the export:



For more details regarding the selection of fields for export see *Select the fields to include in the report: the Fields tab* in the Reports section of the EMu help.

Determine which records to export: the Conditions tab

1. In the Export Properties dialogue box, select the **Conditions** tab.
On the Conditions tab we add a TexQL statement to determine which records will be exported.

2. Enter a TexQL statement into the Conditions text box.

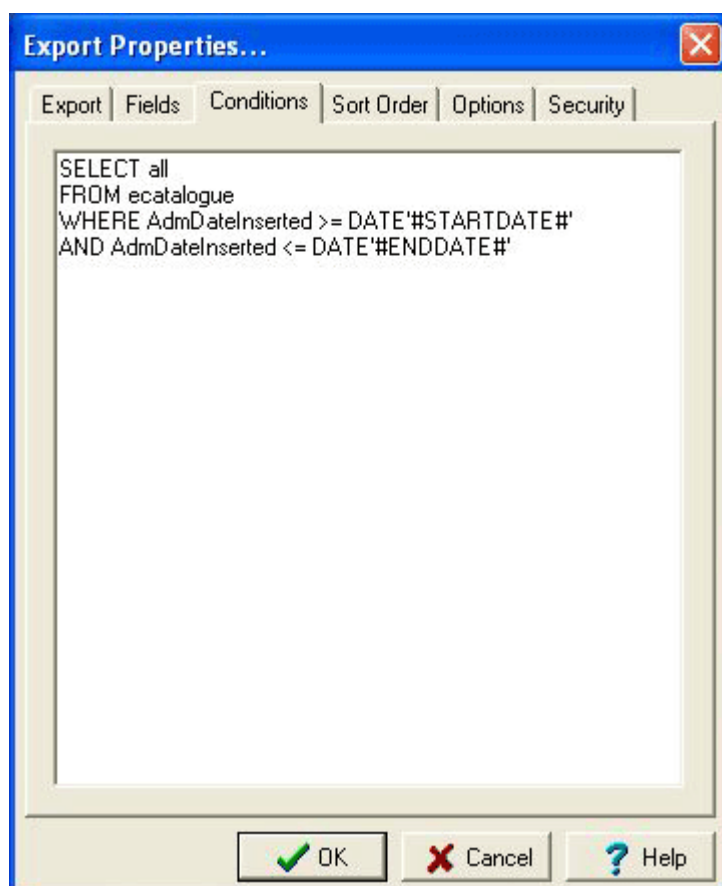
The following placeholders may be used in the TexQL statement:

#TODAY#	This placeholder is replaced with the date on which the export is run. The date order used (e.g. day/month/year, month/day/year) is as specified for your EMu system.
#STARTDATE#	This placeholder is replaced with the date of the first day after the end date of the previous export. It is used for date range queries to specify the starting date for the exported records.
#ENDDATE#	This placeholder is replaced with the date of the last day for the records to be exported. The end date is always the day before the scheduled date as exports are run early in the morning of the scheduled day.
#MONTH#	<p>This placeholder is replaced with the month of the day that falls immediately prior to the day of the export.</p> <p>The month is a number between 1 and 12.</p> <p>For example, if the export date is 3 May, #MONTH# is 5 (May); if the export date is 1 May, #MONTH# is 4 (April).</p>
#YEAR#	This placeholder is replaced with the year of the day prior to the execution of the export.



Placeholders use the day before the date on which the export is run as the closing date because exports are run early in the morning: the closing date for searches does not include the current date as the day is not yet complete.

When the Export Properties dialogue box is closed by selecting the **OK** button, the TexQL statement is checked to ensure that a valid query has been specified:



Hint - Determining the TexQL Conditions statement

A TexQL condition statement is a query used to retrieve the records to be exported. Any valid TexQL query statement may be used. Unfortunately the format of the statement may be hard to piece together unless you are familiar with SQL. Fortunately EMu provides an easy mechanism for generating a statement based on query terms entered into the current module.

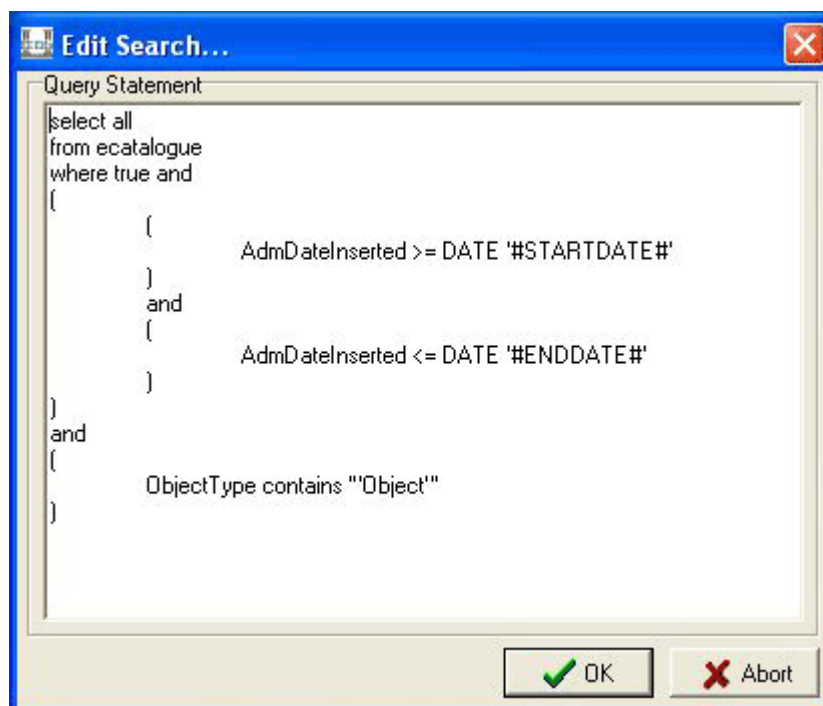
1. Open the module from which the records are to be exported.
The module should be in Search mode.
2. Enter search terms as though you were retrieving the records you would like to export.



Note that there may be search terms specified in a module by default. If these are not required, clear them.

Where you want to use a date range containing place markers, enter the place marker into the control:

3. Select **File>Show Search** in the Menu bar
-OR-
Use the keyboard shortcut, ALT+F+O.
The Edit Search dialogue box displays:



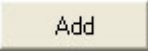
4. Copy the contents of the Edit Search box.

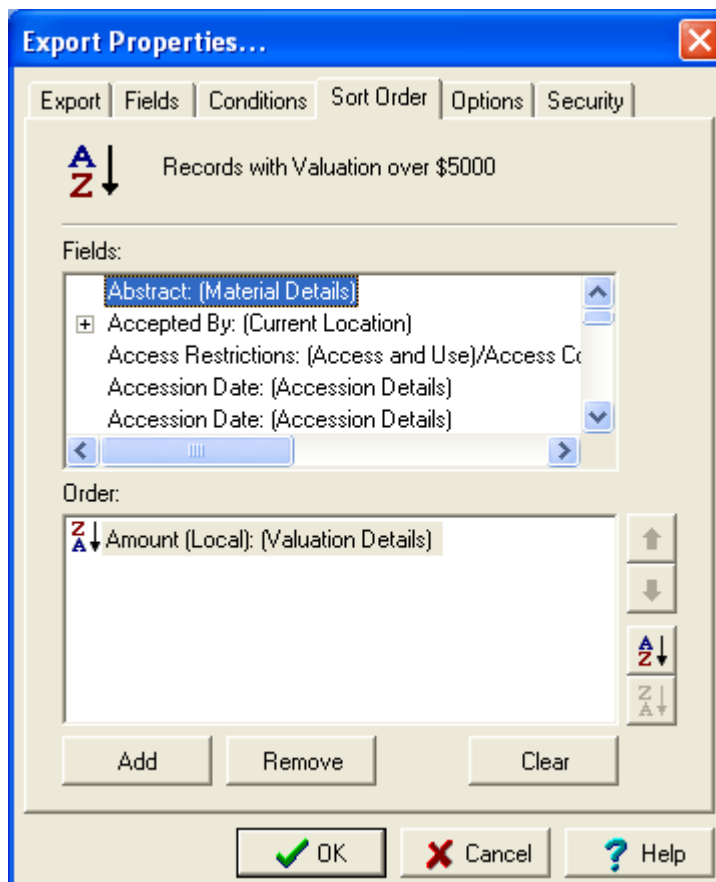
5. Select **Abort** .

The Edit Search box is closed.

The TexQL condition statement to use for your export can now be pasted into the Conditions tab when defining your Export Properties.

Define sort order: the Sort Order tab

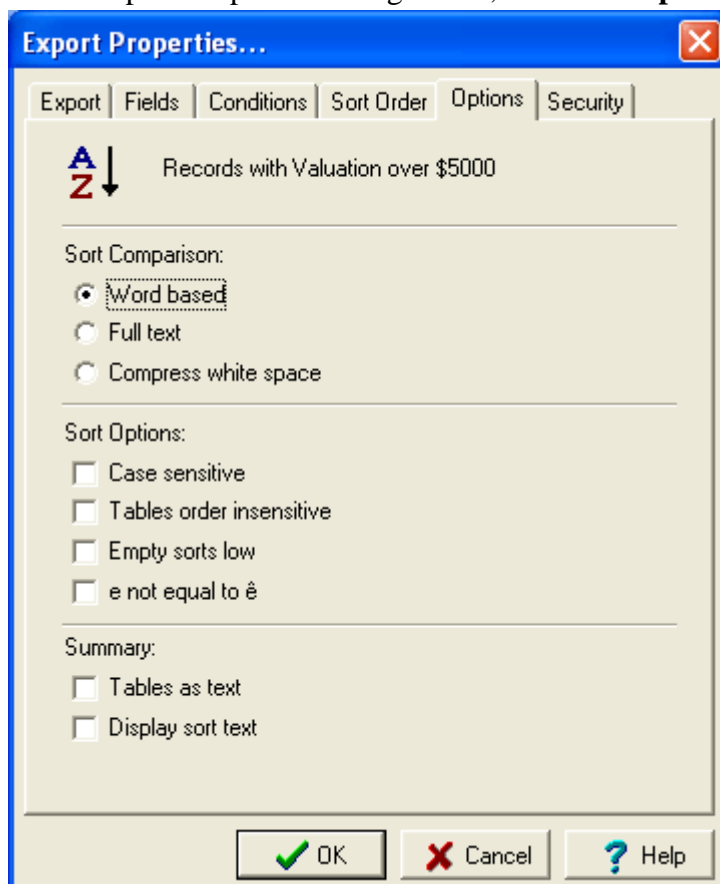
1. In the Export Properties dialogue box, select the **Sort Order** tab.
All fields in the module are listed in the Fields list.
2. Select a field in the Fields list and select **Add**  to add it to the Order list.
-OR-
Double-click the field name in the Fields list:



Add as many fields on which to sort as required.

Set sort options: the Options tab

1. In the Export Properties dialogue box, select the **Options** tab:

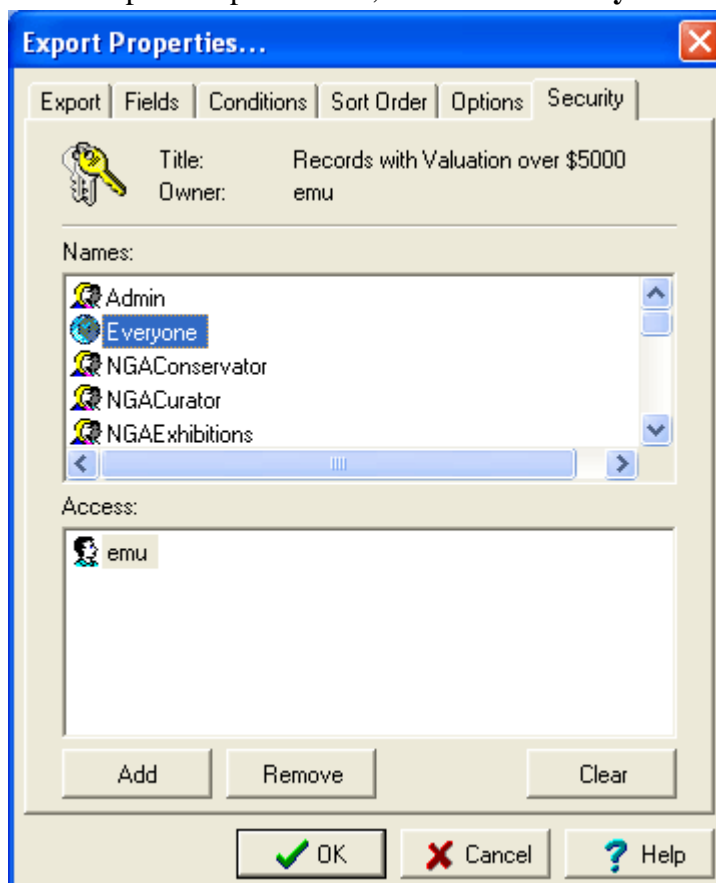


2. Select the required Sort Options.
For more details regarding sort options see *Set sort options: the Options tab* in the Reports section of the EMu help.

