Fitrix 5.20 VDT
RELEASE NOTES

Installation Notes

Genero Desktop Client – Version 2.0:

In order to execute programs generated by the Genero 5.20 VDT, the user must have the Genero Desktop client installed.

*** Programs generated from the Genero 5.20 VDT tools no longer provide runtime support for the character version of the programs.

There is one exception-- when FGLGUI is set to '0' (console mode), report programs that do not require screen input will produce a report. This is necessary for reports that are 'croned' up to run at a later time. ***

The VDT tools themselves still execute in character mode.

Stored Settings
Developers should disable stored settings to ensure that window layouts are reset each time a program starts. Standard users may enable stored settings to retain changes they make to window layout. To enable/disable stored settings:
1. Start the Genero Desktop Client software
2. Click the icon at the top right of the window, with the ToolTip of 'Options'
3. Click the 'Advanced' tab
4. Under 'Stored Settings', check the 'Disable' option to disable stored settings, or uncheck it to enable stored settings.
5. Click the 'Apply' button.

Refer to the Genero Desktop Client - User Guide – Version 2.0 manual for more details.

Genero 4GL Server – Version 2.0:

Refer to the Genero Business Development Language - User Guide – Version 2.0 manual for more details.

Informix IDS Installation

The installation of Fitrix Complete Series Version 5.20 also installs the IBM-Informix IDS Version 10 database engine. (Fitrix Components Series does not)

Putty Configuration

The installation of Fitrix 5.20 includes default session templates for access to the Fitrix server via the Genero Desktop Client and Putty. Putty can define:

- Sessions needed to access the server via SSH, Rlogin, or Telnet as part of the user login.
- Sessions needed by developers to access the server outside of the GDC.

A Putty executable is included in the GDC installation directory, fgss_bin/putty.exe. Developers should create multiple Putty configuration sessions to support the wider screens now available in the screen generator. You should create profiles with the following characteristics:

Window Size		Font Size *
Rows	Columns	
24	132	9 pt
24	192	8 pt (wide)
24	256	5 pt (xwide)

* The font's point size will depend on the screen resolution of the monitor being used. For some monitors, an 5pt size may be unreadable.

See the document 'Screen Width – How to change for Genero' for more details.

New Features

Version 5.20 of the VDT offers new features taking advantage of FourJ's Genero ability to render forms in a windows environment.

Screen Width Changes

Screens can now be designed with widths that exceed 80 characters. In order to design a wider form, it is necessary to use one of the sessions discussed in 'PuTTy Configuration'. The wider sessions (either 132, 192 or 256 characters) will support screen definitions in those respective widths.

Windows Widget Support

New attributes have been added to input fields to render the fields with extra icons, which are processed as buttons, and perform additional business logic:

buttonedit – adds a 'lookup' button to a field. When pressed it fires an actions 'ac_zoom', which the generated program can use to display a zoom screen type.

dateedit – adds a 'calendar' image button to a date-type input field. When clicked, it displays a calendar, and allows the user to click on a date for return to the input field.

Debugging Functions

Version 5.20 of the VDT offers debugging features to assist the developer in troubleshooting programs, from both client and server perspectives.

4GL Debugger

The character-based 4GL debugger offers the same functionality as the BDL version. It is executed by starting a program with the following command:
$ fglrun –d xxxxx.42r, where xxxxx is the program name

AUI Tree Runtime Inspection

Developers can inspect the Genero Client Abstract User Interface tree structure while a generated program is executing. This can be a very helpful client-side debugging tool. To use this function the Genero Desktop Client must be started in 'Debug Mode', with the following additional parameter:
gdc.exe –D

Features Not Implemented Or Implemented Differently

Some features available in BDL are no longer available in the Fitrix 5.20 version of VDT.

Custom Actions

In BDL, generated programs supported user-driven additions of custom actions to the custom toolbar at runtime. This feature is no longer available. Custom actions can still be defined and attached to programs, but programs now require modifications to the generated programs to facilitate this.

Menu Permissions

Menu permissions are no longer managed by the x.prm files used by the 'mz' menu processor. These '.prm' files are loaded into Visual Menus and can be edited from within VM by an administrator by right-clicking on a menu option and selecting 'Security'.

User-level access to menu options can still be controlled by the programs in the security.4gm directory.

Conversion Tool

Screen specifications

Screens created with previous versions of the VDT Form Painter, or manually created screens used in previous versions of the screen generator are not compatible with the 5.20 Form Painter and Screen Generator. Conversion scripts are available to update older screens to the new specifications.

gn-pertr.sh xxxxx.per – Converts a named xxxxx.per file to the new format
gn-mass-pertr.sh – Converts all .per's in a given directory to the new format.

Non-VM Menus

When migrating from a version of BDL which has NOT been using Fitrix Visual Menus, a conversion script exists which will convert the menu options into a format compatible with Visual Menus. To execute the conversion script, execute the following commands from the Linux prompt:
1. company=standard; export company
2. DBDELIMITER='~';export DBDELIMITER
3. mz=$fg/accounting/menu; export mz
4. cd $mz
5. $fg/bin/menucvt.sh mainmenu
Any errors with the conversion can be reviewed in a log file in $mz/mncv.log

Other

VDT Tables

stxprogr – contains one row per program. Anytime a new program is created, an entry should be made in this table. This table is the basis for application-level security. If you create new programs, and want their security to be managed by the Fitrix security subsystem, you must add the program names to this table. The program which manages this table is security.4gm/i_progs.4gs. This table is not used by the new Visual Menu security mentioned above.

Environment Variables

mn_program_title – Stores the title of the program being called. It is displayed on the title bar of the window in which the program executes. If a developer executes a program manually, using the fglrun command, and this variable is NOT set before calling the program, the window title will display as 'Main Screen'.

New Execution Scripts

switch.sh – exists in /fitrix/bin, it supports switching from one database instance and name to another.

Syntax: `in={instance> db={database} . switch.sh`

The unusual syntax is required to pass arguments to a dot-executed (sourced) script.

Example:
```
$ in=prod db=live . switch.sh
```