

# **Blackbaud FundWare iData Installation and Technical Guide**

**VERSION 7.60, SEPTEMBER 2009**

**Blackbaud<sup>®</sup>  
FundWare<sup>™</sup>**

## **TECHNICAL GUIDE HISTORY**

<b>Date</b>	<b>Changes</b>
January 2002	Initial release of iData, Version 1.
July 2002	7.17—Includes edits correcting minor typographical errors.
December 2002	7.20—Includes edits correcting minor typographical errors.
February 2003	7.20 Service Pack 1—Includes minor modifications to the program; reissued with Service Pack 3 (address change).
May 2004	7.30—Includes edits correcting minor typographical errors.
December 2004	7.31—The guide has been edited to reflect new company information.
November 2005	7.40—The guide has been updated with minor corrections.
July 2008	7.50—This guide has been updated with minor corrections and new company information.
September 2009	7.60—The guide has been updated to incorporate 7.60 functionality and to reflect new company information.

© Copyright 2009, Blackbaud FundWare, Inc. All rights reserved.

The Blackbaud FundWare logos, among others, are registered trademarks and/or registered service marks of Blackbaud Fundware, Inc. in the United States and other countries. Other parties' trademarks or service marks are the property of their respective owners and should be treated as such.

# TABLE OF CONTENTS

<b>Chapter 1—Introduction</b> .....	<b>1</b>
How Does iData Work? .....	1
iData Administration .....	2
Smart Views .....	2
Adaptations to Divisional Setups .....	3
Data Dictionaries .....	3
Getting Support .....	3
<b>Chapter 2—iData Business Rules</b> .....	<b>5</b>
Data Scrubbing Rules for All Views .....	5
All Fields .....	5
Text Fields .....	6
Numeric Fields .....	6
SQL Server 2000 Date/Time Fields .....	6
Illegal Characters in the Column Names .....	7
Configuration Rules .....	8
General Ledger–Project/Grant Views .....	8
vwGLAccount .....	8
vwGLTransaction .....	9
vwGLBalance .....	10
<b>Chapter 3—Configuring Blackbaud FundWare for iData</b> .....	<b>11</b>
Account Dimensions .....	11
Journal References .....	11
Data Items .....	12
<b>Chapter 4—Configuring SQL 2000</b> .....	<b>13</b>
Starting the SQL Server Agent .....	13
Adding SQL Server 2000 Logins .....	17
Modifying SQL Server 2000 Logins .....	20
Deleting SQL Server 2000 Logins .....	21

Installing iData and Blackbaud FundWare to Separate Servers . . . . .	21
<b>Chapter 5—Configuring and Administering iData . . . . .</b>	<b>23</b>
Opening The iData Administration Tool . . . . .	23
Entering Blackbaud FundWare and iData Information . . . . .	25
Scheduling a Build . . . . .	29
Overriding a Scheduled Build . . . . .	31
Entering the General Setup Information . . . . .	32
iData Security . . . . .	33
Assigning iData Security Settings . . . . .	33
Copying Users to Security . . . . .	35
Copying Security Settings From Another Division . . . . .	36
Replicating Security Settings From Another Division . . . . .	37
Removing Security Settings . . . . .	38
Viewing the Transaction Logs . . . . .	39
Closing the iData Administration . . . . .	41
<b>Chapter 6—Data Dictionaries . . . . .</b>	<b>43</b>
GL Accounts . . . . .	44
GL Transactions . . . . .	46
GL Balances . . . . .	51

**Appendix A—Adaptable Account Dimensions . . . . . 57**

**Appendix B—Adaptable Period Information . . . . . 59**

**Appendix C—Adaptable Journal References . . . . . 61**

**Appendix D—Adaptable Data Items . . . . . 63**

**Appendix E—SQL Server 2000 Data Types . . . . . 65**

**Appendix F—Warning and Error Messages . . . . . 67**

**Index . . . . . 77**



# CHAPTER 1

## INTRODUCTION

The Blackbaud FundWare™ intelligent Enterprise tool, iData, was created with your reporting needs in mind, allowing you to analyze and report smarter with the power of Microsoft® SQL Server 2000. iData works the way you work—using familiar terms, accessible by standard tools such as Microsoft® Excel, Crystal Reports, and Microsoft® Access. FundWare has designed “Smart Views” to eliminate many of the complicated logical expressions found in reports. The data you request is dynamically pulled into organized, accurate Smart Views that are easily interpreted and conform to your unique system configuration. iData has done the work for you, eliminating the need to decipher complicated, time-consuming data dictionaries.

iData allows immediate access to all data posted to the core FundWare system, with the power of SQL Server 2000. On a user-defined, scheduled basis, your FundWare data is extracted from the core system and loaded into Smart Views within SQL Server 2000. Users can define scheduled extraction and administer security through the iData Administration utility.

Because iData is organized into unique Smart Views, most reports can be derived from a single view. This eliminates the need to link complicated database tables, or hunt between various tables for related data.

Included in this guide is information for the installation, configuration, and administration of iData, as well as technical considerations employed by various iData routines. This guide does not address the actual use of the data created by iData.

### HOW DOES IDATA WORK?

The following sections describe the various parts of iData that combine to allow you to work most efficiently with your data.

## **IDATA ADMINISTRATION**

The iData administration utility provides the following configuration and maintenance features:

- **Data Paths/Servers**—iData can be located on a separate server from the FundWare application server (this is the recommended configuration) or on the same server as FundWare. The paths and server options provide this flexibility. This maintenance feature also allows you to choose what FundWare divisions to include in iData.
- **Scheduling**—This option allows you to schedule when and how often iData will begin the build process.
- **General Setup**—The General Setup utility allows you to complete the administration tasks (e.g., setting the user name and password).
- **Process Log**—This log contains historical information for each iData build, such as Build Date, Division ID, # of Errors, # of Warnings, and Duration. The administrator can expand on any record to see detail about the build when trying to troubleshoot errors or warnings. A text file is provided that contains information about the last build. An icon can be created on the administrator's desktop to view this file.
- **Security**—iData provides a hierarchical security mechanism. Through an easy-to-use graphical interface, the administrator can assign user security access to a Smart View, all Smart Views within a module, or all iData Smart Views. Security setups can easily be copied from one division to another.

## **SMART VIEWS**

iData extracts related business and financial data stored in many different database tables and combines the information into one table, called a “Smart View.” iData further enhances the Smart View to incorporate various reporting features. The following Smart Views are provided for General Ledger:

- **vwGLAccount**—Detailed chart of account information stored in your FundWare system.
- **vwGLTransaction**—Detailed General Ledger transactions that have not been purged from your FundWare system.
- **vwGLBalance**—Detailed record for each defined accounting period for each account number defined in your chart of accounts.



## ADAPTATIONS TO DIVISIONAL SETUPS

iData automatically configures each Smart View based on the FundWare setup defined within a division. In any view where an account number is represented, additional columns are automatically created for each of the account number's dimensions and dimension descriptions. iData automatically creates columns of information for each unique journal reference value or data item defined within FundWare that is associated with any data value. Columns for unused references or data items are not created.

## DATA DICTIONARIES

iData contains a static data dictionary for each Smart View. This data dictionary generally describes the static and variable structure of your FundWare data. The variable structure is dependent on your divisional setup, such as the number of ID types defined, the number of dimensions within each ID type, the number of different journal references and data items used. These dictionaries can be found in *Chapter 6—Data Dictionaries on page 43*.

The dynamic data dictionary, provided as another Smart View called “iDataDictionary,” details the exact layout of each Smart View based on the divisional setup. The iData Dictionary can be viewed or reported with the same third-party tool used to access other Smart Views.

## GETTING SUPPORT

The following table provides a quick reference for getting the support you need for FundWare modules.

*Table 1.1—Getting Support*

Department	Contact	Description
Business Office	800-551-4458	Obtain general information or contact a staff member.

Table 1.1—Getting Support (Continued)

Department	Contact	Description
Technical Support	800-227-7575 e-mail: FWSupport@blackbaud.com	Get help resolving technical issues with the product. Please have the following information ready when you call: <ul style="list-style-type: none"> <li>• Blackbaud Site ID</li> <li>• FundWare version number</li> <li>• Error message details</li> </ul>
	Knowledgebase www.blackbaud.com	A searchable database where you can type your questions and get a list of possible answers. <ul style="list-style-type: none"> <li>• Requires your license number and Blackbaud Site ID.</li> </ul>
Training	800-551-4458 Ask for the Client Sales department.  www.blackbaud.com	Contact us for more information on our training options: <ul style="list-style-type: none"> <li>• Regionals—Classes from 1 to 5 days in length; held in cities around the country.</li> <li>• eClasses—Two- to three-hour classes offered over the Internet.</li> <li>• Onsite custom training and consulting—Our consulting staff can implement FundWare and/or train your employees using your organization's data.</li> </ul>
Product Management	www.blackbaud.com	Contact our Product Management department with any suggestions for improving FundWare.
All other information	www.blackbaud.com	Links are available for all other FundWare information including replacement software, user guides and software updates.

**Blackbaud, Inc.**  
6430 S. Fiddler's Green Circle, Suite 200  
Greenwood Village, CO 80111  
800-551-4458  
303-756-3030

## CHAPTER 2

# IDATA BUSINESS RULES

This chapter outlines many of the business rules used to transform your FundWare data into Smart Views. To successfully work with these business rules, you must have a general knowledge of your organization's financial data and FundWare setup.

There are two types of rules discussed in this chapter:

- **Data Scrubbing Rules**—Illegal “column name” characters found in user-defined setups are replaced or removed prior to use. Other data scrubbing includes, but is not limited to, removing leading or trailing blank spaces and Null values.
- **Configuration Rules**—Smart Views are configured according to a set of business rules designed to interpret the FundWare divisional setup as defined by the user.

This chapter is organized by rules that apply to all views, followed by rules that apply to a specific view.

## DATA SCRUBBING RULES FOR ALL VIEWS

The following sections define rules for data scrubbing the text, numeric, and SQL Server 2000 data/time fields. These rules apply to all Smart Views.

### ALL FIELDS

- **Null Values**

SQL Server 2000 databases introduced a concept called Null, or indeterminate state. This means that the data in the field has no value. When searching or limiting your view of data, you must take the null condition into consideration. iData has simplified this process by eliminating nulls from the data, with the exception of Date/Time fields. Text fields are initialized with blanks (“”) and numeric fields are initialized with zeroes (“0”). Date/Time fields within SQL Server 2000 have special validation parameters and blank dates are not valid. ISAM date fields that contain a blank or 00/00/0000 are converted to null.



When limiting your view of data based on a SQL Server 2000 time/date field, you may want to include “ISNULL” or “NOT ISNULL” in your criteria. Use the proper Null/Not Null expression based on your reporting tool.

## TEXT FIELDS

- Remove double quotes ("), single quotes (‘), and commas (,).
  - Double-quotes are found in many description fields used to highlight information contained within the description. This can cause problems for clients who export iData information and then try to import it into another application using a delimited format. In some instances, the double quotes also interfere with various iData routines.
  - Single-quotes, or apostrophes, can cause problems with SQL Server 2000 statements used by iData.
  - Commas often act as delimiters in data, and can cause problems with iData routines or third-party applications accessing the data.
- Remove leading or trailing blanks.
  - Blank characters are trimmed from the front or back of any text field. Some data is stored with padded blanks that will give unexpected results in reports or query searches.

## NUMERIC FIELDS

- Remove dollar signs (\$) and commas (,).
  - Dollar signs can cause problems with SQL Server 2000 statements used by iData.
  - Commas often act as delimiters in data, and can cause problems with iData routines or third-party applications accessing the data.

## SQL SERVER 2000 DATE/TIME FIELDS


- Set time component for ISAM dates.
  - ISAM date fields have no time component. SQL Server 2000 does not have a date field, it only has a date/time field. When ISAM dates are loaded into SQL Server 2000, the time is automatically set to 12:00 am.



When limiting your view of data based on a SQL Server 2000 time/date field, be sure to include the time component. For example, 3/14/2001 to 3/31/2001 will not return the expected data set. The proper syntax is 3/14/2001 12:00:00 AM to 3/31/2001 11:59:59 PM.


## ILLEGAL CHARACTERS IN THE COLUMN NAMES

For journal references, data items, and account dimensions, iData automatically creates columns of information based on the applicable user-defined setups. In most cases, the column name is based on the short description given the particular data element.

 For example, when each account dimension was defined, it was given a short description. iData assigns that short description as the column name.