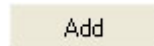
Set security options: the Security tab

On the Security tab we determine who is authorised to run your export by adding users or groups to the Access list:

1. In the Export Properties box, select the **Security** tab:



2. Select a user / group name (e.g. **Everyone**) in the Names box and select **Add**



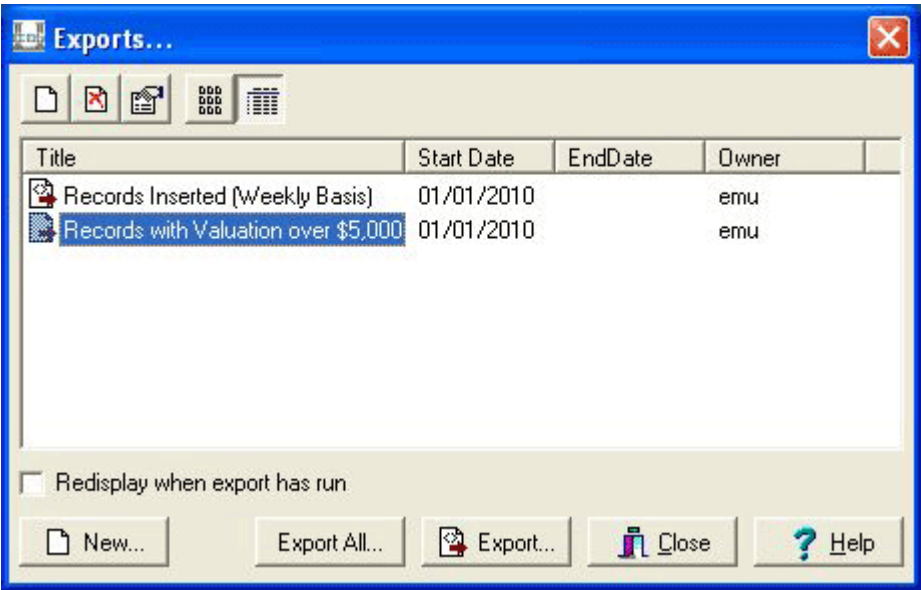
-OR-

Double-click the user / group name in the Names list.

The name is added to the Access list.

3. Select **OK** 

Any TexQL statement added on the Conditions tab is now validated and if all is well the new scheduled export will display in the Exports dialogue box:



Examples

Scenario 1

A record creation project is under way. As part of the outputs of the project a weekly file is to be produced containing the *Accession Number*, *Title* and *Valuation* of all records added in the week.

Solution

As the file is to be produced weekly, the export frequency required is `Daily` (as the frequency is less than monthly). The `Sunday` checkbox should be selected. As no Finish Date is specified, these settings will result in the export running early Sunday morning, once a week, exporting all data from the previous Sunday through to the Saturday.

The Conditions statement needs to query for all records inserted into the Catalogue module in the previous week. As the export is run once a week the `#STARTDATE#` place holder will be set to the date of the previous Sunday and the `#ENDDATE#` will be the date of the Saturday immediately before the export is run.


For example, if the export is run on the morning of Sunday, 14 February 2010, then `#ENDDATE#` will be 13 February 2010 (Saturday) and `#STARTDATE#` will be 7 February 2010 (the previous Sunday). The required Conditions statement is:

```
SELECT all
FROM    ecatalogue
WHERE   AdmDateInserted >= DATE '#STARTDATE#'
AND     AdmDateInserted <= DATE '#ENDDATE#'
```

The Export and Conditions tabs are:

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

 Record Creation Export (Weekly)

Filter:

Language: All Languages

Start Date: 01/01/2010

Finish Date:

Frequency: Daily

Every Day(s) On ☒ Sunday ☐ Thursday
☐ Monday ☐ Friday
☐ Tuesday ☐ Saturday
☐ Wednesday

☒ OK ☐ Cancel ☐ Help

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

```
SELECT all
FROM ecatalogue
WHERE AdmDateInserted >= DATE '#STARTDATE#'
AND AdmDateInserted <= DATE '#ENDDATE#'
```

☒ OK ☐ Cancel ☐ Help

Scenario 2

A quarterly export of all objects with a valuation of at least \$500.00 is required by your insurance company. The export must list the *Accession Number*, *Title*, *Valuation Amount* and *Date Valued*.

Solution

Since the export is required every three months, the `Monthly` export frequency should be chosen. The *Every nnn Month(s)* field should be set to 3 to ensure the export only occurs once every three months (that is, quarterly). The *Day nnn of the Month* field should be set to 1 as we want the export to include all records up to but not including the first day of the month. Finally the *Start Date* needs to be set to the first day of a quarterly date boundary. If our quarters start on 1 January, 1 April, 1 July and 1 October, then one of these dates should be chosen.


The Conditions statement needs to retrieve all records with a valuation of at least \$500.00. As there are no date restrictions, we do not require a date range as part of the statement. A suitable Conditions statement would be:

```
SELECT all
FROM    ecatalogue
WHERE   ValValuationAmount >= 500.00
```

The Export and Conditions tabs are:

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

 Works over \$500.00 (Quarterly)

Filter:

Language:

Start Date:

Finish Date:

Frequency:

Every Month(s)

Day of the Month

☐ First Weekday of the Month

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

```
SELECT all
FROM ecatalogue
WHERE ValValuationAmount >= 500.00
```

Scenario 3

In order to send a copy of the Annual Report to all Financial Contributors to your institution, an export file containing mailing addresses is required. The report is only sent to Financial Contributors registered on 30 June in the given year. The export should produce a CSV file.

Solution

Since we are dealing with people, the Scheduled Export needs to be created in the Parties module. The data is generated annually, so the `Annually` export frequency should be used. The *Day nnn and Month mmm of Each Year* should be set to `Day 1 and Month July of Each Year`. As the export is run early in the morning on 1 July, it will include all records up to and including 30 June. As CSV (Comma Separated Values) output is required, the *Filter* should be set to `CSV Format (iso-8899-1)`.


The Conditions statement needs to find all Financial Contributors registered on 30 June. We only need to find Parties records containing a *Role* of Financial Contributor. A suitable Conditions statement is:

```
SELECT all
FROM    eparties
WHERE   EXISTS
(
        Role_tab WHERE Role CONTAINS 'Financial Contributor'
)
```

The required Export and Conditions tabs are:

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

 Annual Report Addresses (CSV)

Filter: CSV Format (iso-8859-1)

Language: All Languages

Start Date: 01/07/2010

Finish Date:

Frequency: Annually

Day of the Year

Day 1 and Month July of Each Year

☐ First Weekday of the Year

OK Cancel Help

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

```
SELECT all
FROM eparties
WHERE EXISTS
(
  Role_tab WHERE Role CONTAINS 'Financial Contributor'
)
```

OK Cancel Help

Scenario 4

You work in the ornithology department of a natural history museum. From time to time you receive requests from your local environmental authority asking for a data file containing a list of all specimens you hold for a given taxon. The data must be supplied in Darwin Core XML format.

Solution

The data exports are not required on a regular basis, so the `Adhoc` export frequency should be used. As a schedule is not defined for `Adhoc` exports you will need to run an unscheduled export to produce the required data file. See [Unscheduled Exports](#) (page 27) for more details. In order to produce the correct output format the `Darwin Core XML` filter should be used.


The `Conditions` statement needs to retrieve all specimens for a given taxon. The Darwin Core `DarScientificName` field allows us to find the required Catalogue records. A suitable `Conditions` statement would be:

```
SELECT all
FROM    ecatalogue
WHERE   DarScientificName CONTAINS '#TAXON#'
```

We have added our own `#TAXON#` place holder in the `Conditions` statement. Before running an export we need to replace the place holder with the required taxon (simply edit the `TexQL` on the `Conditions` tab, replacing `#TAXON#` with the desired taxon). The required `Export` and `Conditions` tabs are:

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

 Darwin Core XML export

Filter: Darwin Core XML

Language: All Languages

Start Date: 01/01/2010

Finish Date:

Frequency: Adhoc

☒ OK ☐ Cancel ☐ Help

Export Properties...

Export | Fields | Conditions | Sort Order | Options | Security

```
SELECT all
FROM ecatalogue
WHERE DarScientificName CONTAINS '#TAXON#'
```

☒ OK ☐ Cancel ☐ Help

SECTION 3

Unscheduled Exports

The Scheduled Exports facility allows you to have data exports occur at defined date intervals. It may also be useful to perform a data export immediately rather than at the scheduled time. In this case, we generate the export data by running the export manually. These manual runs are known as Unscheduled Exports.

Another difference between an Unscheduled and Scheduled Export is that Unscheduled Exports use the current matching records as the data source, whereas a Scheduled Export uses the Conditions statement to retrieve the records to be exported. Unscheduled Exports ignore the Conditions statement.

Running Unscheduled Exports

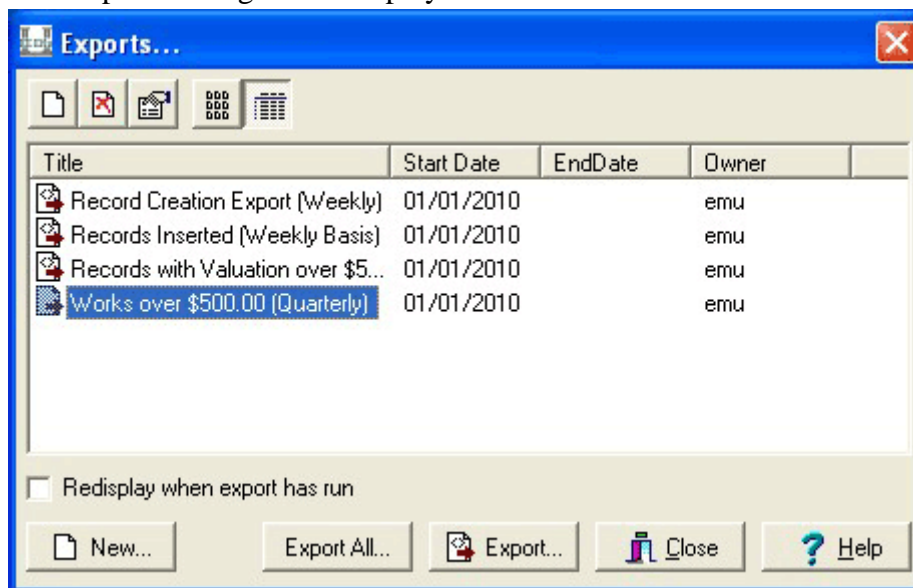
To run an Unscheduled Export:


1. Search for or otherwise list a group of records.
2. Select **Tools>Exports** in the Menu bar.

-OR-

Use the keyboard shortcut, ALT+T+X.

The Exports dialogue box displays:



3. Select the export to run.
 4. Select **Export**  to export the current record or selected records
- OR-

Select **Export All**  to export all listed records.

The export is run with a status dialogue box displayed. When the export has finished, a completed dialogue box displays:



5. Select **OK** .

The results of the export are stored on the EMu server. To view the results it is necessary to look in the Exports module. See Viewing Export Results (page 29) for more details.


SECTION 4

Viewing Export Results

Scheduled Exports are run automatically by EMu. For each export executed a record is created in the Exports module. The record includes:

- Name of the Scheduled Export
- Number of records exported
- The module from which the records were exported
- The filter used (if any)
- The #STARTDATE# and #ENDDATE# values used
- Any errors that occurred
- The export data files

View Export Results

1. Select **Exports**  **Exports** from the Command Centre.
The Exports module displays in Search mode.
2. Enter the name of the Export to view in *Schedule: (Export Details)*.



If you're looking for a recently run export, you could run a search without entering search terms in *Schedule: (Export Details)* as exports are listed with most recent exports first.

