

Blue Moon Software Developer Kit **Guidelines for Integrating with the Blue Moon Forms Software**

The Blue Moon software includes an import feature, which allows you as a software developer to transfer data directly from your application into our Blue Moon database. The import feature provides two methods of accomplishing the data transfer. The first method is through the use of an external disk file. The second is through the use of the Windows Clipboard. Once the data to be imported is written to a file or copied to the clipboard, Blue Moon is launched, using the appropriate command-line parameters. The details of each method are described below.

Note: These instructions and sample applications help you if you want to launch Blue Moon directly from your windows application. Alternatively, you can simply create a text file from you application and have your user manually lauch Blue Moon.

What is included in the SDK

The Blue Moon SDK installs a version of our Texas Forms Software (WinTAA) demo. Also included are sample applications as well as source code written in Visual Basic, Delphi, and C. The sample source code will work with all of our other software products in other states.

The Blue Moon Database

All of our products, except for WinTAA, use a basic apartment database. In Texas we also support a residential database and a condo/townhouse database. The Blue Moon Database Engine uses a proprietary data structure. Each database consists of three files, a data file (*.DB), an index file (*.IX), and a header file (*.DBS). A user can have multiple apartment property databases, which are named APTDB1.*, APTDB2.*, APTDB3.*, and so forth. In Texas, there could also be a residential database (RESDB.*), and a condo database (CONDODB.*).

Each database has about 100 fields. Blue Moon lets you set up all but fourteen fields as a default. Hence, every time you create a new lease, the default lease will automatically generate most fields. The remaining fourteen fields change from lease to lease, and those are the ones we import from an external source.

The Import Data Format

Both the external file method and the clipboard method use the same file structure. If you are going to generate a file, we recommend you give it a *.TXT extension. This format contains the following fourteen fields:

<u>FIELD</u>	<u>TYPE</u>	<u>MAXIMUM LENGTH</u>
UNIT NUMBER	STRING	8
RESIDEN1	STRING	25
RESIDEN2	STRING	25
RESIDEN3	STRING	25
RESIDEN4	STRING	25
OCCUPAN1	STRING	25
OCCUPAN2	STRING	25
OCCUPAN3	STRING	25
OCCUPAN4	STRING	25
BEGLEASE	DATE*	8
ENDLEASE	DATE	8
RENT	NUMBER**	8
PRORATED	NUMBER	8
SECURITY	NUMBER	8

* Date fields must be created using a mm/dd/yy format.

** Numeric fields are formatted with two decimal places and no commas (e.g. 1000.00)

The data for each field is placed in double quotes. The fields are all listed on one line, comma-delimited, in the above order. Here is an example of correctly formatted data:

"106","Resident 1","Resident 2","Resident 3","Resident 4","Occupant 1","Occupant 2","Occupant 3","Occupant 4","12/10/93","11/30/94","1325.00","1000.00","1325.00"

Due to the constraints of this document, we could not actually show the above example all on one line.

Launching Blue Moon

Two steps must be taken to properly launch Blue Moon. First, you must retrieve the application name (Exe Name), and the data/import path from the Windows Registry. These values are all stored in the "HKEY_LOCAL_MACHINE\ SOFTWARE\ Blue Moon" key of the Registry.

You can retrieve the Windows Registry information by calling the following Windows API functions. Here is an example written in C:

```
FILE *f;
char ExeName[100], DataPath[100];
long lRetVal;          /*result of the API functions*/
HKEY hKey;            /*handle of opened key */
long size;
long type=REG_SZ;

ExeName[0] = 0;
DataPath[0] = 0;
```

```

    lRetVal = RegOpenKeyEx(HKEY_LOCAL_MACHINE, "Software\\Blue Moon", 0,
KEY_QUERY_VALUE, &hKey);
    if (lRetVal == 0)
    {
        size=100;
        lRetVal = RegQueryValueEx(hKey, "Exe Name", 0, &type, ExeName, &size);
        size=100;
        lRetVal = RegQueryValueEx(hKey, "Import Path", 0, &type, DataPath, &size);
        if (DataPath[0] == 0) //If import path is not defined, default to data path
        {
            size=100;
            lRetVal = RegQueryValueEx(hKey, "Data Path", 0, &type, DataPath, &size);
        }
        RegCloseKey (hKey);
    }

    if (ExeName[0] == 0)
    {
        printf("Blue Moon is not installed.\n");
        exit(1);
    }
}

```

Then, you must call the WinExec() Windows API function. Here is an example written in C:

```

char cmd[200];
sprintf(cmd,"%s /imp=export.taa",ExeName);
WinExec(cmd, SW_SHOW);

```

See the following sections for details on the necessary parameters needed for each data transfer method.

