New Features and Changes in WebDB*2flex*™ version 2.*eight*

Security

The security model in 2.eight has been expanded to provide more precise control over what a user can or cannot access. The highlights of the changes include:

The user must now log in to WebDB2 prior to seeing the main screen. Once the user has been successfully authenticated by the system they are taken to the main screen where the menu of modules in the left-hand column will show only those modules for which the user has been granted access. The user never sees a menu item link to a module for which they do not have access.

Access is now controlled at the Table level rather than the Module level. For example, in a database covering water valves (where the general valve data is stored in the parent table and maintenance and replacement data is stored in two child tables), a user could be granted read only access the parent and replacement data while being granted read/write access to the maintenance table.

Access rights within a module or table are now more granular than in previous versions of the system. The right to “add new” records to a table is now a separate right. The ability to retrieve the original attached documents and images is also controlled through a separate right. Even access to jump buttons is controlled separately.

Read more about the enhanced security in the “Admin” section.

QuickView

While a user is in a module, either viewing or editing data, they can view another module in a new window by clicking on the “view other” button located adjacent to the “db prefs” button. Once the new window opens, a user can navigate to any module for which they have access rights and can even search within that module. Access through the QuickView window is read-only.

This feature can also be accessed through “lookup” buttons while editing. See the “Admin” section for more details.

Relay

Through properly constructed URLs, other web-based systems (such as GIS mapping systems) can open read-only views into specific records displayed in the standard WebDB2 format. These views are complete with “print” buttons that duplicate the environment provided by WebDB2 and, where appropriate, navigation buttons to “page” through child table data.

Control over what the “outside user” can see is very flexible. See the “Admin” section for more details.

Jump Buttons

Jump buttons have been expanded dramatically in version 2.eight. Highlights of these changes include:

Instead of one jump button per table, there are now up to four jump buttons per table. These buttons (referred to as “Jump1” through “Jump4”) are controlled by the corresponding “Jump1” through “Jump4” rights in the expanded security controls.

In addition to the table level jump buttons, each field in each table can have up to four unique jump buttons. These can be displayed in the list or/and form views of the records, and are controlled by the same “Jump1” through “Jump4” rights as table level jump buttons.

Jump buttons can now be displayed with custom text. For example, a button labeled “jump” in previous versions could now be displayed with the text “generate invoice”.

In previous versions of the software, jump buttons could pass the module name, table name, and record id of the current record in the generated URL query string. Now the contents of any field or fields may be passed in the generated URL query string. See the “Admin” section for more details.

User Preferences

In prior versions of the system, many user preferences were stored as “cookies” in the user’s local browser. Due to interactions with virus and spyware control software, this method had become unreliable. Starting with version 2.eight, all user preferences are stored on the server. As a result of this change, the user will need to reselect his or her preferences once 2.eight is implemented in their organization. The upside to this is that a user’s preferences will now “follow” them from machine to machine.

Other preference changes:

The old “denim”, “grass” and other themes have been removed from the system. Those themes relied heavily on passing background images to the browser on every access. To reduce network traffic, they have been replaced by 25 pastel color palettes that are available to the user in the “global prefs” section of the system.

The new pastel color themes alternate row “shades” for easier reading of lists. Gaps between rows and fields (which previously served to aid reading) have been removed. The net effect is a cleaner and more compact display of data.

The split of list/view frames preference is also now stored on the server. This setting can be saved by dragging the dividing bar to the desired location and then clicking on the new “Display Defaults” button (labeled “d”) in the list frame.

The maximum number of items per page in a list or report view has been increased to 250 in the “global prefs” settings.

In both the “global prefs” and “db prefs” pages, a button has been added to clear the user’s preferences and revert to the system supplied default values.

Browser Support

Only the following browsers are supported as of version 2.eight:

- Internet Explorer versions 6.x and 7.x on Windows
- Firefox version 2.x on Windows and Macintosh
- Safari version 2.x on Macintosh

In each case, only the currently shipping sub-version of each browser is supported. Earlier versions generally work, but will not be supported under any circumstances. Linux browsers are not supported at all.

Admin

Many of the changes in the Admin section of 2.eight relate to the overall changes described above. Reading the preceding sections will provide a good background for the notes that follow.

Security

In the area of security, the finer, more granular access controls created the potential for a huge numbers of settings per user. In addition, modules are no longer grouped into “security codes”. This means that each and every module requires distinct settings for each user. To avoid even more settings that would require input and maintenance by the administrator, rights can be defined initially for the parent table only, and cascade down to the child tables. Every child table, however, can have its own defined rights, including the “deny” right. (A child table that has the “deny” right set is never even seen or searched by the user.) Still, the number of combinations and permutations of settings was staggering.

To make this new scheme workable, the old WebDB2Admin.mdb file was scrapped and new “Users” and “Rights” tables were added to the WebDB2Control.mdb. After the install of 2.eight, these tables are created and populated by a one-time access to a conversion page. During conversion, rights are generated for each user that are consistent with their rights in previous versions. Still, the task of managing these rights can be daunting in a large WebDB2 installation, so the following features were added to the “View/Change User Rights” module in the “admin” section.

As before, you can view/change rights by user. On this page you can view or change the user data and roles, as well as grant or revoke rights. The layout of this page is very similar to the old security page. In the “data and roles” section, you can edit the user’s full name, assign a new password, or grant them one or more of the advanced roles. Advanced roles now include Power User, Security User, and Admin User. Power Users can bypass certain levels of “blockage” into modules just as they could in previous versions. A Security User can make changes to user security settings, add or delete users, but can not make changes to the structure of the database. The Admin User has full control over all functions of the system, cannot be “blocked” from a module, and can change the structure of the system from the web.