However, certain characters or names used can cause conflicts in SQL Server 2000 or iData. To avoid these conflicts, iData automatically resolves these issues according to the following rules:

- Legal character values—iData will create column names that contain characters in the range of A through Z, a through z, 0 through 9, underscore (\_), pound sign (#), and dollar sign (\$).
- Illegal character replacement—Illegal characters are often replaced with an underscore (\_) or simply removed from the column name. The underscore is the preferred method because it makes the replacement more obvious.
- Column names must not begin with a number—SQL Server 2000 does not allow a column to begin with a number. In this case, the number is replaced with an underscore (\_).
- Other characters replaced—There are some characters that are valid in SQL Server 2000, but are invalid in tools such as Microsoft Access (which might be a common tool to access the iData databases). Such characters include, but are not limited to, period (.), exclamation (!), accent grave (`), or brackets ([ or ]). These characters are replaced by an underscore (\_).
- Duplicate column names due to character replacement—iData will automatically resolve duplicate column names created by character replacement by appending “\_Dup2”, “\_Dup3” and so on, for each additional duplicate column name. Change the data item short description to avoid this situation.

 For example, consider the following data item short descriptions, “Oblig.” and “Oblig!”. In both cases, the illegal characters period (.) and exclamation point (!) are replaced by an underscore (\_). This causes two columns of different data to have the same name, which is illegal in SQL Server 2000. iData would resolve this particular problem by creating “Oblig\_” and “Oblig\_Dup2”.

- Reserved description—At this time, the only description in FundWare that causes a problem for iData is the account dimension name “Account”. Having a dimension with this name will cause the iData build to fail.



You must rename this description prior to building iData.

- Blank short description—Short descriptions for journal references, data items, or dimensions are used to name the appropriate column. Therefore, these entries cannot be blank.

## CONFIGURATION RULES

- Data As Of—When the iData build begins, the start date and time is recorded. This value is written to the Data As Of field for each Smart View.

## GENERAL LEDGER–PROJECT/GRANT VIEWS

There are three General Ledger–Project/Grant Smart Views available:

- vwGLAccount,
- vwGLTransaction, and
- vwGLBalance.

All related data in the core FundWare system is available in these three views.

## VWGLACCOUNT

The vwGLAccount Smart View can be divided into four components as follows:

- Account component—Fixed and based on data found in the GL account file.
- Account dimension component—Variable and based on information contained in the FundWare divisional setup.
- Attribute component—Fixed and based on the FundWare divisional setup.
- Data item component—Variable and built only if the client has attached data items to GL accounts.

vwGLAccount adheres to the following configuration rules:

- **Account Dimensions**—In any view that contains an account number, a column is created for each ID Type/Dimension Name combination defined in the GL setup. Each column is preceded with the ID Type, for example, A\_Object. See *Appendix A—Adaptable Account Dimensions on page 57* for a detailed description on this process.
- **Data Items**—Data items can be attached to account numbers. For each data item defined that has a value attached, a column is created for that data item. Data items defined, but not used, will not appear in iData until a value has been assigned. See *Appendix D—Adaptable Data Items on page 63* for a detailed description on this process.

## VWGLTRANSACTION

The vwGLTransaction Smart View can be divided into four components:

- **Transaction component**—Fixed and based on information contained in the Transaction Detail file.
- **Account/dimension/period component**—Variable and based on the FundWare divisional setup.
- **Transaction reference component**—Variable and based on the FundWare divisional setup.
- **Data item component**—Variable and based on the FundWare divisional setup; built only if the client has attached data items to GL transactions.

vwGLTransaction adheres to the following configuration rules:

- **Transaction amount**—The transaction amount, units, and rate are automatically placed in the appropriate column of Actual, Budget, Commitment, or Budget based on the transaction type.
- **Quarter date**—This column is automatically calculated from the posted transaction date.
- **Accounts/account dimensions/period information**—The transaction record in iData will contain columns for each defined ID type defined in your GL setup. A transaction can post to more than one account. See *Appendix A—Adaptable Account Dimensions on page 57* and *Appendix B—Adaptable Period Information on page 59* for detailed descriptions of this process.
- **Journal references**—Each journal key can have up to three reference fields associated with the transaction. iData creates a column in this view for each uniquely named journal reference that has an associated value. Unused references will not appear in iData. See *Appendix C—Adaptable Journal References on page 61* for a detailed description of this process.

- Data items—Data items can be attached to account numbers. iData creates a column for each data item defined that has a value attached. Data items defined, but not used, will not appear in iData until a value has been assigned. See *Appendix D—Adaptable Data Items on page 63* for a detailed description of this process.

## VWGLBALANCE

This document outlines the transformation of ISAM General Ledger Balance data into the GL Balance Smart View.

The vwGLBalance Smart View can be divided into three components:

- Balance component—Fixed and based on data found in the GL Balances file.
- Account dimension component—Variable and based on information contained in the FundWare divisional setup.
- Posted units component—Appended to the view if there are transactions in the GL Units file.

vwGLBalance adheres to the following configuration rules:

- Balance amount—The balance amount and units (if units are used) are automatically placed in the appropriate column of Actual, Budget, Commitment or Budget based on the balance type.
- Account dimensions—In any view that contains an account number, a column is created for each ID Type/Dimension Name combination defined in the GL setup. Each column is preceded with the ID Type, for example, A\_Object. *Appendix A—Adaptable Account Dimensions on page 57* for a detailed description on this process.

In cases where a descriptive account is missing for a particular dimension, the description for the first account number that includes that dimension is used.



## CHAPTER 3

# CONFIGURING BLACKBAUD FUNDWARE FOR iDATA

There are numerous issues that can affect how iData interprets your Blackbaud FundWare data. To ensure a successful iData build and to get the most out of your new reporting data, please read *Chapter 2—iData Business Rules on page 5* carefully before performing the steps outlined in this chapter.



After each build you should check the Process Log found in the iData Administration utility to verify any warnings or errors detected.

### ACCOUNT DIMENSIONS

For each defined ID type, check the dimension short descriptions for the following issues:

- No dimension can have the name “Account.” This is a name reserved for iData. If your setup has such a name, please change the description (e.g., abbreviate).
  - iData will fail if you do not change the dimension name.
- Valid characters in iData for the dimension name include A through Z, a through z, 0 through 9, and #. Any other characters should be removed or replaced.
- Verify that each dimension name is unique for a given ID type.
  - iData will fail if you do not change duplicate dimension names within an ID type.

### JOURNAL REFERENCES

For each journal key, check the reference short description for the following issues:

- Valid characters in iData for the reference name include A through Z, a through z, 0 through 9, and #. Any other characters should be removed or replaced.
- Columns are built based on the reference name; therefore, change similar named references to affect the grouping of reference data.



For example, if the Accounts Payable journals and Payroll journals contain a reference named “Chk #”, and you want a column for AP check numbers and another column for Payroll check numbers, you could rename one of the reference names to “Check #”.

## DATA ITEMS

Check each data item short description for the following issue:

- Valid characters in iData for the data item name include A through Z, a through z, 0 through 9, and #. Any other characters should be removed or replaced.

## C H A P T E R 4

### CONFIGURING SQL 2000

---

You will be required to complete certain tasks in SQL 2000 to effectively work with the Blackbaud FundWare iData feature. This section provides you with instructions for:

- *Starting the SQL Server Agent,*
- *Adding SQL Server 2000 Logins,*
- *Modifying SQL Server 2000 Logins,*
- *Deleting SQL Server 2000 Logins, and*
- *Installing iData and Blackbaud FundWare to Separate Servers.*

#### **STARTING THE SQL SERVER AGENT**

Before you open iData Administration, you must ensure that the Auto restart option is selected for the SQL Server Agent.

1. Click **Start** and select **Programs > Microsoft SQL Server > Enterprise Manager** from the Windows Start Menu. The Enterprise Manager displays (Figure 4.1).

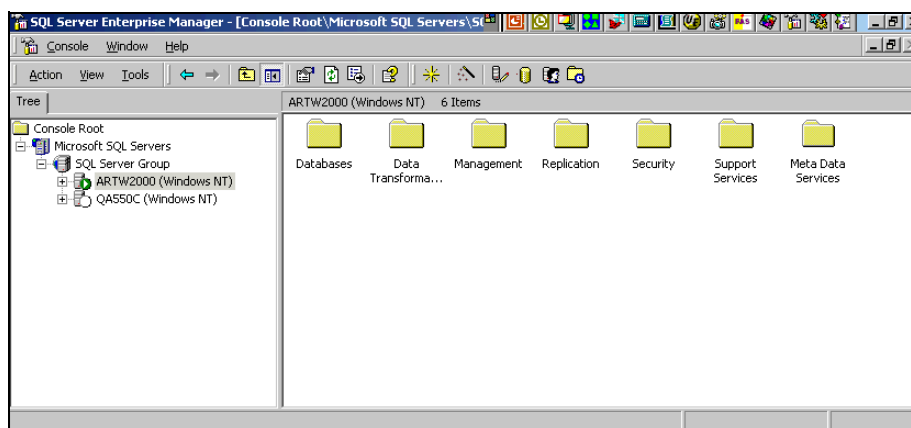
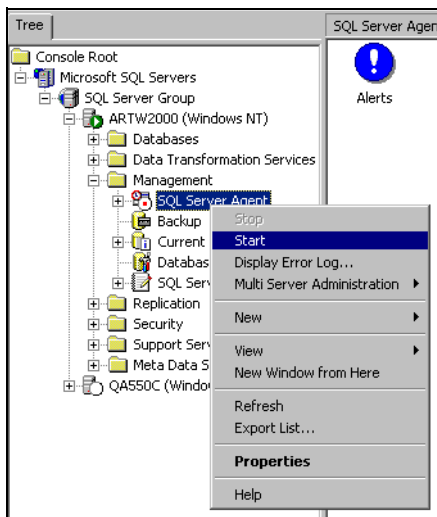


Figure 4.1—Enterprise Manager

2. Open the Management folder on your server.

3. Right-click the SQL Server Agent within this folder and select **Start** from the pop-up menu (*Figure 4.2*). (If this option is disabled, then your SQL Server Agent is already running.)



*Figure 4.2—Starting the SQL Server Agent*

4. If necessary, re-select the SQL Server Agent from the tree.

5. Right-click and select **Properties** from the pop-up menu. The SQL Server Agent Properties dialog box displays, open to the **General** tab (Figure 4.3).

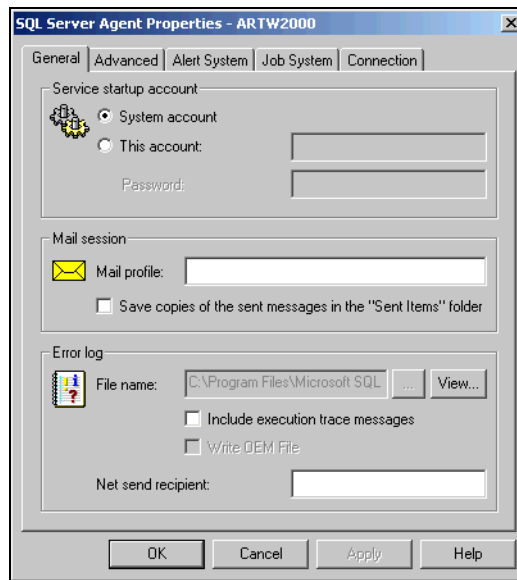


Figure 4.3—SQL Server Agent Properties, General Tab

6. Select the **Advanced** tab (*Figure 4.4*).

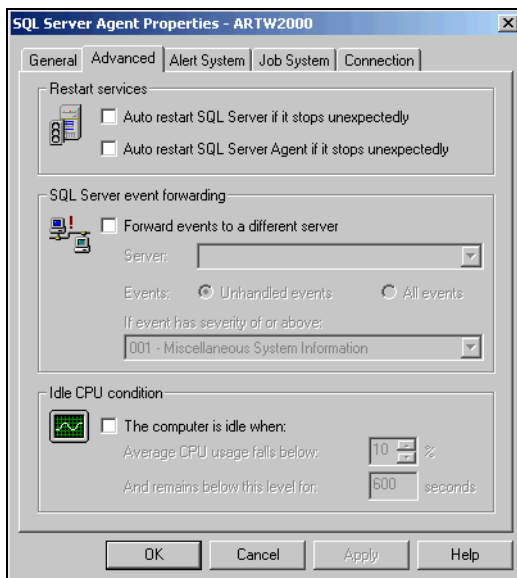


Figure 4.4—SQL Server Agent Properties, Advanced Tab

7. In the Restart services section, select the Auto restart SQL Server Agent if it Stops Unexpectedly check box (*Figure 4.5*). (If there is already a check mark in the check box, ignore this step.)

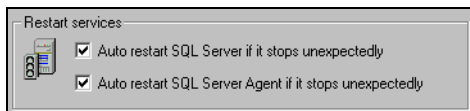


Figure 4.5—Selecting the Check Box

8. Click **OK**. Your changes are saved and the SQL Server Agent Properties dialog box closes. The agent will stop and restart to register these changes.

