

# OPERATIONAL ENHANCEMENTS

**NOTE: This section is of primary interest to SMMS system administrators and IS professionals.**

## Security

Administration of user security in Version 6 is greatly simplified from Version 5.2. This is because many programs have been consolidated into functional security areas, and users are authorized to access these security areas, rather than individual programs.

Ordinarily, these “functional security areas” are groups of related master files. For example, if the user is authorized to the Procedures file, he/she is also authorized to all of the databases related to Procedures: Tasks, Tools, Crafts, Notes, etc.



FIGURE 1

Figure 1 shows the new User Security Maintenance screen. This screen is used to authorize users to the master file database groups, and all of their relations.

Note that, as a result of user requests, SMMS now permits separate authorizations for the add, change, and delete update operations, as well as for inquiry and file data extracts to user PCs. This is intended to better meet the needs of clients with distributed reporting responsibilities in their maintenance operation.

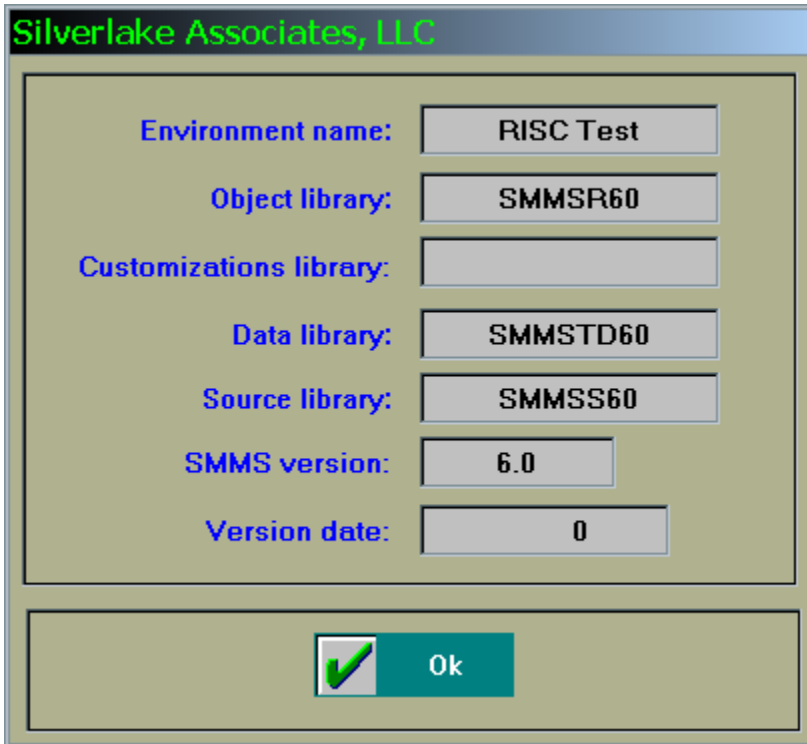
You can establish “default” user profiles for use with SMMS, and copy these default profiles to quickly set-up new users. You can also tailor individual users to suit their needs.

The screen shown in Figure 1 is also used to define user display defaults (file sort sequences and filters discussed in Figures 1a and 1b.)

### **Versioning and Environments**

Often, our clients like to install the SMMS software into a production environment for “live” work, and a separate environment for training and testing.

As a result of client suggestions, Version 6 displays the user name, company name and environment name at the bottom of the main screen. If your Company has a special user id for training, or uses special testing environment libraries, you and each user can quickly tell which library they’re operating from. (Refer to Figure 1, note 3 for an illustration of this feature.)