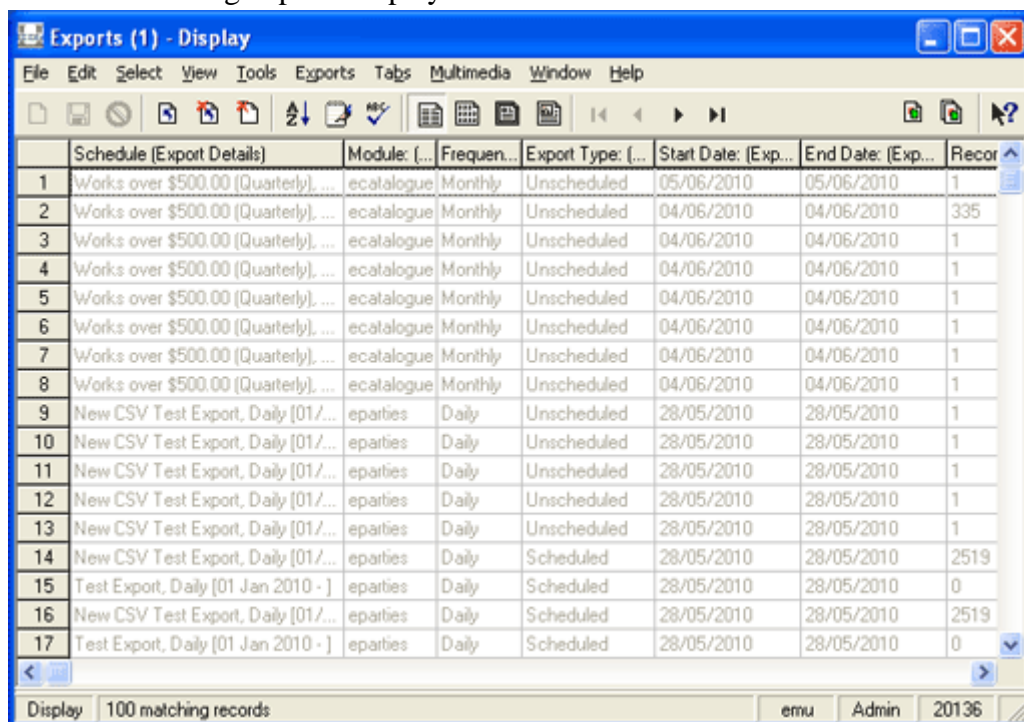
3. Select **File>Search** in the Menu bar
-OR-

Select **Search**  in the Toolbar

-OR-

Use the keyboard shortcut, CTRL+F.

A list of matching exports displays:



	Schedule (Export Details)	Module: [...]	Frequen...	Export Type: [...]	Start Date: (Exp...	End Date: (Exp...	Reco...
1	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	05/06/2010	05/06/2010	1
2	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	335
3	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
4	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
5	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
6	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
7	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
8	Works over \$500.00 (Quarterly), ...	ecatalogue	Monthly	Unscheduled	04/06/2010	04/06/2010	1
9	New CSV Test Export, Daily [01/...	eparties	Daily	Unscheduled	28/05/2010	28/05/2010	1
10	New CSV Test Export, Daily [01/...	eparties	Daily	Unscheduled	28/05/2010	28/05/2010	1
11	New CSV Test Export, Daily [01/...	eparties	Daily	Unscheduled	28/05/2010	28/05/2010	1
12	New CSV Test Export, Daily [01/...	eparties	Daily	Unscheduled	28/05/2010	28/05/2010	1
13	New CSV Test Export, Daily [01/...	eparties	Daily	Unscheduled	28/05/2010	28/05/2010	1
14	New CSV Test Export, Daily [01/...	eparties	Daily	Scheduled	28/05/2010	28/05/2010	2519
15	Test Export, Daily [01 Jan 2010 -]	eparties	Daily	Scheduled	28/05/2010	28/05/2010	0
16	New CSV Test Export, Daily [01/...	eparties	Daily	Scheduled	28/05/2010	28/05/2010	2519
17	Test Export, Daily [01 Jan 2010 -]	eparties	Daily	Scheduled	28/05/2010	28/05/2010	0

Display 100 matching records

emu Admin 20136

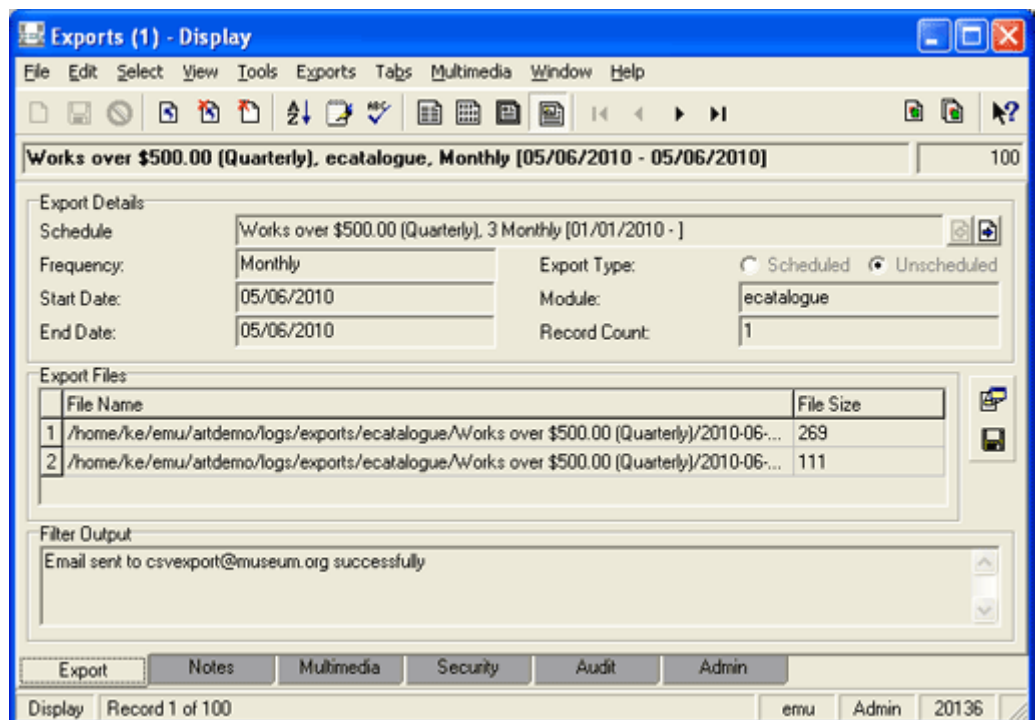
4. Select the export you want to view.
5. Select **View>Details** in the Menu bar
-OR-

Select **View Details**  in the Toolbar

-OR-


Use the keyboard shortcut, ALT+V+D.

The details of the export displays:




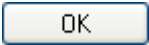
The export files generated in the export process are listed in the *Export Files* table. The files are stored on the EMu server. Any messages or errors are shown in the *Filter Output* box.

View Export Data


1. Click the row in the *Export Files* table with the file to be viewed.
2. Select **Launch Viewer**  next to the Export Files table.
-OR-
Select **Exports>Launch** in the Menu bar and select the file to be viewed from the sub-menu that displays.

The application / viewer associated with the file extension is invoked to display the file.

Save all Export Data files

1. Select **Save All**  beside the *Export Files* table
-OR-
Select **Exports>Save>All** in the Menu bar.
The Browse for Folder dialogue box displays.
2. Choose the directory into which all export files will be saved.
3. Select **OK** .

Save an Export Data file

1. Select **Exports>Save** in the Menu bar
-OR-
Use the keyboard shortcut, ALT+X+S.
The Save sub-menu lists all export data files.
2. Select the file to save from the sub-menu.
The Save As dialogue box displays.
3. Choose the location to save the export file.
4. Select **Save** .

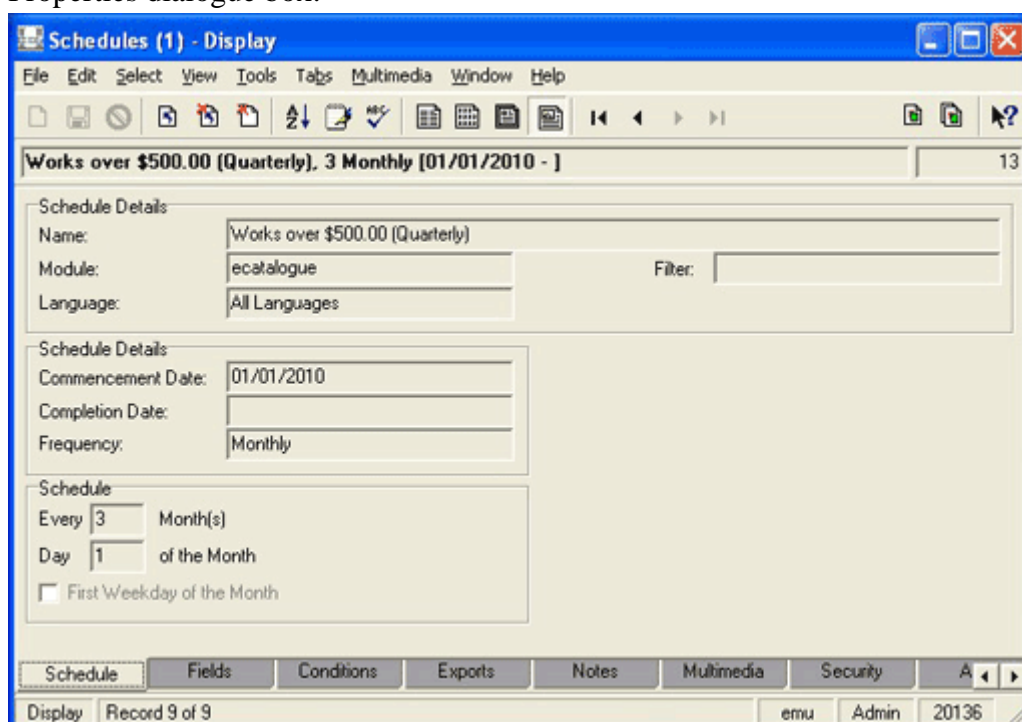
SECTION 5

The Schedules and Exports Modules

The Scheduled Exports facility uses two modules to store export details:

- Schedules Module  Schedules

The Schedules module holds one record for each export defined in the Export Properties dialogue box:



The screenshot shows a software window titled "Schedules (1) - Display". It has a menu bar (File, Edit, Select, View, Tools, Tabs, Multimedia, Window, Help) and a toolbar. The main content area displays details for a schedule named "Works over \$500.00 (Quarterly), 3 Monthly [01/01/2010 -]".

Schedule Details:

- Name: Works over \$500.00 (Quarterly)
- Module: ecatalogue
- Language: All Languages
- Filter: (empty)

Schedule Details:

- Commencement Date: 01/01/2010
- Completion Date: (empty)
- Frequency: Monthly

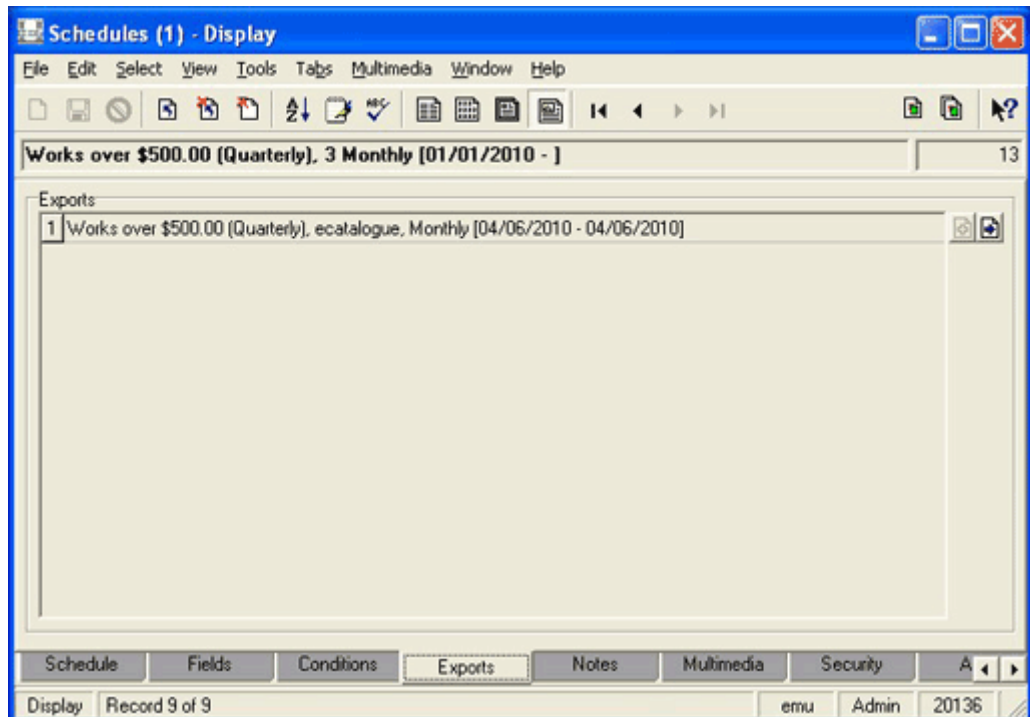
Schedule:


- Every 3 Month(s)
- Day 1 of the Month
- ☐ First Weekday of the Month

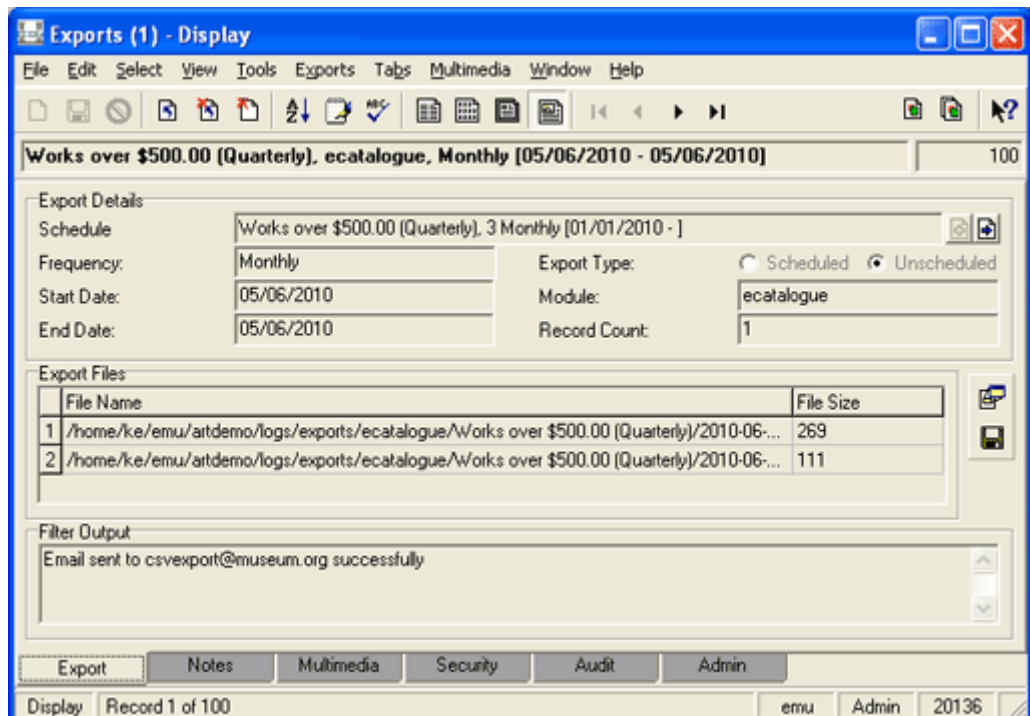
At the bottom, there are tabs: Schedule, Fields, Conditions, Exports, Notes, Multimedia, Security. The "Schedule" tab is active. Below the tabs, it says "Display Record 9 of 9". On the far right, there are fields for "emu", "Admin", and "20136".