## ADDING SQL SERVER 2000 LOGINS

Use the following steps to add new users in SQL Server 2000.

1. Click **Start** and select **Programs > Microsoft SQL Server > Enterprise Manager** from the Windows Start Menu. The Enterprise Manager displays (Figure 4.6).

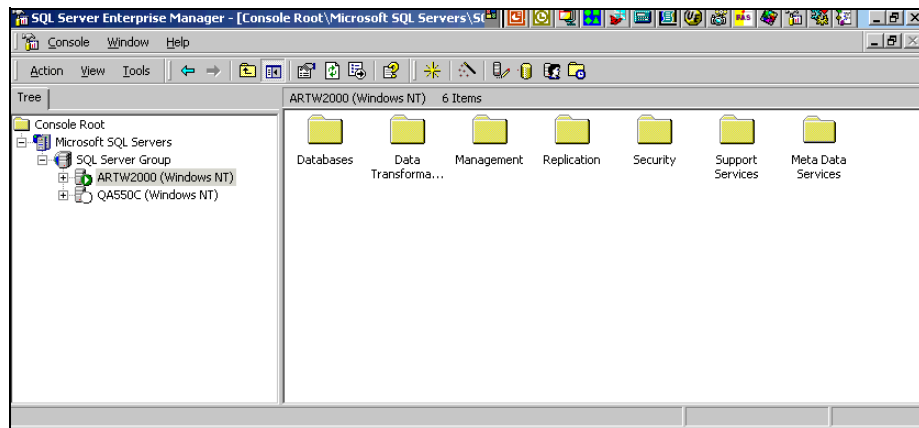


Figure 4.6—Enterprise Manager

2. Open the **Security** folder in your server directory from the tree (you may need to expand the folders).



- Right-click the **Logins** folder and select **New Login** from the pop-up menu. The SQL Server Login Properties - New Login dialog box displays (*Figure 4.7*).

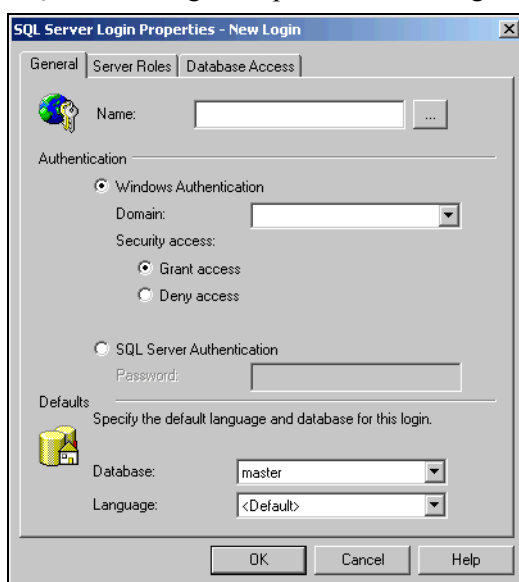


Figure 4.7—SQL Server Login Properties - New Login

- In the Authentication section, select the type of authentication you want.



While you can select either Windows Authentication or SQL Server Authentication, we recommend that you select SQL Server Authentication.

- Enter the required authentication information:
  - Windows Authentication
    - Enter the user's Windows login name in the Name text box, or click the ellipsis to view a list of valid names.
    - Select the appropriate domain name from the Domain drop-down list.
    - Accept the default entries for Security access (Grant access), Database, and Language.
  - SQL Server Authentication
    - Enter the user's SQL Server login name in the Name text box.
    - Enter the user's password in the Password text box.

- We recommend that the user have the same login and password for both FundWare and iData.
  - Accept the default entries for Database and Language.
6. When complete, click **OK**.

## MODIFYING SQL SERVER 2000 LOGINS

Use the following steps to modify the SQL Server 2000 login settings for your users.

1. Click **Start** and select **Programs > Microsoft SQL Server > Enterprise Manager** from the Windows Start Menu. The Enterprise Manager displays.
2. Open the **Security > Logins** folder in your server directory from the tree (you may need to expand the folders).
3. Select the user, right-click and select **Properties** from the pop-up menu. The SQL Server Login Properties dialog box for the selected user displays (*Figure 4.8*).

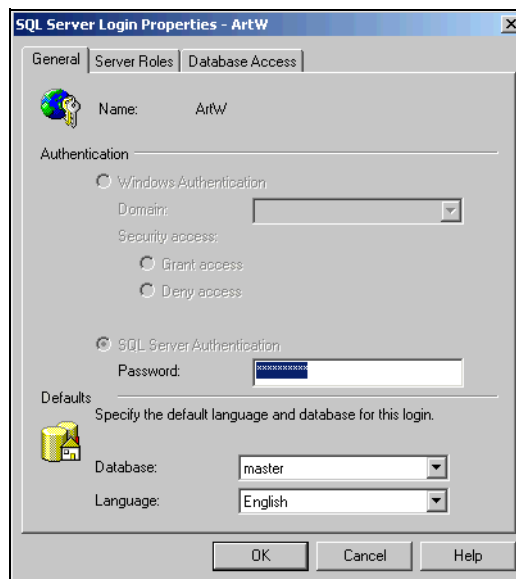


Figure 4.8—SQL Server Login Properties - New Login

4. Modify the user as necessary, according to the following:

- You can change the user's password (on the **General** tab).
  - You can change the Permit settings to grant or remove access to an iData database (**Database Access** tab).
5. When complete, click **OK**.

## DELETING SQL SERVER 2000 LOGINS

Use the following steps to delete the SQL Server 2000 login for a user.

1. Click **Start** and select **Programs > Microsoft SQL Server > Enterprise Manager** from the Windows Start Menu. The Enterprise Manager displays.
2. Open the **Security > Logins** folder in your server directory from the tree (you may need to expand the folders).
3. Select the user, right-click and select **Delete** from the pop-up menu. A confirmation message displays.
4. Click **Yes**. The user is deleted.

## INSTALLING IDATA AND BLACKBAUD FUNDWARE TO SEPARATE SERVERS

You can install iData and FundWare to separate servers. However, you must also install SQL Server 2000 to both servers, and this is best done with the support of the FundWare Professional Services department.



## C H A P T E R 5

# CONFIGURING AND ADMINISTERING iDATA

iData is installed when you install FundWare, and once you have installed the program, you can complete the tasks of the iData Administration tool that allow you to specify necessary directory paths, build processes, and user security rights. To access the options, you must complete the tasks in a specific order:

- Open the iData Administration tool.
- Enter the servers and directory paths for FundWare and iData. It is not required that these be on the same server.
- Select and configure each division that you want iData to build.
- Enter the administrator security and licensing information.
- Schedule the iData build.
- Enter security settings for the iData users. You will not be able to access the security options until you have created your first iData database, which is done through the scheduling task.

### OPENING THE iDATA ADMINISTRATION TOOL

Use the following steps to open the Administration tool.

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu. iData asks for your login information.

2. Enter your User Name and Password. The first time you log on, type *Demo User* as the User Name and *pcfund* as the Password (Figure 5.1).

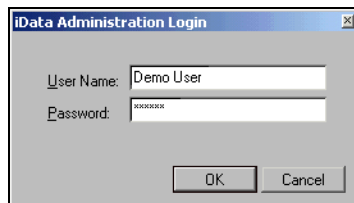


Figure 5.1—iData Login



Changing the user name and password are discussed in *Entering the General Setup Information* on page 32. Once you change the user name and password, the default is no longer active.

3. Click **OK**. The iData Administration feature opens to the Data tasks (Figure 5.2).

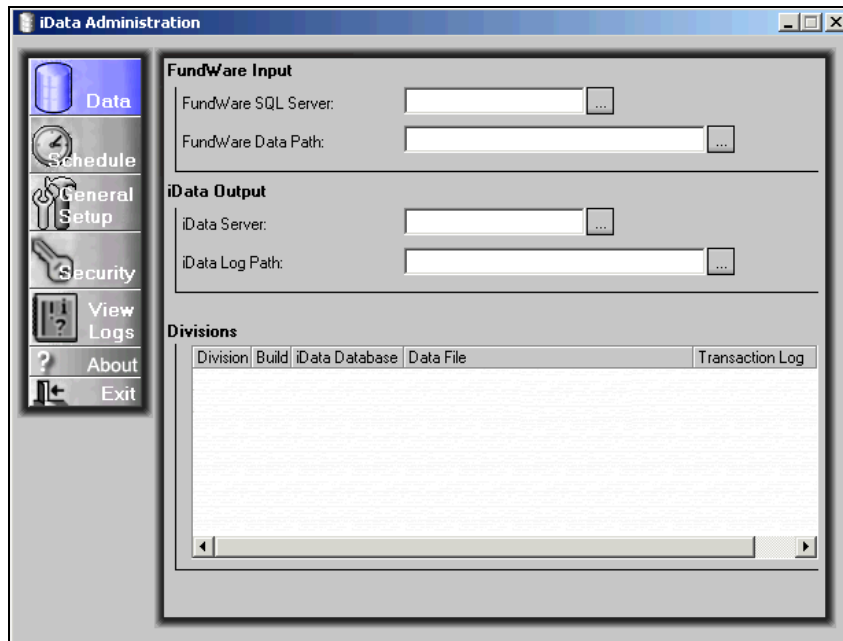


Figure 5.2—Data Administration Tasks

## ENTERING BLACKBAUD FUNDWARE AND IDATA INFORMATION

Because FundWare, SQL Server 2000, and iData can be saved on the same server or on separate servers, you must specify the server and certain directory paths for these programs. When complete, these text boxes will provide the information necessary for the iData program to find the FundWare data.

Use the following steps to enter your server and directory path information for FundWare and iData.

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu and enter your login information. The iData Administration feature displays (*Figure 5.3*).

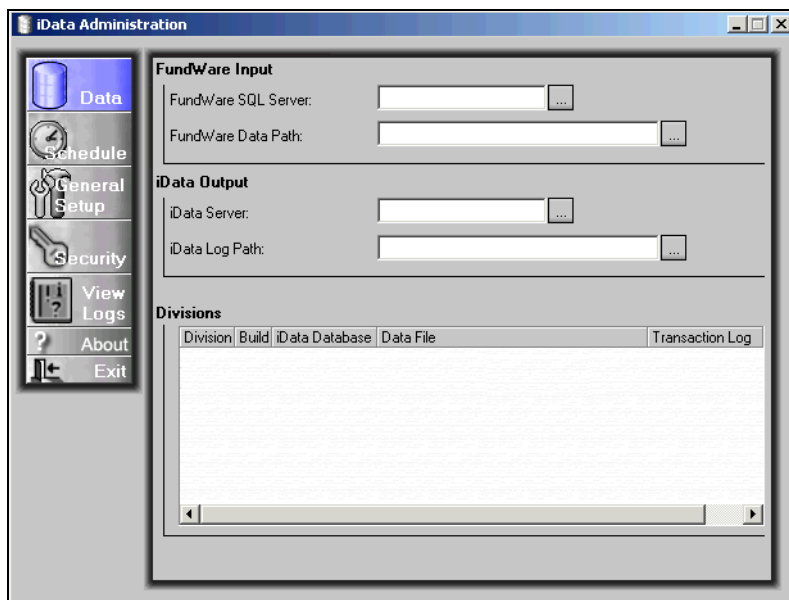


Figure 5.3—iData Administration, Data Tasks

2. Click the ellipsis in the FundWare SQL Server text box to browse to the location of your FundWare Server (*Figure 5.4*).

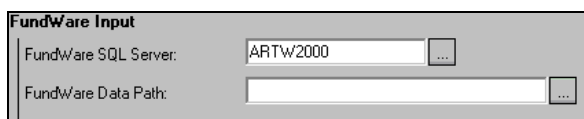
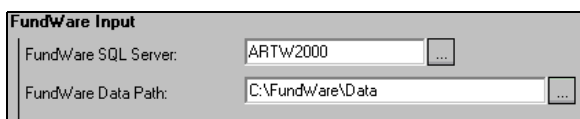


Figure 5.4—Entering the FundWare SQL Server Path (Example)

3. Click the ellipsis in the FundWare SQL Data Path text box to browse to the location of your FundWare data directory (*Figure 5.5*).



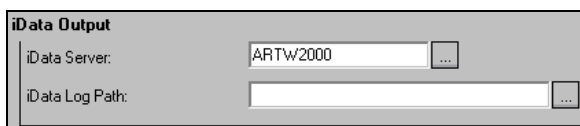
**FundWare Input**

FundWare SQL Server: ARTW2000 ...

FundWare Data Path: C:\FundWare\Data ...