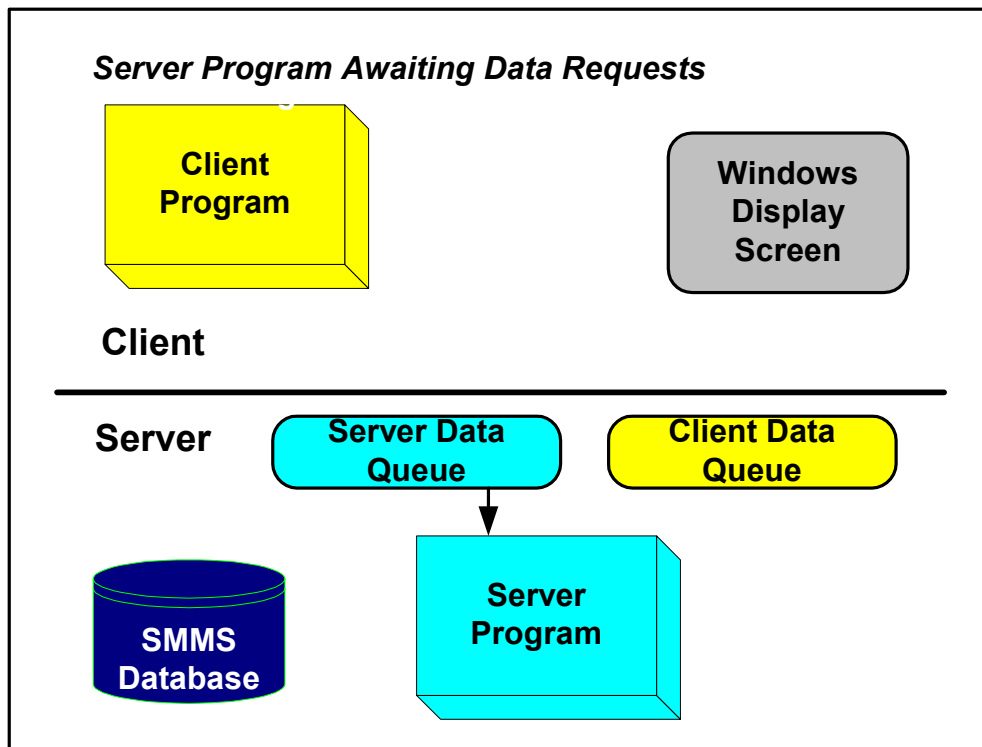
**FIGURE 2**

You can also display SMMS system version and environment information from each program, by making a selection from the Utilities menu. In addition to the current system version, this window (illustrated in Figure 2) shows the current library set for the named environment.

### **File Server Subsystem – Technical Overview**

The SMMS Version 6 File Server Subsystem combines high network response speed with the secure, reliable technology of the iSeries/400, using the “thin client” approach illustrated in Figures 3a, 3b & 3c.

The DB2/400 database server programs operate in subsystem SMMSIO, which is started with every IPL and checked periodically by SMMS user jobs. The database servers use data queues to exchange information with client programs. It does not matter whether these client programs are running on a PC workstation or in the “green screen” environment.



**FIGURE 3A**

Figure 3a shows the system in a “wait state”. The server program is a “never ending” program running on the iSeries/400. It receives data requests from all client programs via the “Server Data Queue”. There is one server program for each database file in the SMMS database.

The server program also enforces SMMS business rules. All client requests to add, change or delete data are verified by the server program before the database update can take place. This ensures that database validations will be consistently enforced.

Figure 3b (below) shows what happens when the client program requests data. The client program sends a request to the server program, via the server’s data queue. As soon as the client program sends the request to the server, it monitors it’s own data queue waiting on data to be returned from the server. Meanwhile, the server program receives the data request and gets the data from the database.