In the “rights” section, you can still grant and revoke rights on a module basis using the new finer granularity. Un-checking the “All Child Tables Inherit these Rights” box reveals the settings for the child tables. Each child table can then be set individually to inherit the rights of the parent, or have settings unique to its table. As noted earlier, “denied” child tables are removed from the user’s scope.

You can now copy a user’s rights (not roles) to any number of other users. One way to use this functionality would be to create a “prototype” user for a particular department or group, set the rights for that “prototype” user, and then copy those rights to members of that department or group. Note that such a copy completely replaces all rights of anyone receiving a copy from the “prototype” user. This functionality is not module specific. It globally replaces all rights for all modules and tables.

For module specific changes, you can now assign rights by module across any group of users. To use this feature, you select the module, set rights for that module, and then assign those rights to any number of users selected from the displayed list. As with the copy function discussed above, this rights assignment replaces all currently held rights for the chosen users, but only for the module being set. This is envisioned as a timesaver for the rights assignment process when (for example) a new module is added to the system.

QuickView

While most users will simply access QuickView from the “view other” button, a new feature in the configuration of editing “lookups” enables the use of QuickView in a unique way. In past versions, lookups either displayed the unique contents of the current field or the contents of a custom source table. In 2.eight, the lookup button can now open a new window to any URL. For example, a part number or description field could have a lookup button that opened a new window into a manufacturer’s or parts supplier’s site or even google. Although this new window will not automatically paste values back into the calling field, it could be very useful as a handy reference.

Using this lookup URL feature, QuickView can be called up with its full interface, or a more “configured” display. Using a URL like:

quickview:?module=db1&searchstring=what+to+find

QuickView can be called up in a new window showing the results of searching for “what to find” in module “db1”. Again, there is no auto-paste back into the calling field, but the reference opportunities are endless.

Relay

Relay opens an exciting array of possibilities for organizations seeking to provide links from other systems (especially GIS) back into the data store of WebDB2. Using a URL like:

http://Server-Name/WebDB2-Directory/WebDB2Relay.asp?module=db1&id=123

the external site can open a new window into record “123” of the module “db1” in a restricted read-only mode. The data is displayed just like it would be inside a WebDB2 session. Even the “print” button opens a new window and displays the data in a format suitable for hardcopy.

Several behind the scenes technologies work to make this possible. To begin with, a special user must be created that grants access to the requested module. In this example that user would be "db1_RelayUser". The rights granted to the "db1_RelayUser" control what this external user will be able to see. For example, if this user can read the parent table but only one of three child tables, the external user is given no knowledge of the other "denied" child tables. Depending on the granted rights, the external user might be able to see the previews of attached images but not be allowed to download the originals.

The rights of the "db1_RelayUser" are adhered to just as if they were logged in to a full WebDB2 session because they are. Relay logs in the user, generates the page, and immediately logs the user out. In addition, Relay blocks any access to the database that would allow changes to the data. A "_RelayUser" session has no write, add new, delete, prototype, batch edit, or move branch rights under any circumstance. Jump buttons respect their individual jump button rights. Export rights are meaningless since the export module cannot be accessed during a Relay session.

The fact that Relay logs the special user in and out provides another unique set of possibilities. The Admin User can log in interactively as the "_RelayUser", and set any combination of preferences for that user. Now that preferences are stored on the server, those preferences will be respected during page generation phase of Relay. In fact, the Admin can even set a default filter (maybe to control the child order sequence or to limit the display of old maintenance records) that will also be respected.

Jump Buttons

Prior to 2.eight, anyone with read access to a module (group) had access to the single jump button that could be configured per table. Security changes in 2.eight now allow for selective access to four levels of jump buttons. Jump buttons only appear if the user has the rights to use them. A full discussion of these new rights is beyond the scope of this document, but they do allow a great deal of flexibility in using the new Jump1, Jump2, Jump3, and Jump4 buttons either from tables or fields.

Now that multiple jump buttons are allowed, the old "one label fits all" model needed to be revisited. As of 2.eight, the graphic buttons used throughout the system have been replaced with CSS styled links. They still look like buttons, but now the text can be generated on the fly rather than in Photoshop. During configuration of the jump button, the administrator can supply the text that will be used for in two versions: one for lists and the other for form views. If no label text is provided, the system uses "1","2","3","4" as labels in lists and "jump1","jump2","jump3","jump4" as labels in form views.

As before, individual jump buttons can be configured to appear in list or form views, or both. In 2.eight, any table jump button can also be configured to appear in the list header.

Configuring jump buttons has been improved in the configuration editor by the addition of an "assist me" button. The admin supplies only the root of the desired destination URL for the jump button and the assistance form will build the required query string that enables the passing of field values to the destination. Those field values are passed by using embedded "tokens" in the supplied URL that are replaced at run-time with the actual field data. The "assist me" form eliminates needing to remember the syntax of the "tokens".

Miscellaneous Changes

The dreaded “Retrieving file…” window has been eliminated. Gone. Period.

The configuration of unique fields has been modified. Previously, a setting of “force unique” on a field would require that the entered value be unique to the table even if it was a field in a child table. New settings support this behavior, or a new behavior in child tables where a field value must only be unique to that related parent.

Prototype buttons which previously read “copy from prototype” can now have custom text.

When configuring default values for a field, an “assist me” button is available for the selection of the correct time/date token.

All changes to configuration information in the admin module now take effect when accepted. Security rights changes, however, take effect on the next user login.

WebDB2ModuleBuilder has been updated to recognize and co-exist better with custom configurations when setting virtual directories.