*Figure 5.5—Entering the FundWare Data Path (Example)*

4. Click the ellipsis in the iData Server text box to browse to the location of your iData server (*Figure 5.6*).



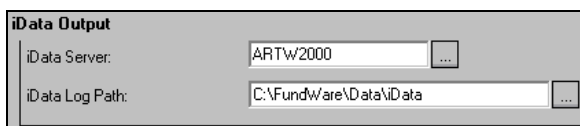
**iData Output**

iData Server: ARTW2000 ...

iData Log Path: ...

*Figure 5.6—Entering the iData Server Path (Example)*

5. Enter the directory path in the iData Log Path text box, or click the ellipsis to browse (*Figure 5.7*).



**iData Output**

iData Server: ARTW2000 ...

iData Log Path: C:\FundWare\Data\iData ...

*Figure 5.7—Entering the iData Log Path (Example)*

The divisions in the data directory are listed in the Divisions grid.



6. Select the check box for each division to be included in the iData build (Figure 5.8).

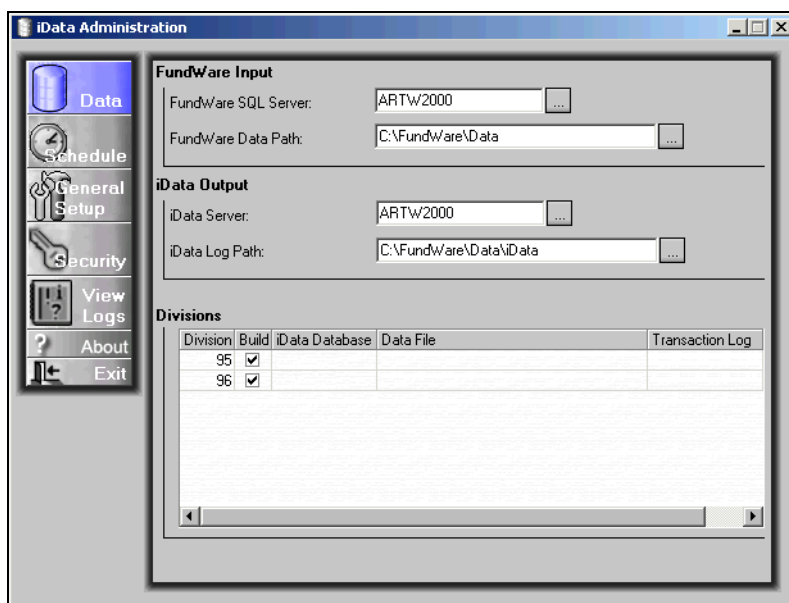


Figure 5.8—Selecting the Divisions (Example)

7. Enter the iData division name (e.g., for Division 95, we have used iData95) according to the following rules:
- The name cannot include blank spaces.
  - You cannot use the names “Div##” where the number symbols represent the division name (e.g., for Division 95, we could not use “Div95”) and “core”, as these names are used by other parts of the program.
8. Click or tab to the Data File grid box. The ellipsis button displays.
9. Click the ellipsis. The Save As dialog box displays. A default iData database file name displays in the File Name text box (e.g., for Division 95, which we named iData95, the database file name would default to iData95.mdf).
10. Browse to the directory where you want iData to store the database file.



As these files can become very large, this allows you to store the iData division file in a noncritical directory. Ensure that the directory’s available space meets the iData requirements.

11. Click **Save**.
12. Click or tab to the Transaction Log grid box. The ellipsis button displays.
13. Click the ellipsis. The Save As dialog box displays, defaulted to the same directory that you selected for the database file. A default transaction log file name displays in the File Name text box (e.g., for Division 95, which we named iData95, the transaction log file name would default to iData95.ldf).
14. Browse to the directory where you want iData to store the transaction log file (Figure 5.9).

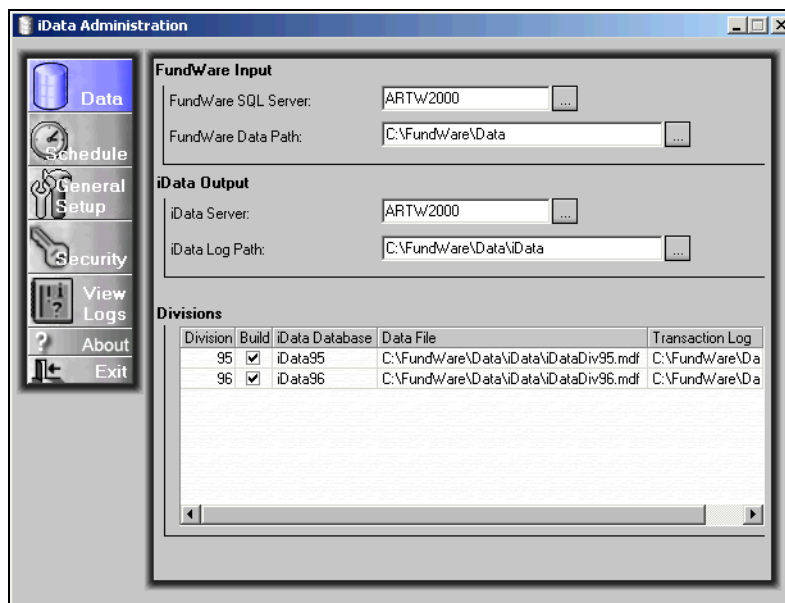


Figure 5.9—Completed Data Information (Example)

15. Click **Save**. A confirmation message displays.
16. Click **OK**. iData creates the preliminary database files. When complete, a second confirmation message displays.
17. Click **OK**.
18. Repeat step 7 through step 17 for each division.

19. When complete, you can close iData Administration by clicking **Exit** or proceed with other administration tasks.

## SCHEDULING A BUILD

You must schedule when your system will build the iData databases (i.e., when iData will pull the information from the FundWare data files and configure this information in the iData format). This process could take a number of hours and no one can be logged in to FundWare while the information is being copied. Therefore, you will probably want to schedule the builds to take place in the middle of the night or on weekends (after your system has completed a tape backup).

The frequency of the builds should be determined by your need for the “most current” data. Remember: the information in iData is only as current as the last build.

Use the following steps to schedule the iData build.

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu and enter your login information. The iData Administration window displays, open to the Data tasks.

1. Click **Schedule**. The Scheduling tasks display (*Figure 5.10*).

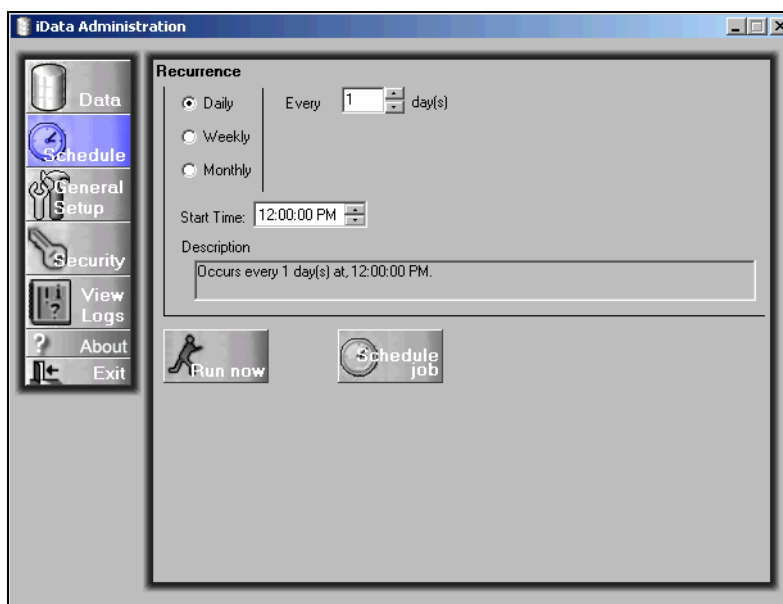


Figure 5.10—iData Administration, Schedule Tasks

2. Select the radio button for the frequency of your build (i.e., daily, weekly, or monthly).



The scheduling options to the right of these radio buttons will change depending on your selection.

3. Complete the scheduling tasks according to the following options:
  - Daily—Set the time when iData will build the data (*Figure 5.11*).

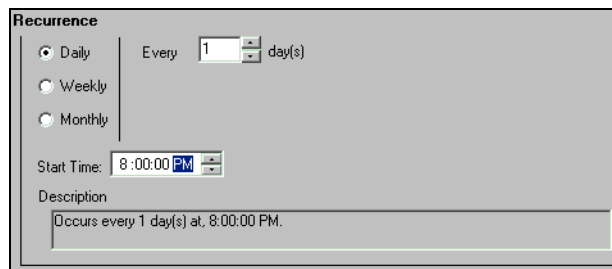


Figure 5.11—Scheduling a Daily iData Build (Example)

- **Weekly**—Set the day and time when iData will build the data (*Figure 5.12*).

**Recurrence**

Daily    Recur every  week(s) on:

**Weekly**     Sunday     Monday     Tuesday     Wednesday

Monthly     Thursday     Friday     Saturday

Start Time:

Description

Figure 5.12—Scheduling a Weekly iData Build (Example)

- **Monthly**—Set the day and time when iData will build the data (*Figure 5.13*). You can choose to build your data on a specific day of the month (e.g., the 2nd day of the month), or you can select a less specific day (e.g., last of the month could be the 28th, 30th, or 31st).
  - You can also build iData less frequently than every month (i.e., “of every \_\_ months”).

**Recurrence**

Daily     Day  of every  month(s)

Weekly     The  of every  month(s)

**Monthly**

Start Time:

Description

Figure 5.13—Scheduling a Monthly iData Build (Example)

4. When complete, click **Schedule Job**. The job is scheduled with your Enterprise Manager.

## OVERRIDING A SCHEDULED BUILD

You can cause iData to immediately build your data by clicking **Run Now**.

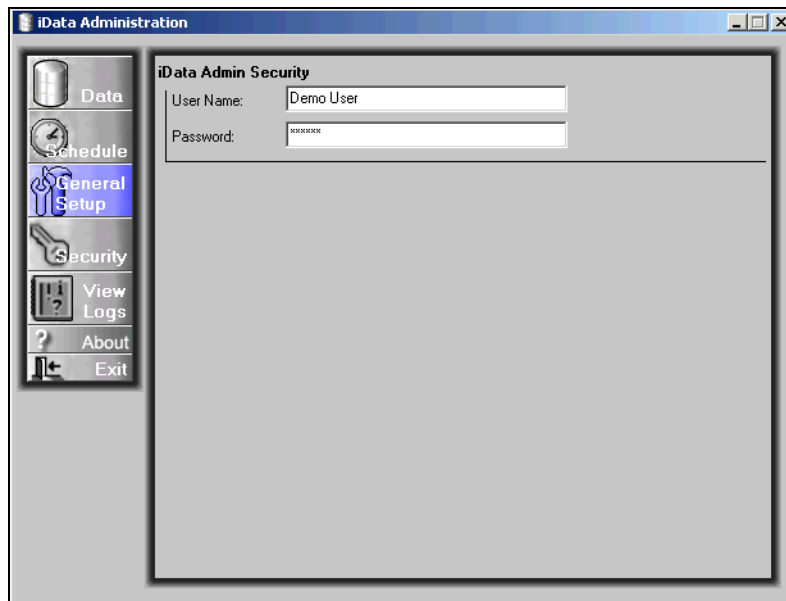


Remember: This process could take time. If you initiate an iData build, your systems could be tied up for a while.

## ENTERING THE GENERAL SETUP INFORMATION

You should enter a user name and password for the iData Administrator. When you change this entry, you will no longer be able to enter the default (User Name: *Demo User* and Password: *pcfund*).

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu and enter your login information. The iData Administration window displays, open to the Data tasks.
2. Click **General Setup**. The General Setup tasks display (*Figure 5.14*).



*Figure 5.14—iData Administration, General Setup Tasks*

3. Enter the new User Name and press **Enter**.
4. Type the new Password and press **Enter**. A confirmation dialog box pops up asking you to confirm your password.
5. Re-type your password and click **OK**.
6. When complete, you can close iData Administration by clicking **Exit** or proceed with other administration tasks.

## idata SECURITY

The security for iData is based on your user setup for SQL Server 2000. Therefore, before you can set the iData security, you must ensure that all your users have a user name in SQL Server 2000. Once you have confirmed this, you can assign security rights through iData Administration. (For more information, see *Adding SQL Server 2000 Logins* on page 17.)

In iData, there are three hierarchical levels of security (also called “Roles”):

- iData Manager—A user assigned to the iData Manager security level has access to all Smart Views in the iData database.
- Module—A user assigned to a module-level security (e.g., GL Manager for General Ledger) has access to all Smart Views for that module.
- Smart View—A user assigned to a Smart View (e.g., GL Account View) has access only to the specified Smart View.