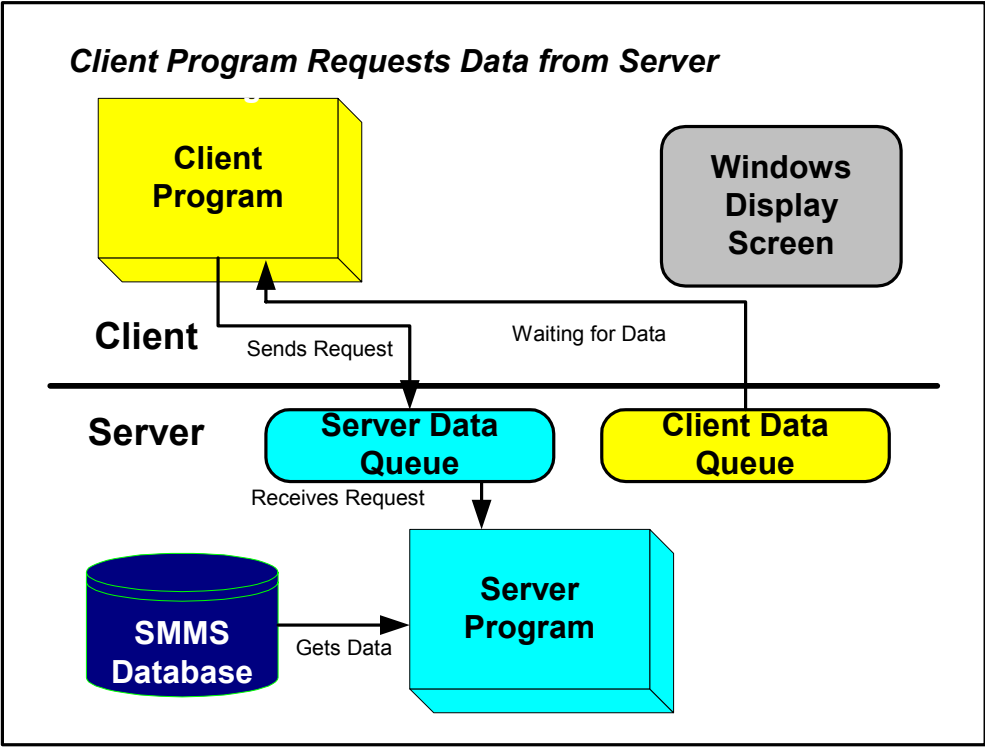


FIGURE 3B

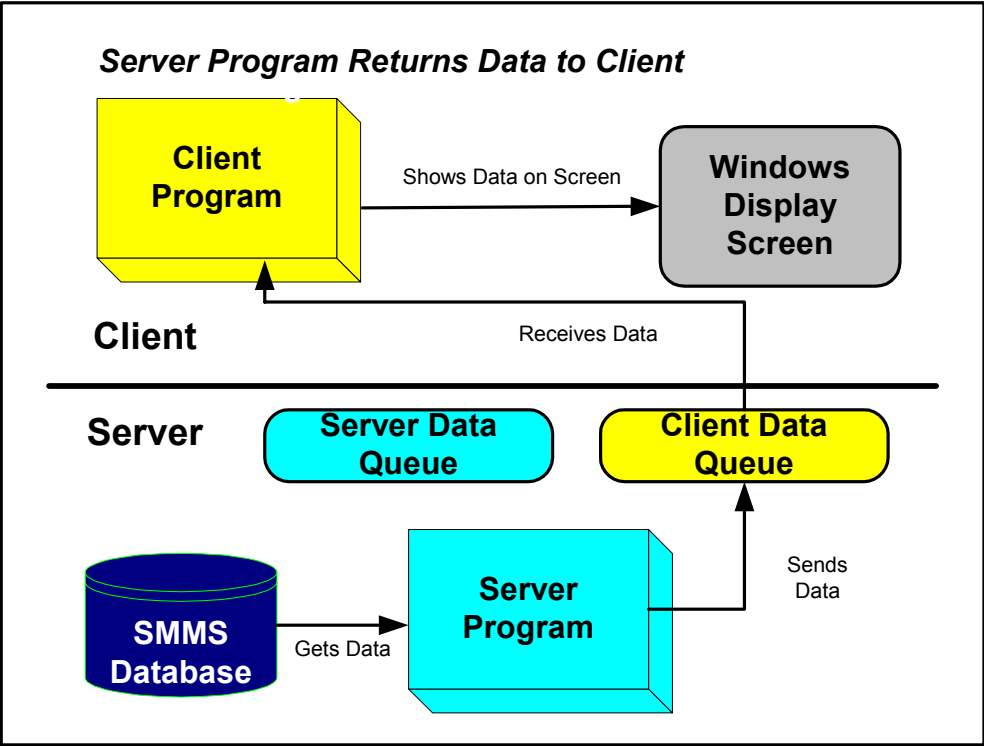


FIGURE 3c

Figure 3c shows what happens when the server program responds to a data request from the client. The server program receives the information request via its data queue, and fetches the requested data from the database. The data is returned to the client program via the client program's data queue. As the client program receives the data, it is able to display the data onto the screen.

The result is that all client programs --- green screen or windows-based --- are served data on a "first come, first served" basis. System business rules are enforced equally, as the server programs also perform edits and validations before database changes are allowed, again whether the source of the data is "green screen" or windows.