The Scheduled Export settings on the Schedule, Fields, Conditions and Exports tabs are read-only as the values are maintained via the Export Properties dialogue box; all other Tabs are editable.

The Exports tab lists all data exports performed via this Schedules record. The list is sorted from most recent export to the oldest. Both Scheduled and Unscheduled Exports are shown:



- Exports Module  Exports
- The Exports module holds one record per export executed. An Exports record is linked to its associated Schedules record:



The *Export Files* table lists all files created when the export was run, along with their size. Any messages or errors generated display in the *Filter Output* box. Data values displaying on the Export tab are read-only; all other Tabs are editable.

As with other EMu modules, the Schedules and Exports modules provide reporting, sorting, etc.

SECTION 6

Creating Filters

When defining a Scheduled Export in the Export Properties dialogue box, it is possible to select a filter that transforms the standard XML export data format into another format suitable for delivery (e.g. CSV). If a filter is not specified, the export files contain EMu XML formatted data (as used by the EMu reporting facility). As final delivery formats may vary widely, System Administrators may need to create localised filters. In this section we look at how to add a filter into the Scheduled Exports facility.

Where are filters stored?

Filters, which are programs or scripts, reside under the `etc/exports` or `local/etc/exports` directory on the EMu server.

They may be module specific or may be applicable to all modules. Module specific filters are stored under a sub-directory named after the module to which they apply. For example, Parties module specific filters would be located under `etc/exports/eparties` or `local/etc/exports/eparties`.



When adding your own filters to the EMu server, they should be placed under `local/etc/exports` as `etc/exports` is reserved for use by EMu upgrades.

The name of the file containing the program or script is the label displayed in the Filters drop list when specifying a new Scheduled Export.

How is a filter invoked?

When an export is performed the initial data export is in EMu XML format. The data is placed in a temporary file on the EMu server and passed as an argument to the filter specified for the Scheduled Export. The filter receives only one argument, the name of the file containing the exported data.

Several environment variables are set and available to the filter when it executes:

EXPORTTITLE	The title given to the Scheduled Export when it was created.
EXPORTIRN	The IRN (Internal Record Number) of the Schedules record used to generate the exported data file. Using <code>EXPORTIRN</code> it is possible to retrieve details from the Schedules record.
EXPORTTABLE	The module from which the data was exported.
EXPORTSTARTDATE	The value of <code>#STARTDATE#</code> used by the export process.
EXPORTENDDATE	The value of <code>#ENDDATE#</code> used by the export process.
EXPORTRECORDCOUNT	The number of records exported.

Filter Output and Errors

If the filter is unable to process the exported data, any error messages should be written to `STDERR` and the export terminated with a non-zero exit status. This notifies EMu that the filter did not complete successfully and an export failure is recorded. Any text written to `STDERR` is displayed in the *Filter Output* box for an Exports record in the Exports module.

Once the filter has processed the export data it should print out to `STDOUT` a list of all the files generated. These files are moved under the `logs/exports` directory on the EMu server. The file names are displayed in the *Export Files* table in the Exports record created. If the filter completed successfully, it should return a zero exit status.

Filtering Data

The filter may manipulate the export data in any way required. It may email the data (via `sendmail`), print it (via `lpr` or `lp`) or even send it to remote machines (via `rsh` or `ssh`). A copy of the data exported should always be attached to the Export record so it can be reviewed at a later date, if required.

An example filter

In this example filter we convert the export data from XML to CSV (Comma Separated Values) and email the file(s) produced to a mail alias address (csvexport@client.org) on our local mail server. The mail server will then forward the email onto a host of recipients.

We can use `texxmlodbc` to convert the export XML to CSV files. The filter is written using the perl scripting language as it provides a nice package for sending multi-part email messages.

The filter is:

```
#!/usr/bin/perl

use strict;
use warnings 'all';
use MIME::Lite;

#
# First we convert the EMu XML to CSV and save
# the list of files generated.
#
if (! open(FILE, "texxmlodbc --output-encoding=iso-8859-1
'$ARGV[0]' |"))
{
    print STDERR "Cannot execute texxmlodbc";
    exit(1);
}
my @files = <FILE>;
close(FILE);

#
# Build up the message to send.
#
my $body = <<END;
The attached files were generated by the EMu Scheduled Export
facility.
The details of the export are:

Title: $ENV{EXPORTTITLE}
Module: $ENV{EXPORTTABLE}
Start Date: $ENV{EXPORTSTARTDATE}
End Date: $ENV{EXPORTENDDATE}
END

#
# Create an email message to csvexport@client.org
#
my $msg = MIME::Lite->new(
    To      => 'csvexport@client.org',
    Subject => 'EMu Scheduled Export CSV Files',
    Type    => 'multipart/mixed'
);

#
```

```

# Add the email text first.
#
$msg->attach(
    Type    => 'TEXT',
    Data    => $body
);

#
# Add each file.
#
foreach my $file (@files)
{
    #
    # Figure out the file mime type and attach it
    #
    chomp($file);
    my $type = ($file =~ /\.ini$/i) ? 'TEXT' : 'text/csv';
    $msg->attach(
        Type          => $type,
        Path          => $file,
        Disposition   => 'attachment'
    );
}

#
# Send the email.
#
if (! $msg->send())
{
    print STDERR 'Cannot send email to csvexport@client.org';
    exit(1);
}
print STDERR 'Email sent to csvexport@client.org successfully';

#
# Output the file list for the export record.
#
print join("\n", @files);
exit(0);

```

Notice how the filter outputs the names of the CSV files at the end so they can be attached to the Export record created. Once attached these files may be reviewed at any time.

SECTION 7

Cron setup

In order for Scheduled Exports to be produced the `emuexport` command must be run, typically early each morning. The server based cron facility provides this ability. The following crontab entry needs to be installed to ensure the correct functioning of the Scheduled Export facility:

```
0 1 * * * ~emu/bin/emurun emuexport 2>&1 | ~emu/bin/emurun  
emulogger -t "KE EMu Scheduled Exports Report" exports
```

where `~emu` is replaced with the full path to where EMu is installed on the server machine.

The time at which the command is run (1:00 am in the example above) should be adjusted to fit in with any EMu maintenance scheduled.

SECTION 8

Using emuexport

The `emuexport` command is the program used to execute Scheduled Exports. It interacts with the Schedules and Exports modules to ensure that data exports occur when scheduled. The program may be used in four different ways:

1. **Run all Scheduled Exports**

Usage: `emuexport`

Any Scheduled Exports required to be run will be executed. The current date is used to determine what exports are required. This form of the command is used by cron on a daily basis to ensure all Scheduled Exports for the given day are performed.

2. **Run an Unscheduled Export**

Usage: `emuexport -f filename -g recordcount -i irn`

The `filename` argument is the path to a file containing the export data in EMu XML format. A suitable file can be generated using the EMu reporting facility and selecting a Report Type of **Export XML Document**. The `recordcount` is the number of records in the XML file and `irn` is the IRN of the Schedules record to execute. An Unscheduled Export run within EMu uses this form of `emuexport`.

3. **Run Scheduled Exports for a specific date**

Usage: `emuexport -R date [-i irn]`

The `date` argument indicates the date to use to determine what Scheduled Exports need to be run. The date format expected is `yyyy/mm/dd`. An optional IRN of a Schedules record may be supplied to force a particular Scheduled Export to execute using the supplied date. This form of `emuexport` is useful for re-running Scheduled Exports for a given date, generally because of problems with the nightly automatic run. It can also be used for internal testing of `emuexport`.

4. **Run a specific Scheduled Export**

Usage: `emuexport -i irn [-s startdate -e enddate]`

The `irn` argument is the IRN of a Schedules record to be executed. Optional `startdate` and `enddate` values (in `yyyy/mm/dd` format) may be used to override the computed `#STARTDATE#` and `#ENDDATE#` values respectively. This form of `emuexport` is very useful for testing filters as it allows a specific schedule to be run without waiting for the Scheduled Export date to arrive.

In all usages of `emuexport` the appropriate Exports record is created for each export executed.

Index

A

An example filter • 44

C

Creating a Scheduled Export • 3

Creating Filters • 1, 5, 41

Cron setup • 47

D

Define sort order
the Sort Order tab • 15

Determine which records to export
the Conditions tab • 11

E

Examples • 19

F

Filter Output and Errors • 42

Filtering Data • 43

H

Hint - Determining the TexQL Conditions
statement • 13

How is a filter invoked? • 42

O

Overview • 1

R

Running Unscheduled Exports • 32

S

Save all Export Data files • 36

Save an Export Data file • 36

Scenario 1 • 19

Scenario 2 • 22

Scenario 3 • 25

Scenario 4 • 28

Select the fields to include in the export
the Fields tab • 9

Set security options
the Security tab • 17

Set sort options
the Options tab • 16

Set up the Export schedule
the Export tab • 4

T

The Schedules and Exports Modules • 1,
37

U

Unscheduled Exports • 6, 28, 31

Using emuexport • 49

V

View Export Data • 36

View Export Results • 34

Viewing Export Results • 32, 33

W

Where are filters stored? • 41



EMu Documentation

Appending data using the EMu Import Facility

Document Version 1.0

EMu Version 4.0



Contents

SECTION 1	Overview	1
SECTION 2	Appending data	3
	Updating tables	5
	Example one - Replacing the contents of a table	5
	Example two - Replacing the contents of a table and skipping rows	6
	Example three - Appending a value to the end of a table	7
	Example four - Appending multiple values to the end of a table	8
	Example five - Prepending a value to the front of a table	9
	Example six - Replacing values in a table	10
	Example seven - Appending a reference to the end of a table	11
	Updating nested tables	12
	Example one - Replacing the contents of a nested table	15
	Example two - Appending an outer row to a nested table	16
	Example three - Appending values to an inner table in a nested table	17
	Example four - Prepending an outer row to a nested table	18
	Example five - Replacing an outer row in a nested table	19
	Example six - Replacing an inner row in a nested table	20
	Groups	21
	Index	29

SECTION 1

Overview

KE EMu 3.2.01 added the ability to import records specified in either XML (eXtensible Markup Language) or CSV (Comma Separated Values). The new Import Facility enabled records to be created, including the creation of any records referenced. Hence it is possible to import a new Catalogue record and have a Party record created for the *Creator* field if the Party record does not already exist. The Import Facility also provided a mechanism for updating records. Using the update feature it is possible to replace the contents of an existing field with new data. Unfortunately the complete contents of the field are replaced with the data imported.

In some instances it may be useful to update a list of values by:

- appending values after a given position or after the last value.
- prepend values before a given position or before the first value.
- replace values at a given position.

EMu 4.0.02 extends the Import Facility, allowing tables and nested tables of values to be updated rather than replaced. It is now possible to append, prepend and replace values at given positions with imported data.

A new *group* mechanism allows separate columns to be tied together when appending data, ensuring all values are placed on the same row in each column. For example, when updating a Catalogue record you may want the *Creator* and *Date Created* values to appear on the same row, as each piece of information is related to the other.

In the following section we will look at the extensions added to the Import Facility to support the appending of data for both XML and CSV based data sources.