## ASSIGNING idata SECURITY SETTINGS

Use the following steps to assign security rights for your users.



If the user’s SQL Server login is part of the Local Admin Group, then the user will be able to see all of the tables and views if the SysAdmin Role includes Builtin\Administrators. This is regardless of the setting you give the user in iData.

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu and enter your login information. The iData Administration window displays, open to the Data tasks.

2. Click **Security**. The Security tasks display, showing your SQL Server 2000 users in the SQL Logins pane and the existing divisions and Smart Views in the Roles pane (Figure 5.15).

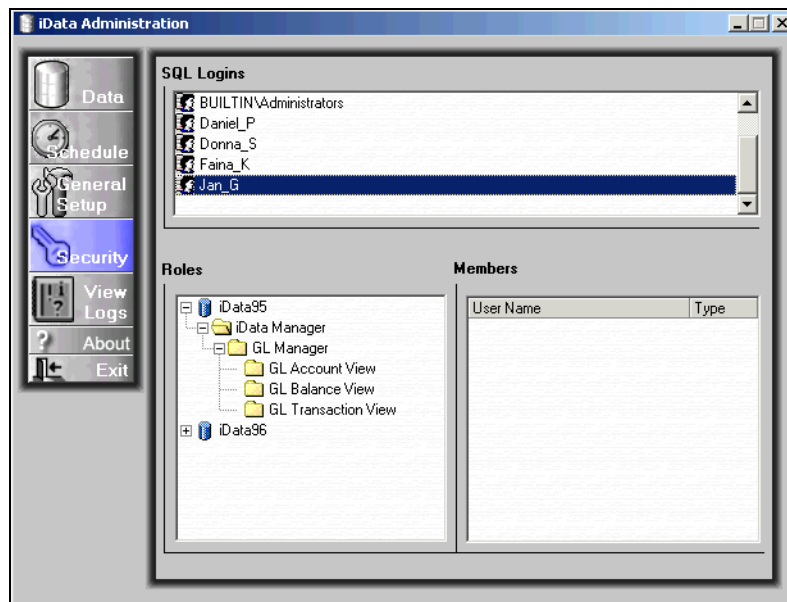


Figure 5.15—iData Administration, Security Tasks

3. Select the security level (i.e., role) to which you want to add a user (or users).
4. Select the user's login name in the SQL Logins pane.



You can use the SHIFT or CTRL keys to select more than one user, or you can select all the users in the SQL Logins pane by right-clicking and **Select All** from the pop-up menu. Using this method you can also select all, then deselect one or more users.



- Right-click and select **Add Login(s) to role** from the pop-up menu. The user (or users) is added to the Members pane (*Figure 5.16*).

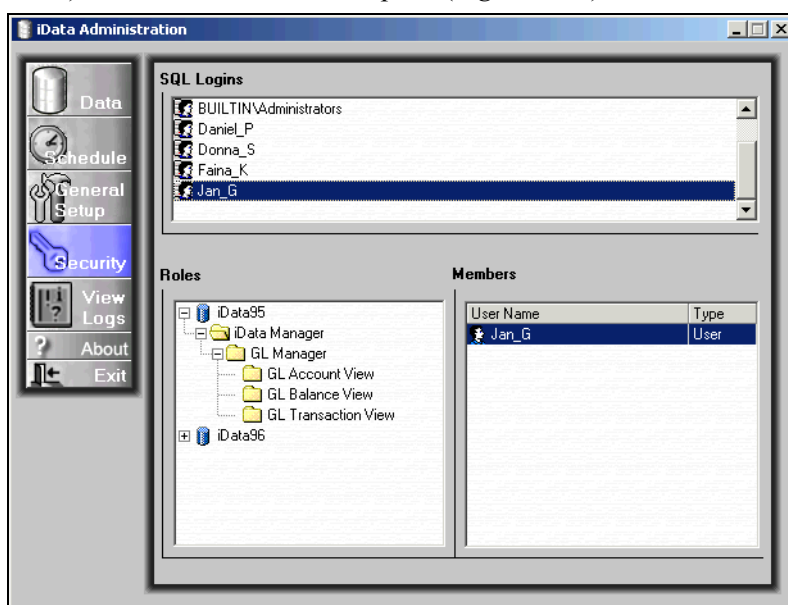


Figure 5.16—User Added to a Role



You can reverse the order of step 3 and step 4; however, the cursor must be in the SQL Logins pane to access the correct pop-up menu.



Remember: Users added to a higher role are allowed access to all roles below. For example, a user added to GL Manager has access to the GL Account View, GL Balance View, and the GL Transaction View.

- To add all users to a role, right-click and select **Select All** from the pop-up menu.
- When you have added all your users to the correct roles, you can close iData Administration by clicking **Exit** or proceed with other administration tasks.

## COPYING USERS TO SECURITY

You can use a copy command to add one or more users to a security role.

- Select the security level (i.e., role) to which you want to add a user (or users).

2. Select the user with the security settings to be copied.



You can use the SHIFT or CTRL keys to select more than one user, or you can select all the users in the SQL Logins pane by right-clicking and **Select All** from the pop-up menu. Using this method you can also select all, then deselect one or more users.

3. Right-click and select **Copy** from the pop-up menu.
4. Move the cursor to the Members pane.
5. Right-click and select **Paste** from the pop-up menu. The selected users are added to the selected role.

## COPYING SECURITY SETTINGS FROM ANOTHER DIVISION

You can copy the security settings for an entire role (regardless of level) from one division to another. When you copy this information, iData applies the security settings to the selected role. When you apply this option, iData adds the new security settings to any existing settings (i.e., nothing is overwritten).



iData will copy the security settings for all Smart Views included in the level you have selected. For example, if you select the module level GL Manager for Division 95, which includes three Smart Views, and choose to copy from Division 96, the security settings for all three Division 96 Smart Views will be applied to the security settings for the equivalent Smart Views in Division 95.

Use the following steps to copy the security settings from one division to another.

1. Select the role to which you want to apply new security settings.

- Right-click and select **Copy Security From** from the pop-up menu. A second pop-up menu displays the available divisions. For example, in *Figure 5.17* we are copying the GL Balance View security settings for Division 96 to the GL Balance View in Division 95.

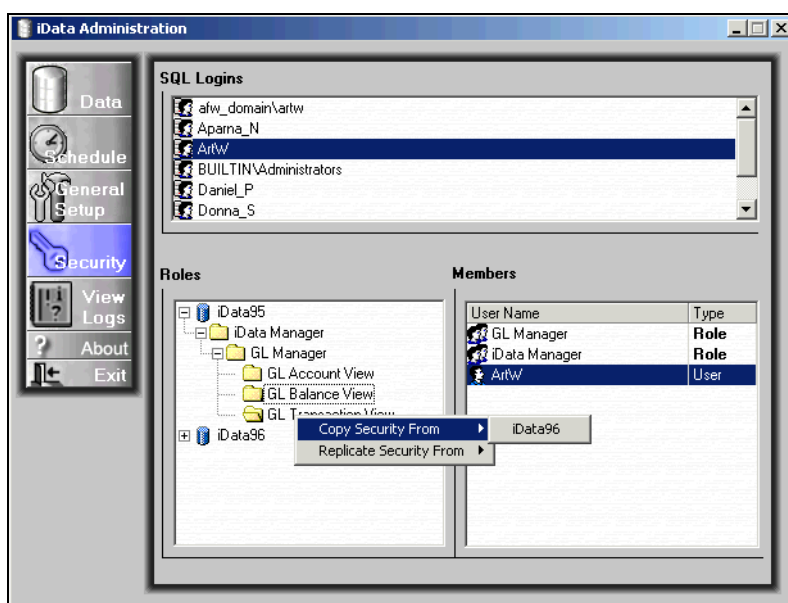


Figure 5.17—Example of Copying From Another Division

## REPLICATING SECURITY SETTINGS FROM ANOTHER DIVISION

You can replicate the security settings for an entire role (regardless of level) from one division to another. When you replicate this information, iData applies the security settings to the selected role. When you apply this option, iData overwrites any existing security settings with the new security settings.

**e** iData will replicate the security settings for all Smart Views included in the level you have selected. For example, if you select the module level GL Manager for Division 95, which includes three Smart Views, and choose to replicate from Division 96, the security settings for all three Division 96 Smart Views will overwrite the security settings for the equivalent Smart Views in Division 95.

Use the following steps to replicate the security settings from one division to another.

- Select the role to which you want to apply new security settings.

2. Right-click and select **Replicate Security From** from the pop-up menu. A second pop-up menu displays the available divisions. For example, in *Figure 5.18* we are copying the GL Account View security settings for Division 96 to the GL Account View in Division 95.

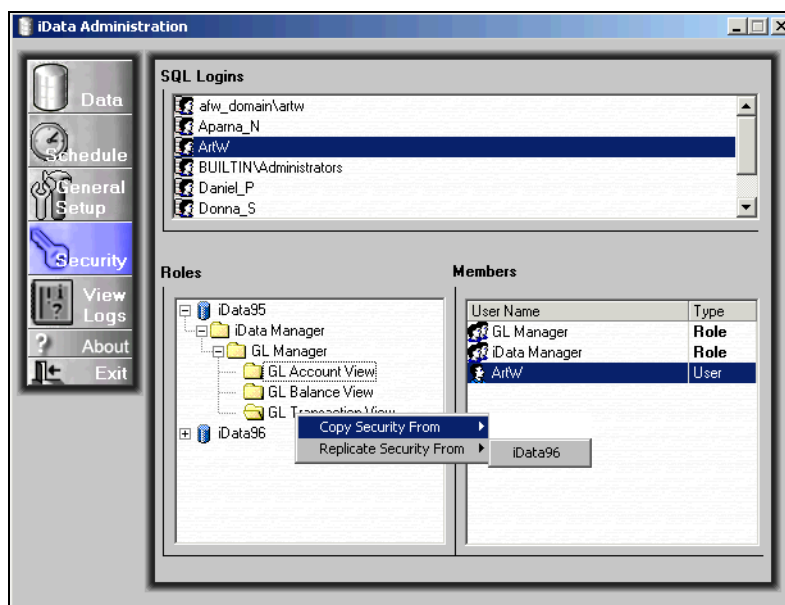


Figure 5.18—Example of Replicating From Another Division

## REMOVING SECURITY SETTINGS

You can remove the security settings from a role, or remove a user's setting.

1. Select the role in the Roles pane. The current members display in the Members pane.
2. Select the user to remove from the role.



You can use the SHIFT or CTRL keys to select more than one user, or you can select all the users in the Members pane by right-clicking and **Select All** from the pop-up menu. Using this method you can also select all, then deselect one or more users.

3. Right-click and select Remove Login(s) from role. The user is removed.

## VIEWING THE TRANSACTION LOGS

Each time iData builds a database, specifics about that build are saved to a transaction log. This log can display:

- Any warnings or errors encountered during the build, or
- The steps of the iData build process.

Use the following steps to view either log.

1. Click **Start** and select **Programs > Blackbaud FundWare > iData Admin** from the Windows Start Menu and enter your login information. The iData Administration window displays, open to the Data tasks.
2. Click **View Logs**. The existing transaction logs display (*Figure 5.19*).

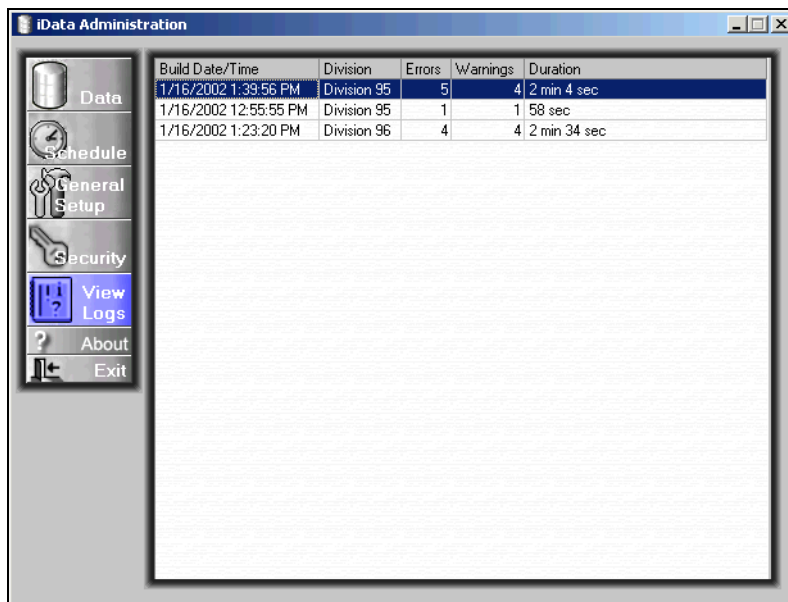


Figure 5.19—iData Administration, View Logs Options

3. Double-click the transaction log that you want to view. The log displays, defaulting to the error log view (*Figure 5.20*).