Using the External File Method

To use the external file method, create a file using the format described above and place the file in the Data/Import Directory. You will need to know the location of the user's Data (normally this is in their Main Path+\Data. C:\WinTAA\Data, for example.) If you have a Windows application, we recommend you use the Clipboard method (discussed below). Once the file is written to the Data/Import Directory, start Blue Moon by calling WinExec() and pass the following parameter, where filename is the name of the file:

/imp=filename

This will cause the data contained in the file specified by filename to be imported into the default database (usually APTDB1). If you would like the data to be placed in a database other than the default, you may also pass the following parameter, where dbname is the name of the database to which you want the data imported:

/db=dbname

The dbname parameter refers the DOS filename of the database as it appears in the user's Data directory. No extension should be supplied.

Here is an example of a command-line which will import the file TEST.TXT, located in the Data directory, into the user's APTDB2 database:

```
WinTAA.EXE /imp=TEST.TXT /db=APTDB2
```

(The name of the executable could differ depending on which version of Blue Moon you are using.)

To manually import the file if you already have Blue Moon open, first select the appropriate database (an apartment property, residential, or condo database). Next, go to the File pulldown menu and select Import from File. You will see all the *.TXT files in your data directory. Select the file to import. Blue Moon will automatically find the unit in the database (if it does not exist it will create it for you) and will import the data. Once the data is imported Blue Moon will delete the file from the data directory.

Using the Windows Clipboard Method

To use the Windows Clipboard method, create a string using the format described above and copy it to the Windows Clipboard. Once the data is copied to the Windows Clipboard, start Blue Moon by calling WinExec() and pass the following parameter:

```
/imp=<CB>
```

This will cause the data contained in the Windows Clipboard to be imported into the default database. If you would like the data to be placed in a database other than the default, you may also pass the following parameter, where dbname is the name of the database to which you want the data imported:

```
/db=dbname
```

The dbname parameter refers the DOS filename of the database as it appears in the user's Data directory. No extension should be supplied.

Here is an example of a command-line which will import the data on the Clipboard into the user's APTDB2 database:

```
WinTAA.EXE /imp=<CB> /db=APTDB2
```

(The name of the executable could differ depending on which version of Blue Moon you are using.)

NOTE: If you are already running Blue Moon, you will want to copy the data to the clipboard without launching Blue Moon again. From Blue Moon you can use the "Import from Clipboard" function in the File pulldown menu to import your data.

To manually import the data if you already have Blue Moon open, first select the appropriate database (an apartment property, residential, or condo database). Next, go to the File pulldown menu and select Import from Clipboard. Blue Moon will automatically find the unit in the database (if it does not exist it will create it for you) and will import the data. Once the data is imported Blue Moon will delete the file from the data directory.

The Sample Programs

We provide you with sample source code for Visual Basic, Delphi, and C in the SDK to demonstrate the use of the import function.

The C sample source code is located in the SDK\C directory. This program demonstrates the use of the external file method in C. The program was written as a Win32 console application, but the code should be compatible with any Windows-based C compiler. To open the project, you must use Microsoft Visual C++ Version 5.0 or later.

The Delphi sample source code is located in the SDK\Delphi directory. This program demonstrates the use of the Clipboard method in Delphi 4.0 or later, using Borland's TRegistry object (which simplifies registry access).

The Visual Basic sample program is located in the SDK\VB directory. This program is appears almost identical to the Delphi sample, except it is written in Visual Basic. In addition, since there is no TRegistry object in VB, it uses code similar to the C example. This program also uses the Clipboard method. Visual Basic 5.0 or later must be used to open this project.

Each demo comes with a precompiled executable, so you will not be required to have a compiler to see the code in action.