SECTION 2

Appending data

The EMu Import Facility allows XML or CSV to be used for defining records for importing. The XML format used is the same as that produced by the EMu Reporting Facility, while CSV is a de facto standard for data interchange. Consider the XML below:

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple>
        <atom>Map Number 2314</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

or the equivalent CSV:

irn	LatSource_tab(1)
5000432	Map Number 2314

When imported, all values in the *LatSource_tab* column will be replaced with the value Map Number 2314. To provide support for updating rows in a table a new row attribute has been added to the `<tuple>` tag for XML, allowing a row position to be specified. The row attribute provides a mechanism for indicating what type of update should be applied and which row is affected. The format of the attribute is:

```
<tuple row='value'>
```

For a CSV data source the row number appears between brackets. CSV allows two formats to be used to specify the row attribute:

```
(row='value')
```

or the shorter form:

```
(value)
```

While double quotes may be used to enclose the row *value*, single quotes are recommended for CSV files as the CSV format uses double quotes to enclose data values. When saving CSV files in Excel, incorrect output will be produced if double quotes are used when specifying the row *value*.

The *value* of the row attribute may be:

nnn	where nnn is a row number. The number indicates the row position in the list of values to be modified. The first row is numbered 1. In essence this setting replaces the current value at that position.
+	indicates the row position is after the last value in the table. Any data will be appended to the list of values.
-	indicates the row position is before the first value in the table. Any data added will be put at the start of the list with existing values moved down.
=	indicates the row position is row one. Any data added will replace the existing value at this position.
nnn+	appends any data after row number <i>nnn</i> .
nnn-	prepends any data before row number <i>nnn</i> .
nnn=	replaces any data at row number <i>nnn</i> .

In all cases, if the new row number does not exist, it is created. For example, if the row setting is:

```
<tuple row='12+'>
```

or for CSV:

```
(row='12+') OR (12+)
```

and there are not twelve values in the table, the table would be padded out to twelve values (with empty rows) and the new value appended, creating a thirteenth row. If a row attribute is not defined, the default behaviour is to append to the end of the table.

In order to provide backwards compatibility with the existing Import Facility and to allow all values in a table to be replaced, the first <tuple> in a table for XML, or the first row specifier for CSV, define whether the values in the table are being updated or whether all values are being replaced. If the row attribute is not specified in XML, or just a row number is specified, then the contents of the column will be cleared and the imported values added. If a row attribute is provided and it contains an update operator (that is +, - or = with or without a leading row number), the contents of the table are updated, with the existing values retained.

The following section contains examples on how to replace or update the contents of a table using the new row attribute. The examples provide both XML and CSV solutions.

Updating tables

The examples below show how the row attribute can be used to overwrite or update the contents of table based columns.

Example one - Replacing the contents of a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple>
        <atom>new source 1</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(1)
5000432	new source 1

As the row attribute value in the *LatSource_tab* column does not contain an update modifier (that is, +, - or =), the contents of the column are cleared before adding in the new data. The tables below show before and after cases for the above import:

Before		After	
1	source 1	1	new source 1
2	source 2		
3	source 3		
4	source 4		

Example two - Replacing the contents of a table and skipping rows

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="1">
        <atom>new source 1</atom>
      </tuple>
      <tuple row="3">
        <atom>new source 3</atom>
      </tuple>
      <tuple row="5">
        <atom>new source 5</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(1)	LatSource_tab(3)	LatSource_tab(5)
5000432	new source 1	new source 3	new source 5

Once again as the first row attribute does not contain an update modifier, the contents of *LatSource_tab* will be cleared before the imported values are added. The use of the row attribute allows specific row positions to be set. The example below shows before and after cases of the above import:

Before		After	
1	source 1	1	new source 1
2	source 2	2	
3	source 3	3	new source 3
4	source 4	4	
		5	new source 5

Example three - Appending a value to the end of a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+">
        <atom>append source 1</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(+)
5000432	append source 1

As the row attribute value contains an update modifier, +, the existing contents of the *LatSource_tab* field are retained and added to:

Before		After	
1	source 1	1	source 1
2	source 2	2	source 2
3	source 3	3	source 3
4	source 4	4	source 4
		5	append source 1

Example four - Appending multiple values to the end of a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+">
        <atom>append source 1</atom>
      </tuple>
      <tuple row="+">
        <atom>append source 2</atom>
      </tuple>
      <tuple row="+">
        <atom>append source 3</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(+)	LatSource_tab(+)	LatSource_tab(+)
5000432	append source 1	append source 2	append source 3

Since the first row attribute contains an update modifier (the + modifier), the imported data updates the existing table values. Notice the use of the append modifier for each row to be appended:

Before		After	
1	source 1	1	source 1
2	source 2	2	source 2
3	source 3	3	source 3
4	source 4	4	source 4
		5	append source 1
		6	append source 2
		7	append source 3

Example five - Prepending a value to the front of a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="-">
        <atom>prepend source 1</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(-)
5000432	prepend source 1

Inserting a value into the first position in a table column is similar to appending, except that the prepend row operator (-) is used. When the value is inserted, existing values are moved down the table:

Before		After	
1	source 1	1	prepend source 1
2	source 2	2	source 1
3	source 3	3	source 2
4	source 4	4	source 3
		5	source 4

Example six - Replacing values in a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="2=">
        <atom>replace source 1</atom>
      </tuple>
      <tuple row="3=">
        <atom>replace source 2</atom>
      </tuple>
      <tuple row="6=">
        <atom>replace source 3</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(2=)	LatSource_tab(3=)	LatSource_tab(6=)
5000432	replace source 1	replace source 2	replace source 3

The = row operator is used to replace the contents of a row in a table. If the row does not exist, it is created and the appropriate value set:

Before		After	
1	source 1	1	source 1
2	source 2	2	replace source 1
3	source 3	3	replace source 2
4	source 4	4	source 4
		5	
		6	replace source 3

Example seven - Appending a reference to the end of a table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatDeterminedByRef_tab'>
      <tuple row="+">
        <atom name="NamFirst">firstname</atom>
        <atom name="NamLast">lastname</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatDeterminedByRef_tab(+).NamFirst	LatDeterminedByRef_tab(+).NamLast
5000432	firstname	lastname

All the update modifiers available for tables of values may be used for tables of references. For CSV data sources, columns with the same name and row attribute are handled as one update. For example, the CSV row attributes above both use the append operator (+). This does not result in two new rows, rather the same row is used to contain the reference:

Before		After	
1	reference 1	1	reference 1
2	reference 2	2	reference 2
3	reference 3	3	reference 3
4	reference 4	4	reference 4
		5	append reference 1

Updating nested tables

The mechanism for updating nested tables, that is tables within tables, is very similar to updating a single table. The only difference is that nested tables have an outer row number and an inner row number. Each outer row contains a table of inner row values:

Outer Row	Inner Row	Data
1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

As you can see from the table above, outer row one contains a table with three values, while outer row two has one value and outer row three, two values. The XML representation of the above table would be:

```

<table table='ecollectionevents'>
  <tuple>
    <table name='LatComment_nesttab'>
      <tuple>
        <table>
          <tuple>
            <atom>opinion 1, comment 1</atom>
          </tuple>
          <tuple>
            <atom>opinion 1, comment 2</atom>
          </tuple>
          <tuple>
            <atom>opinion 1, comment 3</atom>
          </tuple>
        </table>
      </tuple>
      <tuple>
        <table>
          <tuple>
            <atom>opinion 2, comment 1</atom>
          </tuple>
        </table>
      </tuple>
      <tuple>
        <table>
          <tuple>
            <atom>opinion 3, comment 1</atom>
          </tuple>
          <tuple>
            <atom>opinion 3, comment 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>

```

The nested table *LatComment_nesttab* starts with the `<table name='LatComment_nesttab'>` tag. Each outer row is enclosed in a `<tuple>` tag. The outer rows are coloured green. Within each outer row is a table of inner rows. The inner rows are enclosed in red `<tuple>` tags. When updating nested tables both the outer and inner row `<tuple>` tags may have row attributes. Hence it is possible to append/prepend/replace outer and/or inner rows.

The equivalent CSV representation is shown below. The table has been turned on its side for ease of viewing. The column names should appear in the first row rather than the first column:

irn	5000432
LatComment_nesttab(1:1)	opinion 1, comment 1
LatComment_nesttab(1:2)	opinion 1, comment 2
LatComment_nesttab(1:3)	opinion 1, comment 3
LatComment_nesttab(2:1)	opinion 2, comment 1
LatComment_nesttab(3:1)	opinion 3, comment 1
LatComment_nesttab(3:2)	opinion 3, comment 2

The outer and inner rows for nested tables in CSV are recorded after the column name enclosed in brackets separated by a colon. The outer row is shown in green, while the inner row is red. The example above uses the short form for the row attributes. The long form would look like:

```
LatComment_nesttab(row='1':row='1')
```

As with XML the outer and/or inner rows may contain update modifiers. The following examples detail some common uses.

Example one - Replacing the contents of a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple>
        <table>
          <tuple>
            <atom>new opinion 1, new comment 1</atom>
          </tuple>
        </table>
      </tuple>
      <tuple>
        <table>
          <tuple>
            <atom>new opinion 2, new comment 1</atom>
          </tuple>
          <tuple>
            <atom>new opinion 2, new comment 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(1:1)	LatComment_nesttab(2:1)	LatComment_nesttab(2:2)
5000432	new opinion 1, new comment 1	new opinion 2, new comment 1	new opinion 2, new comment 2

As the first outer row and first inner row do not contain row attributes the current contents of the *LatComments_nesttab* column will be cleared:

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	new opinion 1, new comment 1
2	1	new opinion 2, new comment 1
	2	new opinion 2, new comment 2

Example two - Appending an outer row to a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple row="+">
        <table>
          <tuple>
            <atom>append opinion 1, comment 1</atom>
          </tuple>
          <tuple>
            <atom>append opinion 1, comment 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(+:1)	LatComment_nesttab(+:2)
5000432	append opinion 1, comment 1	append opinion 1, comment 2

The first outer row has a row attribute with a value of +, indicating an outer row is to be appended to the existing data. The inner table contains the values to be added:

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2
4	1	append opinion 1, comment 1
	2	append opinion 1, comment 2

Example three - Appending values to an inner table in a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple row="2">
        <table>
          <tuple row="+">
            <atom>opinion 2, append comment 1</atom>
          </tuple>
          <tuple>
            <atom>opinion 2, append comment 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(2:+)	LatComment_nesttab(2:+)
5000432	opinion 2, append comment 1	opinion 2, append comment 2

The outer row contains a fixed row number while the first inner row contains an update modifier, the append (+) modifier. The combination indicates that values are to be appended to the end of the second outer row:

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
	2	opinion 2, append comment 1
	3	opinion 2, append comment 2
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

Example four - Prepending an outer row to a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple row="2-">
        <table>
          <tuple>
            <atom>prepend opinion 2, comment 1</atom>
          </tuple>
          <tuple>
            <atom>prepend opinion 2, comment 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(2-:1)	LatComment_nesttab(2-:2)
5000432	prepend opinion 2, comment 1	prepend opinion 2, comment 2

The outer row attribute value of 2- will prepend an outer row before row two and move the existing second row and subsequent rows down. The inner table contains the values to be set:

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	prepend opinion 2, comment 1
	2	prepend opinion 2, comment 2
3	1	opinion 2, comment 1
4	1	opinion 3, comment 1
	2	opinion 3, comment 2

Example five - Replacing an outer row in a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple row="2=">
        <table>
          <tuple>
            <atom>replace opinion 2, comment 1</atom>
          </tuple>
          <tuple row="3">
            <atom>replace opinion 2, comment 3</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(2=:1)	LatComment_nesttab(2=:3)
5000432	replace opinion 2, comment 1	replace opinion 2, comment 3

The 2= row attribute on the outer row indicates row two is to be replaced. Note the use of the row attribute in the inner table to skip row two (causing a blank value to be created):

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	replace opinion 2, comment 1
	2	
	3	replace opinion 2, comment 3
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

Example six - Replacing an inner row in a nested table

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatComment_nesttab'>
      <tuple row="3">
        <table>
          <tuple row="1=">
            <atom>opinion 3, replace comment 1</atom>
          </tuple>
          <tuple row="+">
            <atom>opinion 3, append comment 1</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatComment_nesttab(3:1=)	LatComment_nesttab(3:++)
5000432	opinion 3, replace comment 1	opinion 3, append comment 1

The outer row attribute value indicates row 3 is to be changed, while the first inner row indicates row one is to be replaced. The second inner row attribute will append a value to the end of the inner table:

Before

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, comment 1
	2	opinion 3, comment 2

After

1	1	opinion 1, comment 1
	2	opinion 1, comment 2
	3	opinion 1, comment 3
2	1	opinion 2, comment 1
3	1	opinion 3, replace comment 1
	2	opinion 3, comment 2
	3	opinion 3, append comment 1

Groups

The ability to append data to an existing list allows new values to be added to columns over time. One issue that may arise is the need to append values to more than one column, ensuring the values are all added at the same row position. Consider the following update:

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+">
        <atom>append source 1</atom>
      </tuple>
    </table>
    <table name='LatLatLongDetermination_tab'>
      <tuple row="+">
        <atom>append determination 1</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(+)	LatLatLongDetermination_tab(+)
5000432	append source 1	append determination 1

with the following result:

Before			After		
1	source 1	determination 1	1	source 1	determination 1
2	source 2	determination 2	2	source 2	determination 2
3	source 3		3	source 3	append determination 1
			4	append source 1	

The resulting data is almost certainly not what was envisioned. The two columns, *LatSource_tab* and *LatLatLongDetermination_tab* are related and appear in the one grid. As such there is an implied relationship between values in a row. For example, all values in the first row apply to the first opinion, while values in the second row apply to the second opinion and so on. When appending a new opinion, all the added data values need to appear in the same row regardless of any empty values already present.