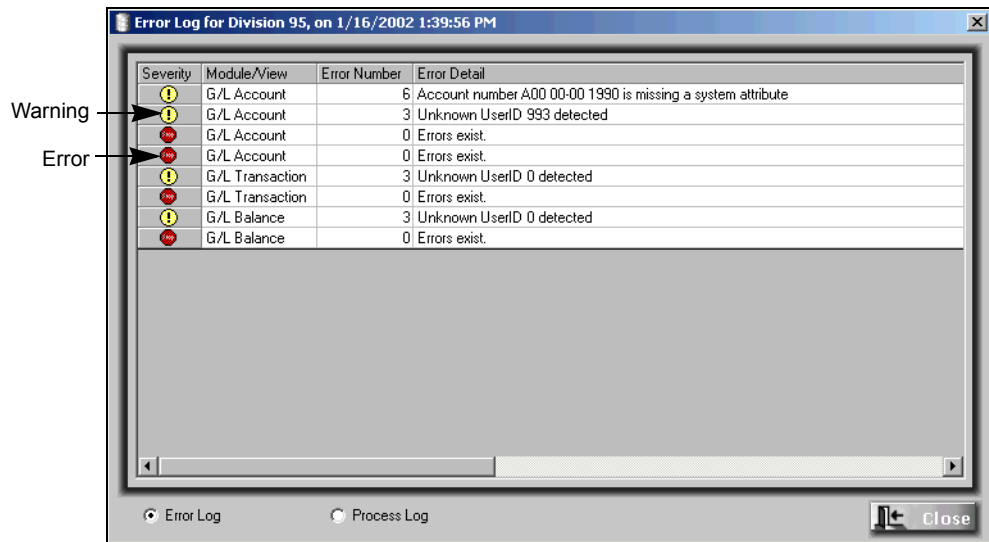


Figure 5.20—Example Transaction Error Log



For more information on these messages, see *Appendix F—Warning and Error Messages* on page 67.

4. Select the Process Log radio button to view the transaction process log (Figure 5.21).

Type	Module/View	Process	Start Time	End Time
Extract	General Ledger	Accounts	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Account Attributes	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Account Items	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Account Types	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Amount Units	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Attributes	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Dimensions	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Item Definitions	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Journals	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Period Dates	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Transaction Accounts	1/16/2002 1:39:57 PM	1/16/2002 1:39:57 PM
Extract	General Ledger	Transaction Details	1/16/2002 1:39:57 PM	1/16/2002 1:39:58 PM
Extract	General Ledger	Transaction Items	1/16/2002 1:39:58 PM	1/16/2002 1:39:58 PM
Load	General Ledger	Truncate tables	1/16/2002 1:40:02 PM	1/16/2002 1:40:02 PM
Load	General Ledger	Truncate tables	1/16/2002 1:40:00 PM	1/16/2002 1:40:02 PM
Load	General Ledger	Truncate tables	1/16/2002 1:40:00 PM	1/16/2002 1:40:02 PM
Load	General Ledger	CSB	1/16/2002 1:40:00 PM	1/16/2002 1:40:20 PM
Load	General Ledger	CSB	1/16/2002 1:40:00 PM	1/16/2002 1:40:21 PM
Load	General Ledger	CSD	1/16/2002 1:40:02 PM	1/16/2002 1:40:19 PM

Figure 5.21—Transaction Process Log

5. If you wish, select the Error Log radio button to return to the error log.
6. Click Close to close the log and return to the View Logs task.

## CLOSING THE iDATA ADMINISTRATION

To close the iData Administration tool, click **Exit**. Any changes to the iData settings will be saved at this time.





## CHAPTER 6

### DATA DICTIONARIES

iData databases are uniquely configured by user-defined divisional setups. As a result, there are two types of dictionaries available in iData: static and dynamic. The static dictionary defines the fixed and variable components of a Smart View. The variable components are described in a general format that allows the user to predict how a particular Smart View will appear according to their setup, and includes a description field for each column.

The dynamic dictionary is automatically generated by iData and contains the actual field names and data type/sizes for each Smart View. The dynamic dictionary does not contain field descriptions. The dynamic dictionaries are contained in a Smart View called “iDataDictionary” found in each iData divisional database.

This chapter contains the static dictionaries for each available Smart View. *Table 6.1* defines the notations that the static dictionaries use to describe the column data type.

*Table 6.1—Static Dictionary Notations*

Data Type	Data Type Description
Numeric	A positive numeric value (cannot be negative).
Numeric +/-	A positive or negative numeric value.
Date/Time	A field containing both date and time.
Date/Time +	A field containing date information, and where the time component is defaulted by iData to 12:00 a.m. because the source field did not contain a time component. SQL Server 2000 does not have a date field, only a date/time field.
Text	A field containing both text and numeric information.
DI as Defined	One of the field types above as defined by the data item type selected in FundWare.

## GL ACCOUNTS

The GL Account Smart View can be divided into four components as follows:

- Account component
- Account dimension component
- Attribute component
- Data item component

The account component is fixed based on data found in the GL Account file, as shown in *Table 6.2*.

*Table 6.2—Fixed Account Component*

View Order	Column Name	Data Type	Length	Description
01	Data As Of	Date/Time		The start date and time of the iData build.
02	ID Type	Text	1	An ID type is a single-letter code that identifies a specific group of accounts for reporting purposes. The ID type controls the end of the fiscal year, the type of accounting, fiscal period, names for the default journals, and account number structure.
03	Account Number	Text	30	The number that stores information about transactions to an account.
04	Account Description	Text	30	A textual description of the account.
05	Account Stat	Text	1	The account status, as described in the field Account Stat Desc.
06	Account Stat Desc	Text	15	Valid descriptions are Active, Budget, Descriptive, System or Inactive.
07	Account Alias 1	Text	30	Alternate account number, name, or any additional reference that assists in finding and/or reporting accounts. This is optional.
08	Account Alias 2	Text	30	Alternate account number, name, or any additional reference that assists in finding and/or reporting accounts. This is optional.
09	Modify User_GLAccount	Text	40	The name of the last person who modified the Account record.

Table 6.2—Fixed Account Component (Continued)

View Order	Column Name	Data Type	Length	Description
10	Modify Date_GLAccount	Date/Time		The last date the Account record was modified.

The account dimension component is variable based on information contained in the FundWare divisional setup, as shown in *Table 6.3*.

Table 6.3—Account Dimension Component

View Order	Column Name	Data Type	Length	Description
11A	"IDType"_"IDType Dimension Name"	Text	9	Dimensions are the subsets of an account number used to determine how the account information can be reported and processed.
11B	"IDType"_"IDType Dimension Name" Desc	Text	30	Dimension description.

The attribute component is based on attributes in the FundWare setup, as shown in *Table 6.4*.

Table 6.4—Attribute Component

View Order	Column Name	Data Type	Length	Description
12A	"Attribute Column Name"	Text	4	An attribute is a code that is one to three characters in length and allows accounts to be grouped on reports according to their type. You can have up to 15 attributes per account.
12B	"Attribute Column Name" Desc	Text	60	Attribute description.
12C	"Attribute Column Name" Stat	Text	10	Attribute status: Active or Inactive.

The data item component is variable and built only if the Client has attached data items to GL Accounts, as shown in *Table 6.5*.

*Table 6.5—Data Item Component*

View Order	Column Name	Data Type	Length	Description
13A	"Data Item Name"_DI_GLAacct	DI As Defined		Data items are pieces of information that can be included in account definition or in a transaction. Typically they are used to store non-financial information. Transactional data items are defined in Journal Maintenance.

## GL TRANSACTIONS

The GL Transaction Smart View can be divided into five components as follows:

- Transaction component
- Account component
- Period component
- Transaction reference component
- Data item component

The Transaction Component is fixed based on information contained in the Transaction Detail file, as shown in *Table 6.6*.

*Table 6.6—Transaction Component*

View Order	Column Name	Data Type	Length	Description
01	Data As Of	Date/Time		The start date and time of the iData build.
02	Post Date	Date/Time		The date used to post a transaction to a ledger. Also called the business date.
03	Tran Type	Text	1	An ID used to identify the type of transaction, as defined in the Tran Type LG Desc. Valid values are C, I, L, M, N, O, W, and X.

Table 6.6—Transaction Component (Continued)

View Order	Column Name	Data Type	Length	Description
04	Tran Type SH Desc	Text	10	A short transaction type description. Valid values are Beg Bal, Detail, 1st Adj, 2nd Adj, 3rd Adj, 4th Adj, Reclass, and End Bal.
05	Tran Type LG Desc	Text	30	The transaction type full description. Valid values are Beginning Balance, Detail, First Adjustment, Second Adjustment, Third Adjustment, Fourth Adjustment, Reclassification, and Ending Balance.
06	Batch Nbr GLTran	Numeric +/-	10	A unique number that identifies a group of items (account entries) entered at the same time. The group of items are sometimes referred to as transactions.
07	Batch Item Nbr GLTran	Numeric +/-	10	A unique number within a batch that identifies an item (account entry), or sometimes referred to as a transaction.
08	Modify User GLTran	Text	40	The name of the last person who modified the transaction record prior to posting
09	Modify Date GLTran	Date/Time		The last date the record was modified prior to posting.
10	Post Sys SH Desc	Text	3	A short description of the originating system the transaction was posted from.
11	Post Sys LG Desc	Text	30	A long description of the originating system the transaction was posted from.
12	Jrnl Abbr	Text	5	An abbreviation of the “book of original entry” that records transactions to be posted to a ledger.
13	Jrnl Description	Text	30	The journal description.
14	Jrnl Post Type	Text	1	There are four valid journal post types: 0, 1, 2, and 3 as described by Jrnl Post Type LG Desc.
15	Jrnl Post Type SH Desc	Text	4	A short description of the journal post type. Valid values are Act, Bud, Enc, and Comm.
16	Jrnl Post Type LG Desc	Text	20	A full description of the journal post type. Valid values are Actual, Budget, Encumbrance, and Commitment.

Table 6.6—Transaction Component (Continued)

View Order	Column Name	Data Type	Length	Description
17	Transaction Description	Text	30	A description of the transaction record.
18	Actual Post Units_GLTran	Numeric +/-	13.4	Actual posted units.
19	Actual Post Rate_GLTran	Numeric +/-	13.4	Actual posted rate.
20	Actual Post Amount_GLTran	Numeric +/-	15.2	Actual posted amount.
21	Budget Post Units_GLTran	Numeric +/-	13.4	Budget posted units.
22	Budget Post Rate_GLTran	Numeric +/-	13.4	Budget posted rate.
23	Budget Post Amount_GLTran	Numeric +/-	15.2	Budget posted amount.
24	Encum Post Units_GLTran	Numeric +/-	13.4	Encumbrance posted units.
25	Encum Post Rate_GLTran	Numeric +/-	13.4	Encumbrance posted rate.
26	Encum Post Amount_GLTran	Numeric +/-	15.2	Encumbrance posted amount.
27	Comm Post Units_GLTran	Numeric +/-	13.4	Commitment posted units.
28	Comm Post Rate_GLTran	Numeric +/-	13.4	Commitment posted rate.
29	Comm Post Amount_GLTran	Numeric +/-	15.2	Commitment posted amount.
30	Cal Qtr	Text	6	Calendar quarter based on Post Date.

The account component is based on the ID type dimension definitions defined in the FundWare setup. Since a transaction can be posted to multiple accounts, columns are created for each possible account component (*Table 6.7*).

*Table 6.7— Account Component*

View Order	Column Name	Data Type	Length	Description
31A	"ID Type" Sys Attr	Text	4	An attribute is a code that is one to three characters in length and allows accounts to be grouped on reports according to their type. Attribute 1 is sometimes referred to as the System attribute because they are predefined by FundWare. Examples of a system attribute are EX (expense), LI (liability), and AS (Asset).
31B	"ID Type" Sys Attr Description	Text	60	The description of the System attribute.
31C	"ID Type"_ID Type	Text	1	An ID type is a single-letter code that identifies a specific group of accounts for reporting purposes. The ID type controls the end of the fiscal year, the type of accounting, fiscal period, names for the default journals, and account number structure.
31D	"ID Type" Account Number	Text	30	The number that stores information about transactions to an account.
31E	"ID Type" AccountDescription	Text	30	A textual description of the account.
31F	"ID Type"_ "IDType Dimension Name"	Text	9	Dimensions are the subsets of an account number used to determine how the account information can be reported and processed.
31G	"ID Type"_ "IDType Dimension Name" Desc	Text	30	Dimension Description

The period component is based on the ID type period definitions created in the FundWare setup. Since a transaction can be posted to multiple accounts, columns are created for each possible period component possible (*Table 6.8*).

*Table 6.8—Period Component*

View Order	Column Name	Data Type	Length	Description
32A	"ID Type" FscI Per	Text	2	The General Ledger-Project/Grant module gives you the ability to set up fiscal periods for each ID Type and to determine the number of years to maintain/keep on-line. Date Tables are used to store period lengths and period-ending dates. Valid values are BG, ## (which represents a period number), A1, A2, A3, A4, RC or CL.
32B	"ID Type" FscI Per SH Desc	Text	10	A short description of the Fiscal Period. Valid values are Beg Bal, Det Per ## (where # represents the period number), Adj 1, Adj 2, Adj 3, Adj 4, Reclass or Close Bal.
32C	"ID Type" FscI Per LG Desc	Text	30	A full description of the Fiscal Period. Valid values are Beginning Balance, Detail Period ## (where # represents the period number), Adjustment 1, Adjustment 2, Adjustment 3, Adjustment 4, Reclass or Closing Balance.
32D	"ID Type" FscI Per Begin Date	Date/Time		The beginning date of the Fiscal Period.
32E	"ID Type" FscI Per End Date	Date/Time		The ending date of the Fiscal Period.
32F	"ID Type" FscI Per Year	Numeric	4	The year of the Fiscal Period.



The transaction reference component is based on unique reference names assigned to journals, as shown in *Table 6.9*.

*Table 6.9—Transaction Reference Component*

View Order	Column Name	Data Type	Length	Description
33A	"Trans_Ref_Name"_Ref	Text		A column is created for each uniquely named reference defined in journal controls that has some transactional data assigned. The contents of reference fields vary depending on the journal used. These fields contain information you need about a particular transaction.

The Data Item Component is built only if the Client has attached data items to GL transactions, as shown in *Table 6.10*.

*Table 6.10—Data Item Component*

View Order	Column Name	Data Type	Length	Description
34A	"Data Item Name"_DI_GLTran			A column is created for each uniquely named Data Item that has some data assigned. Data items are pieces of information that can be included in account definition or in a transaction. Typically they are used to store non-financial information. Transactional data items are defined in Journal Maintenance.

## GL BALANCES

The GL Balance Smart View can be divided into three components:

- Balances component
- Account dimension component
- Posted units component

The Balances Component is fixed based on data found in the GL Balances file, as shown in *Table 6.11*.

*Table 6.11—Balances Component*

View Order	Column Name	Data Type	Length	Description
01	Data As Of	Date/Time		The start date and time of the iData build.
02	Per Yr End Date	Date/Time		The Period year end date in the format mm/dd/yyyy.
3A	ID Type	Text	1	An ID type is a single-letter code that identifies a specific group of accounts for reporting purposes. The ID type controls the end of the fiscal year, the type of accounting, fiscal period, names for the default journals, and account number structure.
3B	Account Number	Text	30	The number that stores information about transactions to an account.
04	Account Description	Text	30	A textual description of the account.
05	Account Stat	Text	1	The account status, as described in the field Account Stat Desc.
06	Account Stat Desc	Text	10	Valid descriptions are Active, Budget, Descriptive, System, or Inactive
07	FscI Per	Text	2	The General Ledger-Project/Grant module gives you the ability to set up fiscal periods for each ID Type and to determine the number of years to maintain/keep on-line. Date Tables are used to store period lengths and period-ending dates. Valid values are BG, ## (which represents a period number), A1, A2, A3, A4, RC or CL.
08	FscI Per SH Desc	Text	10	A short description of the Fiscal Period. Valid values are Beg Bal, Det Per ## (where # represents the period number), Adj 1, Adj 2, Adj 3, Adj 4, Reclass or Close Bal.

Table 6.11—Balances Component (Continued)

View Order	Column Name	Data Type	Length	Description
09	Fscl Per LG Desc	Text	30	A full description of the Fiscal Period. Valid values are Beginning Balance, Detail Period ## (where # represents the period number), Adjustment 1, Adjustment 2, Adjustment 3, Adjustment 4, Reclass, or Closing Balance.
10	Fscl Per Begin Date	Date/Time		The beginning date of the Fiscal Period.
11	Fscl Per End Date	Date/Time		The ending date of the Fiscal Period
12	Fscl Per Year	Numeric	4	The year of the Fiscal Period
13	Cal Qtr	Text	6	Calendar quarter based on Post Date.
14	Sys Attr	Text	4	An attribute is a code that is one to three characters in length and allows accounts to be grouped on reports according to their type. Attribute 1 is sometimes referred to as the System attribute because they are predefined by FundWare. Examples of a system attribute are EX (expense), LI (liability), and AS (Asset).
15	Sys Attr Desc	Text	60	The description of the System attribute.
16	Sys Attr Stat	Text	10	Attribute Status: Active or Inactive
17	Actual Amount_GLBal	Numeric +/-	16.2	Summary of the actual posted amount for the accounting period.
18	Budget Amount_GLBal	Numeric +/-	16.2	Summary of the budgeted posted amount for the accounting period.
19	Comm Amount_GLBal	Numeric +/-	16.2	Summary of the commitment posted amount for the accounting period.
20	Encum Amount_GLBal	Numeric +/-	16.2	Summary of the encumbrance posted amount for the accounting period.
21	Modify User Actual Amt_GLBal	Text	40	The name of the last person who modified the actual transaction record prior to posting.
22	Modify Date Actual Amt_GLBal	Date/Time		The last date the actual record was modified prior to posting.

Table 6.11—Balances Component (Continued)

View Order	Column Name	Data Type	Length	Description
23	Modify User Budget Amt_GLBal	Text	40	The name of the last person who modified the budget transaction record prior to posting.
24	Modify Date Budget Amt_GLBal	Date/Time		The last date the budget record was modified prior to posting.
25	Modify User Comm Amt_GLBal	Text	40	The name of the last person who modified the commitment transaction record prior to posting.
26	Modify Date Comm Amt_GLBal	Date/Time		The last date the commitment record was modified prior to posting.
27	Modify User Encum Amt_GLBal	Text	40	The name of the last person who modified the encumbrance transaction record prior to posting.
28	Modify Date Encum Amt_GLBal	Date/Time		The last date the encumbrance record was modified prior to posting.

The account dimension component is variable based on information contained in your FundWare divisional setup, as shown in *Table 6.12*.

Table 6.12—Account Dimension Component

View Order	Column Name	Data Type	Length	Description
29A	"ID Type"_"IDType Dimension Name"	Text	9	Dimensions are the subsets of an account number used to determine how the account information can be reported and processed.
29B	"ID Type"_"IDType Dimension Name" Desc	Text	30	Dimension Description

The posted units component is appended to the view if there are unit transactions in the GL Balance file, as shown in *Table 6.13*.

*Table 6.13—Posted Units Component*

View Order	Column Name	Data Type	Length	Description
30A	Actual Units_GLBal	Numeric +/-	14.4	Summary of the actual posted units for the accounting period.
30B	Budget Units_GLBal	Numeric +/-	14.4	Summary of the budgeted posted units for the accounting period.
30C	Comm Units_GLBal	Numeric +/-	14.4	Summary of the commitment posted units for the accounting period.
30D	Encum Units_GLBal	Numeric +/-	14.4	Summary of the encumbrance posted units for the accounting period.
30E	Modify User Actual Units_GLBal	Text	40	The name of the last person who modified the actual unit record prior to posting.
30F	Modify Date Actual Units_GLBal	Date/Time		The last date the actual unit record was modified prior to posting.
30G	Modify User Budget Units_GLBal	Text	40	The name of the last person who modified the budget unit record prior to posting.
30H	Modify Date Budget Units_GLBal	Date/Time		The last date the budget unit record was modified prior to posting.
30I	Modify User Comm Units_GLBal	Text	40	The name of the last person who modified the commitment unit record prior to posting.
30J	Modify Date Comm Units_GLBal	Date/Time		The last date the commitment unit record was modified prior to posting.
30K	Modify User Encum Units_GLBal	Text	40	The name of the last person who modified the encumbrance unit record prior to posting.
30L	Modify Date Encum Units_GLBal	Date/Time		The last date the encumbrance unit record was modified prior to posting.



# A P P E N D I X A

## ADAPTABLE ACCOUNT DIMENSIONS

An ID type is a single-letter code that identifies a specific group of accounts for the purposes of tracking accounts and reporting. Each ID type can then have its own account number structure that is divided into dimensions defining funds, departments, subdepartments, and other cost centers. You can use account number dimensions to sort, sum, and select information for reporting.



You could create ID types F and S for separate sets of federal and state grant accounts. Your federal grant accounts may need 12 numbers divided into 5 dimensions, while your state grant accounts need 10 numbers divided into 4 dimensions.

Table A.1 contains an example definition of dimensions for ID type A.

*Table A.1—Example Definition of Dimensions  
(ID Type A)*

Dimension Name	Dimension Size
Fund	2
Dept	2
Sub-Dept	2
Object	4

Table A.2 shows how iData displays an account number and its dimensions based on the setup in Table A.1.

*Table A.2—Account Number and Dimensions Based On Setup*

A_ID Type	A_Account	A_Account_Description	A_Fund	A_Fund_Desc
A	A01 01-01 1111	Cash in Bank-Citywide	01	General
A	A01 02-04 1112	Denver Payroll Bank Account	01	General

A_Dept	A_Dept_Desc	A_Sub_Dept	A_Sub_Dept_Desc	A_Object	A_Object_Desc
01	Accounting	01	Payroll	111	Cash in Bank
02	Water	04	Road & Bridge	112	Savings Account

In each Smart View where an account number is included, the account dimensions and dimension descriptions are also included. This breakdown of the account number provides a significant benefit to the ease and flexibility of reporting. Note that all columns are preceded by the account number ID type and an underscore (“\_”) character.



Whenever reporting by a specific account dimension, be sure to include a filter on the account ID type. Otherwise, you will see information from other ID types that you probably were not expecting.

Table A.3 contains an example definition of dimensions for ID type F. Notice that this account structure is considerably different than ID type A as defined in Table A.1.

Table A.3—Example Definition of Dimensions  
(ID Type F)

Dimension Name	Dimension Size
Project	3
Grant	2
Object	4

Table A.4 shows how iData displays an account number and its dimensions based on the setup in Table A.3.

Table A.4—Account Number and Dimensions Based On Setup

F_ID Type	F_Account	F_Account_Description	F_Project	F_Project_Desc
F	F100-02 7910	Copy Expense	100	Pregnant Teens
F	F200-00 0000	Supplies	200	Homeless Shelter

F_Grant	F_Grant_Desc	F_Object	F_Object_Desc
02	Research	7910	Copying
00	Subsidizing	0000	Office Supplies



“Account” is a reserved word in iData dimension names. If you have a dimension with this name, it must be renamed prior to your next iData build or iData will fail.



# A P P E N D I X B

## ADAPTABLE PERIOD INFORMATION

The General Ledger–Project/Grant module gives you the ability to set up fiscal periods for each ID type and to determine the number of years to maintain. Date tables are used to store period lengths and period-ending dates. *Table B.1* shows a typical period table.

*Table B.1—Example Period Table*

ID Type	A_FscI Per	A_FscI Per End Date	A_FscI Per Year
A	BG	01/01/2001	2001
A	01	01/31/2001	2001
A	02	02/28/2001	2001
A	03	03/31/2001	2001
A	04	04/30/2001	2001
A	05	05/31/2001	2001
A	06	06/30/2001	2001
A	07	07/31/2001	2001
A	08	08/31/2001	2001
A	09	09/30/2001	2001
A	10	10/31/2001	2001
A	11	11/30/2001	2001
A	12	12/31/2001	2001
A	a1	12/31/2001	2001
A	a4	12/31/2001	2001
A	rc	12/31/2001	2001
A	CL	12/31/2001	2001

When a view contains a fiscal period, iData interprets ID type fiscal period information and creates columns for the:

- Fiscal period,
- Short fiscal period description,
- Long fiscal period description,
- Fiscal period begin date,

- Fiscal period end date, and
- Fiscal period year.

Each of these columns are specifically created to meet various reporting needs. *Table B.2* shows how iData interprets the information found in *Table B.1*.