As the Import Facility does not know the relationship between columns, a new attribute, the group attribute, has been added allowing columns to be tied together.

The attribute is also used to set the row position to be the first row to which data can be appended across all the columns in the group. For XML data, the group attribute appears in `<tuple>` tags along with the row attribute. For CSV, the group attribute is placed after the row attribute. It only makes sense to set a group where the row attribute is `row="+"`.

Consider the following update where a group is specified:

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+" group="opinion">
        <atom>append source 1</atom>
      </tuple>
    </table>
    <table name='LatLatLongDetermination_tab'>
      <tuple row="+" group="opinion">
        <atom>append determination 1</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV

irn	LatSource_tab(+ group='opinion')	LatLatLongDetermination_tab(+ group='opinion')
5000432	append source 1	append determination 1

Notice how the group attribute is defined on both the *LatSource_tab* and *LatLatLongDetermination_tab* columns. The group value can be any string. All columns with the same group value are examined to determine the row position to use when appending data. More than one group may be used where disjoint (unrelated) groups of related columns exists. Each disjoint set of related columns should have its own group name. The above update will produce:

Before

1	source 1	determination 1
2	source 2	determination 2
3	source 3	

After

1	source 1	determination 1
2	source 2	determination 2
3	source 3	
4	append source 1	append determination 1

which is probably what was required in the first place. Groups may be used with nested tables in the same way they can be with tables. A group may be applied to the outer and/or inner tuples in the nested table. The following update ensures the Source (*LatSource_tab*), Determination (*LatLatLongDetermination_tab*), Latitude

(*LatLatitude_nesttab*) and Longitude (*LatLongitude_nesttab*) values all append to the same row:

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+" group="opinion">
        <atom>append source 1</atom>
      </tuple>
    </table>
    <table name='LatLatLongDetermination_tab'>
      <tuple row="+" group="opinion">
        <atom>append determination 1</atom>
      </tuple>
    </table>
    <table name='LatLatitude_nesttab'>
      <tuple row="+" group="opinion">
        <table>
          <tuple>
            <atom>new opinion 1, latitude 1</atom>
          </tuple>
          <tuple>
            <atom>new opinion 1, latitude 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
    <table name='LatLongitude_nesttab'>
      <tuple row="+" group="opinion">
        <table>
          <tuple>
            <atom>new opinion 1, longitude 1</atom>
          </tuple>
          <tuple>
            <atom>new opinion 1, longitude 2</atom>
          </tuple>
        </table>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV



The table has been turned on its side for ease of viewing.

irn	5000432
LatSource_tab(+ group='opinion')	append source 1
LatLatLongDetermination_tab(+ group='opinion')	append determination 1
LatLatitude_nesttab(+ group='opinion':1)	new opinion 1, latitude 1
LatLatitude_nesttab(+ group='opinion':2)	new opinion 1, latitude 2
LatLongitude_nesttab(+ group='opinion':1)	new opinion 1, longitude 1
LatLongitude_nesttabb(+ group='opinion':2)	new opinion 1, longitude 2

Notice how the group attribute is applied to the outer row to ensure the new latitude and longitude values are added to the same row position as the source and determination values:

Before

1	1	source 1	determination 1	opinion 1, latitude 1	opinion 1, longitude 1
	2			opinion 1, latitude 2	opinion 1, longitude 2
	3			opinion 1, latitude 3	opinion 1, longitude 3
2	1	source 2		opinion 2, latitude 1	opinion 2, longitude 1

After

1	1	source 1	determination 1	opinion 1, latitude 1	opinion 1, longitude 1
	2			opinion 1, latitude 2	opinion 1, longitude 2
	3			opinion 1, latitude 3	opinion 1, longitude 3
2	1	source 2		opinion 2, latitude 1	opinion 2, longitude 1
3	1	append source 1	append determination 1	new opinion 1, latitude 1	new opinion 1, longitude 1
	2			new opinion 1, latitude 2	new opinion 1, longitude 2

The final point to cover concerning groups is that the row position at which new data is appended is calculated the first time the group is encountered in the data file. Once the row position is determined it is used for all columns in the group for the current import record. This means the group name used to bind columns together should only appear on the first row to be appended, rather than every row. If the group name appears on every row, each value updated will overwrite the previous value. Consider the update:

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+" group="opinion">
        <atom>append source 1</atom>
      </tuple>
      <tuple row="+" group="opinion">
        <atom>append source 2</atom>
      </tuple>
    </table>
    <table name='LatLatLongDetermination_tab'>
      <tuple row="+" group="opinion">
        <atom>append determination 1</atom>
      </tuple>
      <tuple row="+" group="opinion">
        <atom>append determination 2</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV



The table has been turned on its side for ease of viewing.

irn	5000432
LatSource_tab(+ group='opinion')	append source 1
LatSource_tab(+ group='opinion')	append source 2
LatLatLongDetermination_tab(+ group='opinion')	append determination 1
LatLatLongDetermination_tab(+ group='opinion')	append determination 2

This results in:

Before

1	source 1	determination 1
2	source 2	

After

1	source 1	determination 1
2	source 2	
3	append source 2	append determination 2

You may have noticed the value `append source 2` has overwritten the value `append source 1`. The reason is that when `group opinion` was encountered there were two rows in `LatSource_tab`. Hence `group opinion` refers to row three. As the second `<tuple>` uses the same group name as the first, they both refer to the row used by `group opinion` (row three in this case). Hence the second value overwrites the first value. If you want to append multiple values where the columns are related, a separate group name should be given for each row to be appended. That is:

XML

```
<table table='ecollectionevents'>
  <tuple>
    <atom name='irn'>5000432</atom>
    <table name='LatSource_tab'>
      <tuple row="+" group="opinion 1">
        <atom>append source 1</atom>
      </tuple>
      <tuple row="+" group="opinion 2">
        <atom>append source 2</atom>
      </tuple>
    </table>
    <table name='LatLatLongDetermination_tab'>
      <tuple row="+" group="opinion 1">
        <atom>append determination 1</atom>
      </tuple>
      <tuple row="+" group="opinion 2">
        <atom>append determination 2</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

CSV



The table has been turned on its side for ease of viewing.

irn	5000432
LatSource_tab(+ group='opinion 1')	append source 1
LatSource_tab(+ group='opinion 2')	append source 2
LatLatLongDetermination_tab(+ group='opinion 1')	append determination 1
LatLatLongDetermination_tab(+ group='opinion 2')	append determination 2

Which results in:

Before

1	source 1	determination 1
2	source 2	

After

1	source 1	determination 1
2	source 2	
3	append source 1	append determination 1
4	append source 2	append determination 2

Index

A

Appending data • 3

C

CSV • 21, 22, 24, 25, 26

E

Example five - Prepending a value to the front of a table • 9

Example five - Replacing an outer row in a nested table • 19

Example four - Appending multiple values to the end of a table • 8

Example four - Prepending an outer row to a nested table • 18

Example one - Replacing the contents of a nested table • 15

Example One - Replacing the contents of a table • 5

Example seven - Appending a reference to the end of a table • 11

Example six - Replacing an inner row in a nested table • 20

Example six - Replacing values in a table • 10

Example three - Appending a value to the end of a table • 7

Example three - Appending values to an inner table in a nested table • 17

Example two - Appending an outer row to a nested table • 16

Example Two - Replacing the contents of a table and skipping rows • 6

G

Groups • 21

O

Overview • 1

U

Updating nested tables • 12

Updating tables • 5

X

XML • 21, 22, 23, 25, 26



EMu Documentation

The ArcExplorer Report

Document Version 1.0

EMu Version 4.0.02



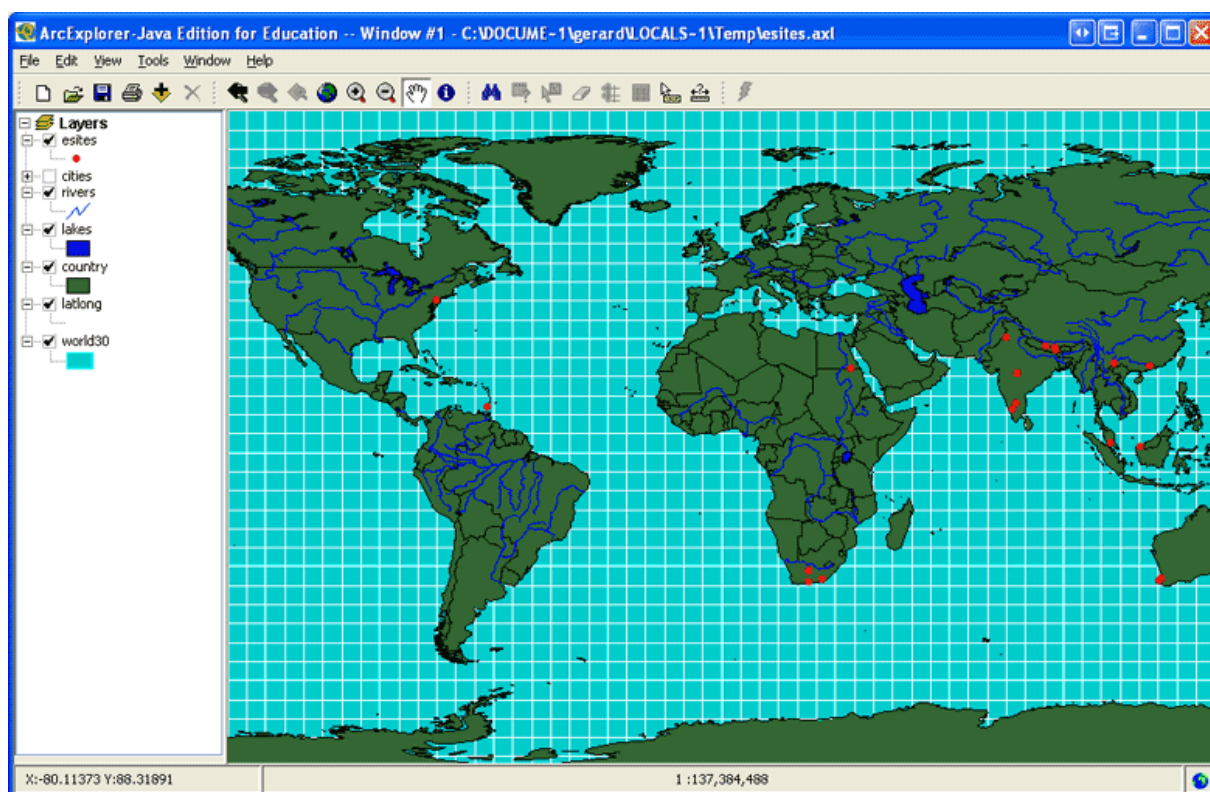
Contents

SECTION 1	The ArcExplorer Report	1
SECTION 2	How to set up EMu to run the ArcExplorer report	3
	Download and setup ArcExplorer on your computer	4
SECTION 3	How to run an ESRI ArcExplorer report in EMu	5
SECTION 4	How to make the ArcExplorer report	7
	Using an ArcExplorer AXL file	8
	Making a custom AXL file	13
	Index	15

SECTION 1

The ArcExplorer Report

The EMu ArcExplorer report takes EMu records with location (latitude / longitude) information and displays them on maps in the third party application, ArcExplorer:



ArcExplorer (<http://www.esri.com/software/arcexplorer/download.html>) is a freely available lightweight GIS data viewer that lets you perform a variety of basic GIS functions. With ArcExplorer, it is possible to:

- Display and query a variety of standard data sources, including:
 - ESRI shapefiles
 - ArcInfo coverages
 - Images
 - ArcIMS services
- Pan and zoom through multiple layers and identify, locate, and query geographic and attribute data.
- Symbolise your data based on attributes contained in your data layers to create thematic maps.
- Perform basic spatial analysis tasks on the geographic data, such as selecting and buffering features.

SECTION 2

How to set up EMu to run the ArcExplorer report

Running the ArcExplorer report requires that:

1. You are using an EMu client built after March 2010.
2. Two ArcExplorer Registry entries are specified in the EMu Registry.
3. ArcExplorer is installed on your computer (page 4).

Download and setup ArcExplorer on your computer

In order to run ArcExplorer reports in EMu it is necessary that the ESRI ArcExplorer application is installed on your computer. The application can be freely downloaded from <http://www.esri.com/software/arcexplorer/download.html> (it is necessary to register with ESRI in order to download the installation file).



The ArcExplorer report does not work with ArcExplorer 9.3.1, so be sure to download **ArcExplorer 2.3.2 (Java Edition for Education)** or later.

Installation instructions and System Requirements are provided by ESRI (although it is a fairly straightforward installation).

Information on the usage and licensing of ArcExplorer can be found at <http://www.esri.com/software/arcexplorer/common-questions.html> and in the PDF installed along with the application.


SECTION 3

How to run an ESRI ArcExplorer report in EMu


Running an ArcExplorer report is similar to running any other report in EMu:

1. Search for or otherwise list a group of records in a module.
2. Select **Tools>Reports** in the module Menu bar


-OR-

Select **Reports**  in the Toolbar.

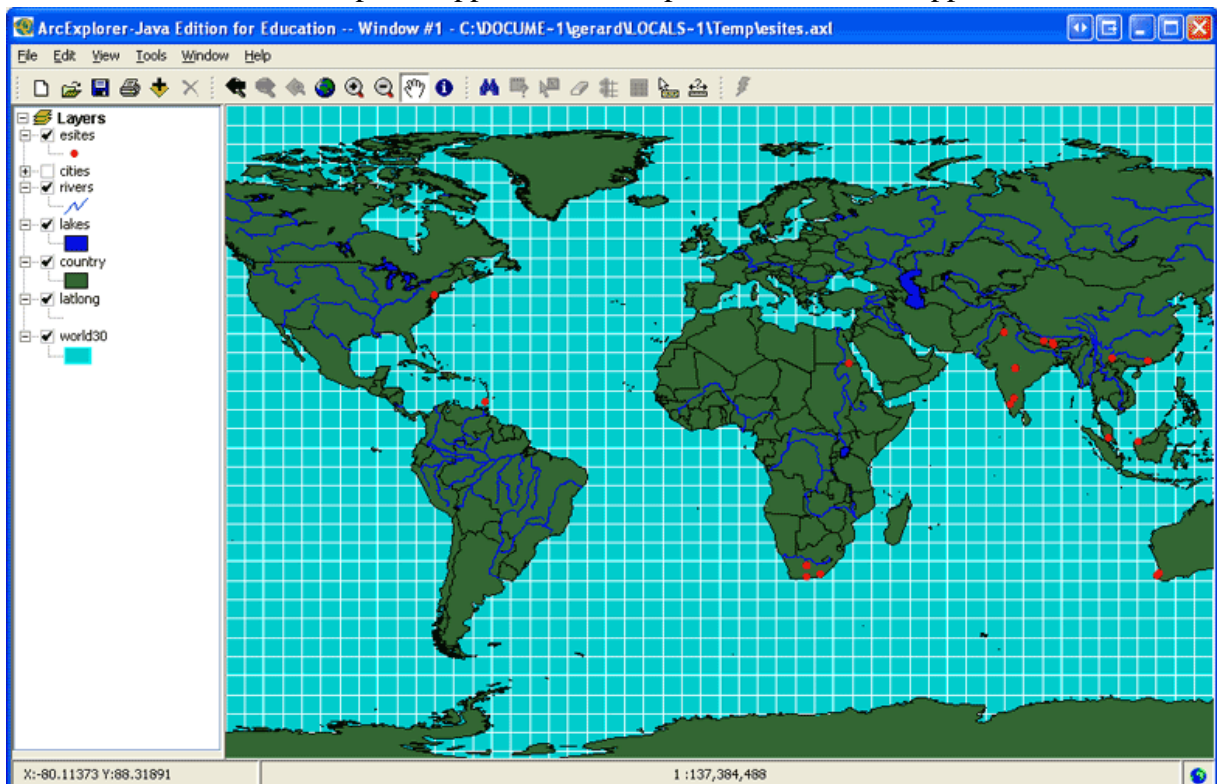
The Reports box displays with a list of pre-defined reports. The list includes the report name, its Type and the name of the owner of the report (who created it).

3. Select a report of Type **ArcExplorer**.
4. Select **Report**  to generate the report using the current record or selected records

-OR-

Select **Report All**  to generate the report using all listed records.

The ArcExplorer application will open with the data mapped:



With the data displayed in ArcExplorer, it can now be manipulated using the many tools available in the application.

SECTION 4

How to make the ArcExplorer report

In order to run an ArcExplorer report it is first necessary to design and create a report in EMu (page 8). This requires that you:

- Build or choose a map template.
Map templates can be created in ArcExplorer using the demo files provided as a starting point, or you can simply use any of the template maps provided with ArcExplorer. The end result is an AXL file.
- Create an ArcExplorer report in EMu and associate the AXL file with it.

Multiple reports can be designed, each using different map templates (e.g. plotting EMu records by ecological zones; or on a simple map of world boundaries, etc.).

When an ArcExplorer report is run in EMu, ArcExplorer opens with a map displaying the EMu data and any GIS layers that were specified in the template. Within ArcExplorer it is then possible to add GIS layers, query points and generally use the functionality of ArcExplorer to view the EMu records.

Using an ArcExplorer AXL file

ArcExplorer is loaded with a range of base templates. If the application was installed in the default location, these are stored in:

ESRI/AEJEE/Data/

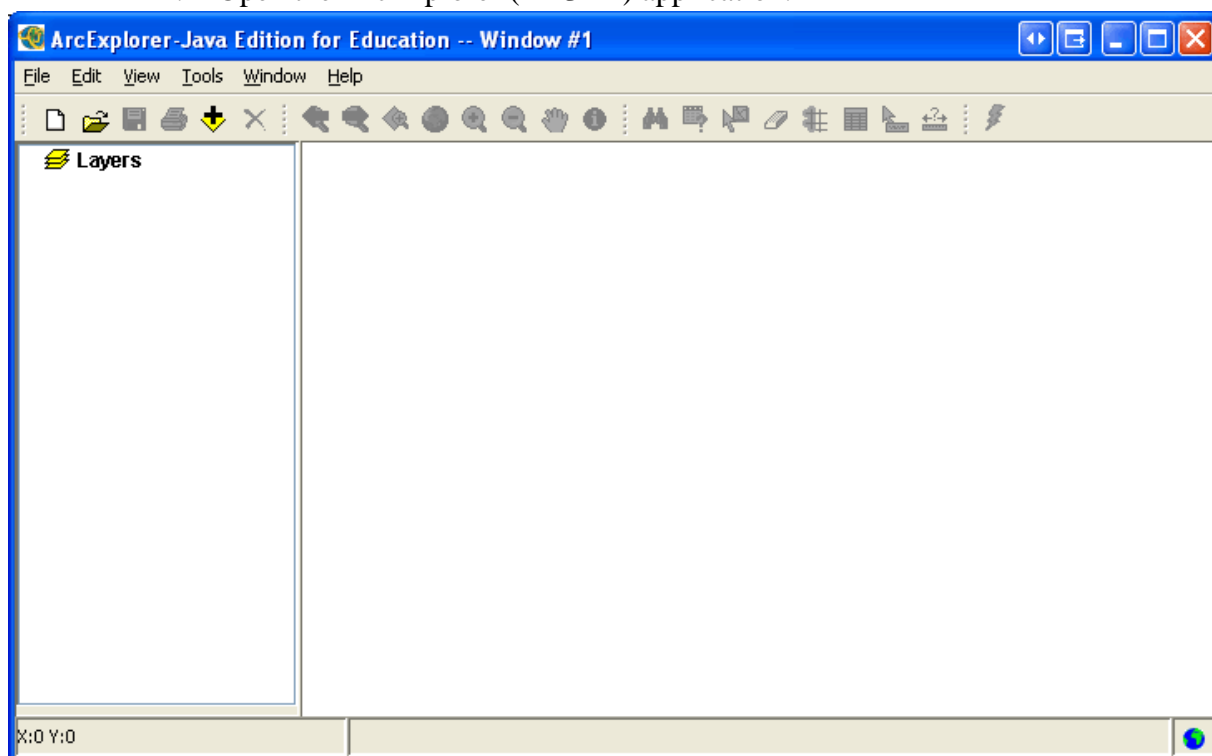
A good starting point is to load the `world_hd.axl` template.



If you use one of the demo AXL files supplied with ArcExplorer, it is necessary to open and save it first in ArcExplorer prior to adding it to a report for the first time. These demo AXL files are not *portable*, that is they must always be located in the one place on a user's PC. EMu requires them to be *portable*. A demo AXL file is made *portable* when it is opened in ArcExplorer and saved, either under its own name or a new name. For details of building your own template from scratch, see Making a custom AXL file (page 13).

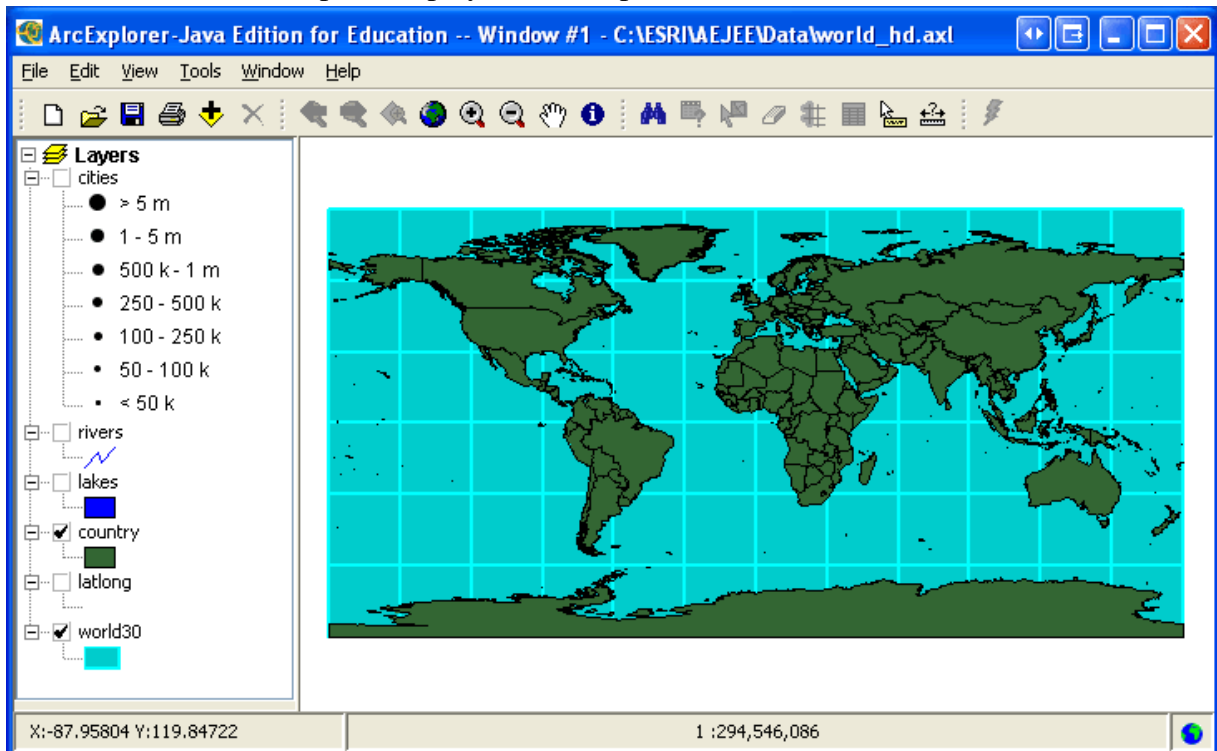
To make an ArcExplorer report using a demo AXL file:

1. Open the ArcExplorer (AEJEE) application.

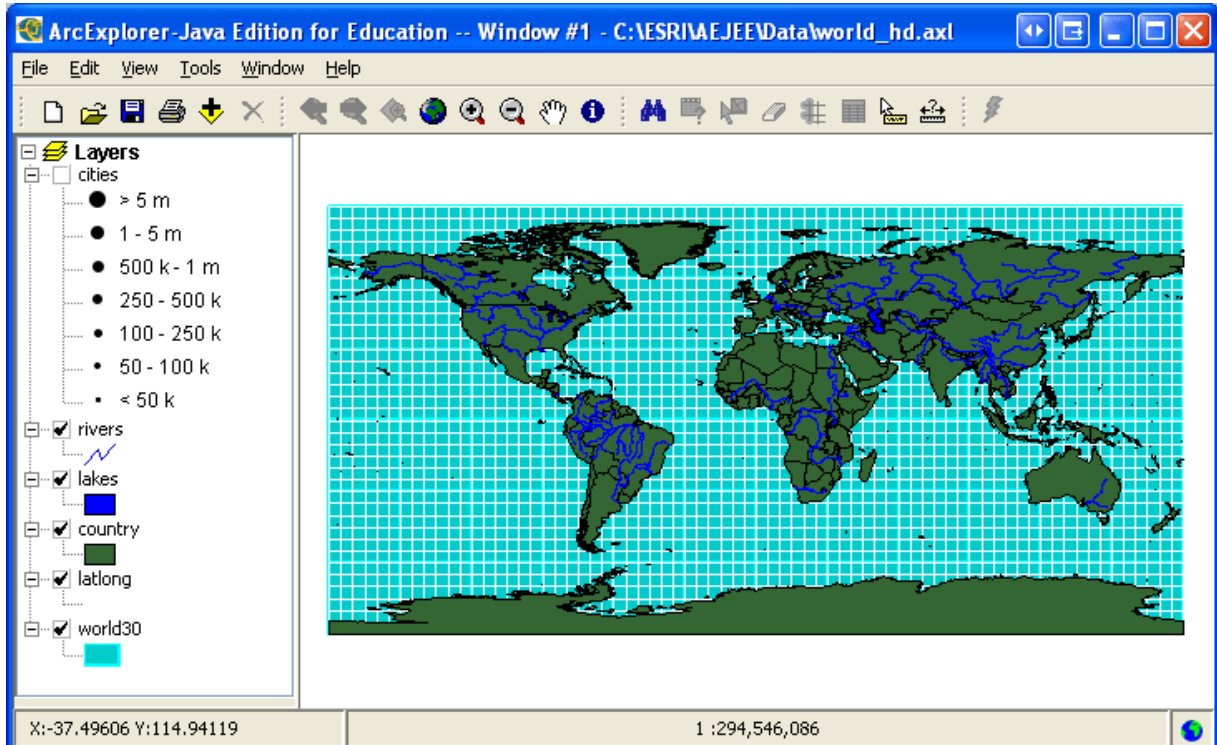


2. Open an AJEE template by selecting **File>Open**, navigate to ESRI/AJEE/Data and select the required AXL file.
In this example, `world_hd.axl` was selected.

The template displays in ArcExplorer:

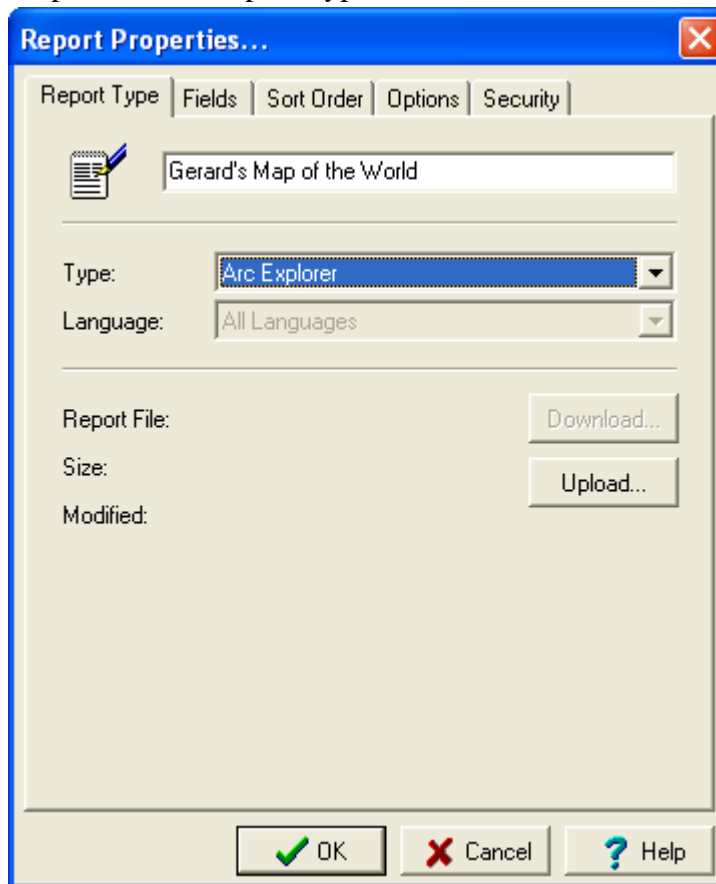



3. Modify the template as required, for example by adding layers:

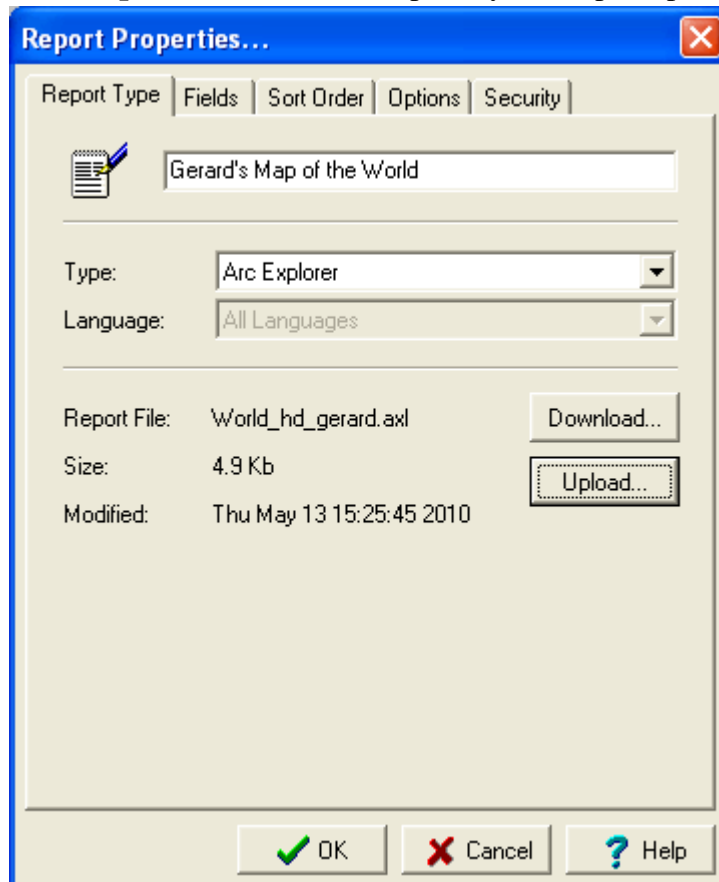


4. Select **File>Save As** to save the template without over-writing the original template. Be sure to use a descriptive name.
5. In EMu, create a new report in the appropriate module.
In this example the report is created in the Sites module.

6. Give the report a descriptive name and select **Arc Explorer** from the Type drop list on the Report Type tab:



7. Select **Upload**  to upload your map template:



8. Select the **Fields** tab and add fields to the report.

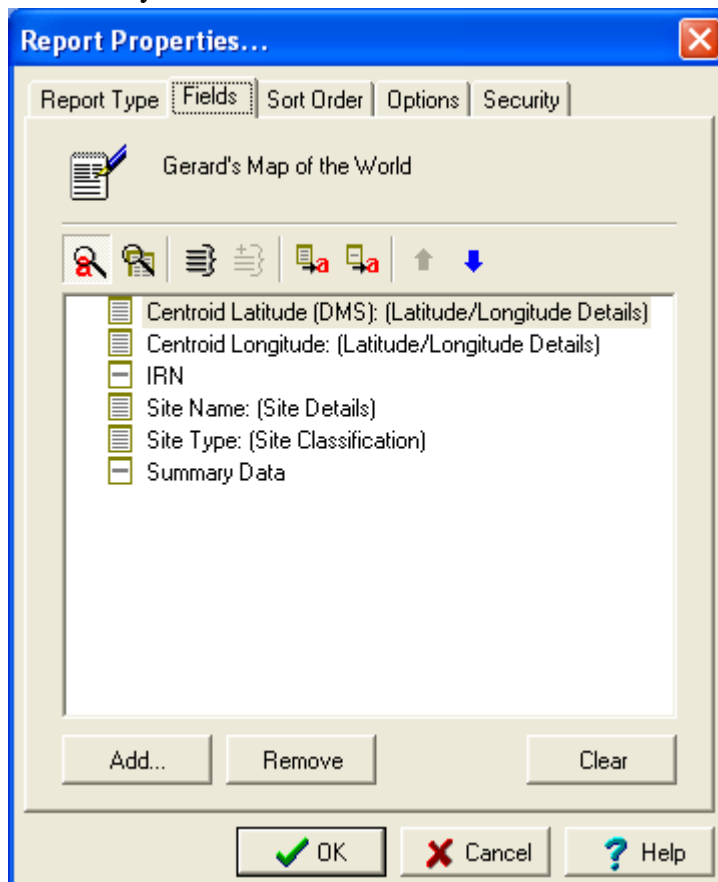


You must add one pair of Latitude / Longitude fields.

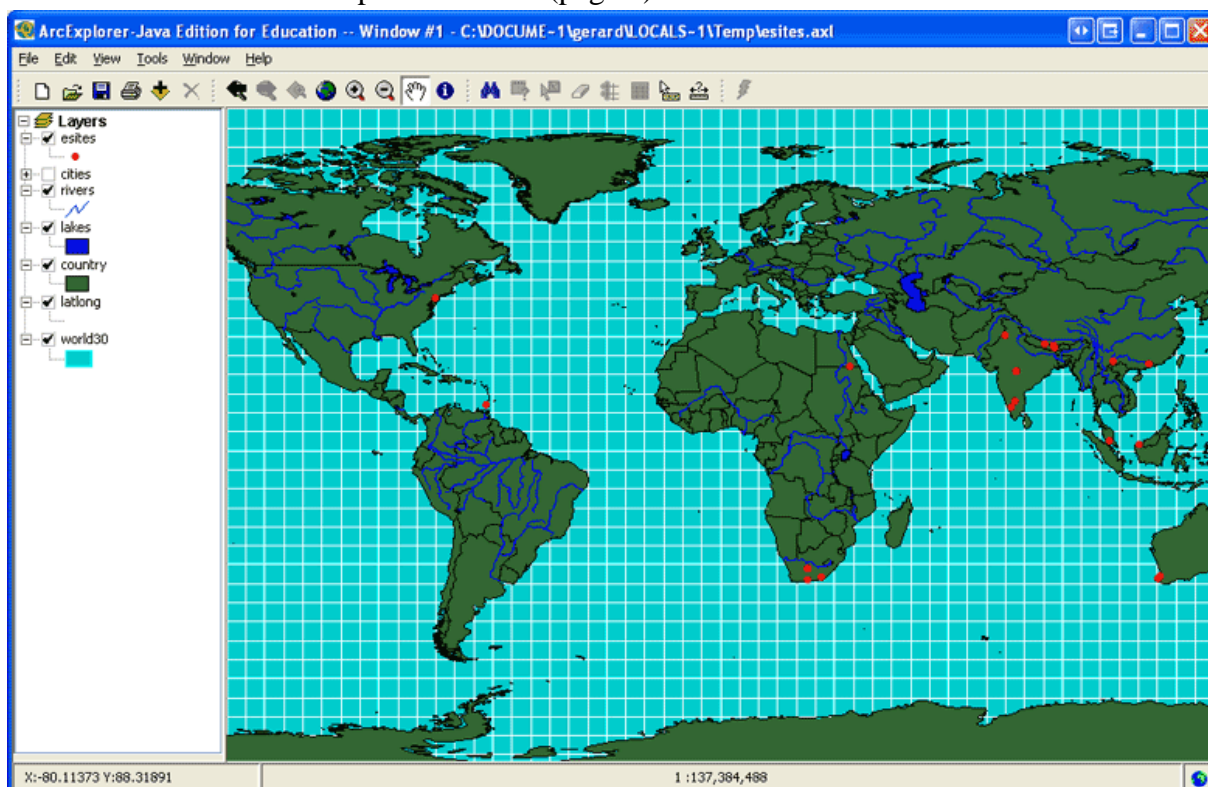
Other fields that would typically be added include:

- IRN
- SummaryData

- Family / Genus / Scientific Name


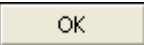


9. Save the report and run it (page 5):



Making a custom AXL file

ArcExplorer is provided with a wide range of shapefiles (layers) which you can use to build a template from the ground up.

1. Start a new session in ArcExplorer (**File>New**).
2. Begin adding shapefiles by clicking the **Add Data**  button in the Toolbar and navigating to `ESRI/AJEE/Data`.
3. From there, select one of the folders containing shapefile:
 - i. Select a shapefile.
 - ii. Click **OK**  to add it to your new template.
Add as many shapefiles as required.
4. Save the template with a descriptive name.
This generates an AXL file that can be used as the template for an EMu ArcExplorer report.

See Using an ArcExplorer AXL file (page 8) for details of building the EMu report.

Index

D

Download and setup ArcExplorer on your computer • 3

H

How to make the ArcExplorer report • 7

How to run an ESRI ArcExplorer report in EMu • 5, 14

How to set up EMu to run the ArcExplorer report • 3

M

Making a custom AXL file • 7, 14

T

The ArcExplorer Report • 1

U

Using an ArcExplorer AXL file • 7, 14