*Table B.2—iData Interpretation of Table B.1*

A_Fscl Per	A_Fscl Per SH Desc	A_Fscl Per LG Desc	A_Fscl Per Begin Date	A_Fscl Per End Date	A_Fscl Per Year
BG	Beg Bal	Beginning Balance	01/01/2001	01/01/2001	2001
01	Det Per 01	Detail For Period 01	01/01/2001	01/31/2001	2001
02	Det Per 02	Detail For Period 02	02/01/2001	02/28/2001	2001
03	Det Per 03	Detail For Period 03	03/01/2001	03/31/2001	2001
04	Det Per 04	Detail For Period 04	04/01/2001	04/30/2001	2001
05	Det Per 05	Detail For Period 05	05/01/2001	05/31/2001	2001
06	Det Per 06	Detail For Period 06	06/01/2001	06/30/2001	2001
07	Det Per 07	Detail For Period 07	07/01/2001	07/31/2001	2001
08	Det Per 08	Detail For Period 08	08/01/2001	08/31/2001	2001
09	Det Per 09	Detail For Period 09	09/01/2001	09/30/2001	2001
10	Det Per 10	Detail For Period 10	10/01/2001	10/31/2001	2001
11	Det Per 11	Detail For Period 11	11/01/2001	11/30/2001	2001
12	Det Per 12	Detail For Period 12	12/01/2001	12/31/2001	2001
a1	Adj 1	Adjustment 1	12/31/2001	12/31/2001	2001
a2	Adj 2	Adjustment 2	12/31/2001	12/31/2001	2001
rc	Reclass	Reclassification	12/31/2001	12/31/2001	2001
CL	Close Bal	Closing Balance	12/31/2001	12/31/2001	2001



Note that all columns are preceded by the account number ID type and an underscore ( ) character.



iData allows for 60 periods; however, iData only creates 20 periods at a time. For example, if you post to week 1, iData will create periods 1-20. When you post to week 21 (or later), periods 21-40 will be created.

# A P P E N D I X C

## ADAPTABLE JOURNAL REFERENCES

All transactions in General Ledger–Project/Grant are posted through journals, which are defined through Journal Maintenance in FundWare. When iData creates the General Ledger transaction Smart View, it incorporates much of this journal information, including up to three journal reference fields, as seen in *Table C.1*.



iData does not build columns for journal references that have no stored value. For example, if there is a reference called “Oblig #”, but the source files have no stored value for that reference, this column will not be included in the Smart View.

*Table C.1—Journal References*

Journal Key	Description	Reference 1	Reference 2	Reference 3
GJNLA	General Journal	Source	JE #	Ref 3
BDGTA	Budget Journal	Ref 1	Ref 2	Ref 3
ENCUM	Encumbrance Journal	Oblig #	Invoice #	Check #
PURCH	Purchasing Journal	Oblig #	Invoice #	Check #
DEPR	Depreciation Journal	Asset Num	Ref 2	Ref 3
CDIS	Cash Disbursements Journal	Oblig. #	Invoice #	Check #



Reference 1 for the CDIS journal is “Oblig. #”. Because a period is an illegal character, iData will change this reference name to “Oblig\_ #”. Also, even though it appears that this column is the same as Reference 1 for the ENCUM and PURCH journals, the slight difference in name will cause iData to treat them as separate entries and create separate columns for Oblig # and Oblig\_ #. See *Table C.2*.

Notice that a particular reference, such as Check #, can appear on multiple journals and in any reference column (1,2, or 3). iData automatically creates a column for each unique reference name, and where appropriate, places information for that reference in that column. This allows you to report on similar data from multiple journals (or posting systems).

If dissimilar information is combined (e.g., you want the payroll check information in a different column from other check information), then rename the appropriate reference name. The new column displays the next time iData rebuilds the Smart Views. *Table C.2* displays how iData creates unique columns based on the journal setup as shown in *Table C.1*.

*Table C.2—Example iData Columns Based on Table C.1*

Source_Ref	JE #_Ref	Ref 1_Ref	Ref 2_Ref	Ref 3_Ref	Oblig #_Ref	Oblig #_Ref	Invoice #_Ref	Check #_Ref	Asset Num_Ref

All journal reference columns have a “\_Ref” suffix to distinguish reference information from data item information (data items are discussed in the *Appendix D—Adaptable Journal References on page 61*).

# A P P E N D I X D

## ADAPTABLE DATA ITEMS

Data items are a unique feature of FundWare. They can be used to store financial and non-financial information about an account or a transaction. Data items can be combined with the financial data to customize analysis and reports on a specific subject. *Table D.1* shows some examples of typical data items and their definitions.

*Table D.1—Example Data Items*

Item Abbreviation	Item Description	Item Short Description	Item Type
CUSTOMERS	Customers Served	Customers Servd	Signed Number
PHONE	Phone number project contact	Phone	Phone
ASSED VALU	Assessed Value	Assessed Value	Signed Number
RATE	Rate for an account	Rate	Rate
CHECK NO	Check No.	Check No.	Signed Number
DESC 1	Description Line 1	Desc 1	Text
REF P	Reference Project Name	Reference P	Keyword
INVOICE DT	Invoice Date	inv date	Date
P.O. NUMBR	Purchase Order Number	PO number	Signed Number
.YEAR END	Grant Year Ending Date	Grant Year End	Date
UNITS	Number of Units Owned	Units	Dollar Amount
UNIT PRICE	Current Unit Price	Unit Price	Dollar Amount

Figure 4.1 shows how iData creates unique columns based on the data item short descriptions in Table D.1.

<b>Customers Servd_DIGLTran</b>	<b>Phone_DIGLTran</b>	<b>Assessed Value_DIGLTran</b>	<b>Rate_DIGLTran</b>		
	<b>Check No_DIGLTran</b>	<b>Desc 1_DIGLTran</b>	<b>Reference P_DIGLTran</b>	<b>inv date_DIGLTran</b>	
		<b>PO number_DIGL Tran</b>	<b>Grant Year End_DIGLTran</b>	<b>Units_DIGLTran</b>	<b>Unit Price_DIGLTran</b>

Figure 4.1—Unique Data Items Column Headings



The check number reference, “Ck #”, on the Cash Receipts Journal CREC is in a different column than the other check number references because its name is spelled differently.



The Cash Disbursements Journal CDIS has a reference name “Oblig.#” that contains the illegal period character (.). This character is automatically replaced by iData with an underscore (\_) character and makes the reference name different than the other “Oblig.#” references.



iData does not build columns for data items that have no value stored in the source file. For example, if there was a data item “Oblig.#” and the source files had no value for that data item, then this column would not be included in the Smart View.

All data item columns have a “\_DIxxxxxx” suffix (where xxxxxx is the short name of the Smart View, such as GLTran or GLAcct) to distinguish data item information within a specific view from similar data items in another view.

The SQL Server column type (such as numeric, date/time, text, etc.) is determined by the Item Type column for the data item selected when defining data items in FundWare’s setup screens.

Notice that the data item short description “Check No.” contains the illegal period (.) character. This character is automatically replaced by iData with an underscore (\_) character and makes the data item name different.

# A P P E N D I X E

## SQL SERVER 2000 DATA TYPES

*Table E.1—SQL Server 2000 Data Types and Descriptions*

Data Types	Description
<b>Integers</b>	
bigint	Integer (whole number) data from $-2^{63}$ (-9223372036854775808) through $2^{63}-1$ (9223372036854775807).
int	Integer (whole number) data from $-2^{31}$ (-2,147,483,648) through $2^{31} - 1$ (2,147,483,647).
smallint	Integer data from $2^{15}$ (-32,768) through $2^{15} - 1$ (32,767).
tinyint	Integer data from 0 through 255.
<b>Bit</b>	
bit	Integer data with either a 1 or 0 value.
<b>Decimal and Numeric</b>	
decimal	Fixed precision and scale numeric data from $-10^{38} + 1$ through $10^{38} - 1$ .
numeric	Functionally equivalent to decimal.
<b>Money and Smallmoney</b>	
money	Monetary data values from $-2^{63}$ (-922,337,203,685,477.5808) through $2^{63} - 1$ (+922,337,203,685,477.5807), with accuracy to a ten-thousandth of a monetary unit.
smallmoney	Monetary data values from -214,748.3648 through +214,748.3647, with accuracy to a ten-thousandth of a monetary unit.
<b>Approximate Numerics</b>	
float	Floating precision number data from $-1.79E + 308$ through $1.79E + 308$ .
real	Floating precision number data from $-3.40E + 38$ through $3.40E + 38$ .
<b>Datetime and Smalldatetime</b>	

Table E.1—SQL Server 2000 Data Types and Descriptions (Continued)

Data Types	Description
datetime	Date and time data from January 1, 1753, through December 31, 9999, with an accuracy of three-hundredths of a second, or 3.33 milliseconds.
smalldatetime	Date and time data from January 1, 1900, through June 6, 2079, with an accuracy of one minute.
<b>Character Strings</b>	
char	Fixed-length non-Unicode character data with a maximum length of 8,000 characters.
varchar	Variable-length non-Unicode data with a maximum of 8,000 characters.
text	Variable-length non-Unicode data with a maximum length of $2^{31} - 1$ (2,147,483,647) characters.



# A P P E N D I X F

## WARNING AND ERROR MESSAGES

<b>Error #:</b>	1
<b>Error Message:</b>	Invalid record count
<b>Error Description:</b>	Invalid Record count
<b>Resolution:</b>	This may require some assistance from the FundWare Support Department to resolve this problem.
<b>Error #:</b>	2
<b>Error Message:</b>	Unknown acct stat desc
<b>Error Description:</b>	Unknown account status description
<b>Resolution:</b>	If an unknown Account Status is detected, then the Acct Stat Desc will be set to "Unknown". This may require some assistance from the FundWare Support Department to resolve this problem.
<b>Error #:</b>	3
<b>Error Message:</b>	Unknown user
<b>Error Description:</b>	Unknown User ID detected in data file.
<b>Resolution:</b>	If there is an Unknown User ID, the User Name field is set to "Unknown". This may require some assistance from the FundWare Support Department to resolve this problem.

<b>Error #:</b>	4
<b>Error Message:</b>	Invalid dimension name
<b>Error Description:</b>	Invalid dimension name exists with a name = Account
<b>Resolution:</b>	Change any account dimension named "Account", for example, Acct. "Account" is an iData reserved keyword for Dimension names.
<b>Error #:</b>	5
<b>Error Message:</b>	Unknown Attribute Status
<b>Error Description:</b>	An unknown attribute status was detected in the data.
<b>Resolution:</b>	If there is an Unknown Attribute Status, the Attribute Status is set to "Unknown". This may require some assistance from the FundWare Support Department to resolve this problem.
<b>Error #:</b>	6
<b>Error Message:</b>	Missing System Attribute
<b>Error Description:</b>	Missing System Attribute
<b>Resolution:</b>	Run the Account Verification report for both General Ledger and Project/Grant to identify account numbers that do not have an assigned attribute in Position 1. Assign the appropriate attribute.
<b>Error #:</b>	7
<b>Error Message:</b>	Duplicate dimension name
<b>Error Description:</b>	Duplicate dimension names exist after removing invalid characters
<b>Resolution:</b>	Duplicate dimension names are replaced with a Dimension Name_Dup#, where # indicates the occurrence. Replace any duplicate dimension names. If this is caused by the removal of illegal characters, remove the illegal characters from the Dimension Name and make the name unique.

**Error #:** 8  
**Error Message:** Duplicate data item name  
**Error Description:** Duplicate data item exist after removing invalid characters  
**Resolution:** Duplicate data item names are replaced with a Data Item Name\_Dup#, where # indicates the occurrence. Replace any duplicate data item names. If this is caused by removal of illegal characters, remove the illegal characters from the Data Item Name and make the name unique.

**Error #:** 9  
**Error Message:** Duplicate data item name  
**Error Description:** Duplicate data item exist after removing invalid characters  
**Resolution:** Duplicate data item names are replaced with a Data Item Name\_Dup#, where # indicates the occurrence. Replace any duplicate data item names. If this is caused by removal of illegal characters, remove the illegal characters from the Data Item Name and make the name unique.

**Error #:** 10  
**Error Message:** Unknown trans type  
**Error Description:** Unknown transaction type found in data.  
**Resolution:** Unknown trans type exists. The Trans Type Sh Desc is set to “Ukn” and the Trans Type Lg Desc is set to “Unknown Trans Type”. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 11  
**Error Message:** Unknown post system  
**Error Description:** Unknown post system found in data.  
**Resolution:** Unknown post system exists. The post system short desc will be set to “Ukn” and the Post system large desc will be set to “Unknown post system”. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 12  
**Error Message:** Unknown journal post type  
**Error Description:** Unknown journal post type found in data.  
**Resolution:** Unknown journal post type exists. The journal post type short desc is set to "Ukn" and the journal post type long desc is set to "Unknown Post Type". This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 13  
**Error Message:** Invalid data item name  
**Error Description:** Invalid data item name found in setup.  
**Resolution:** Run the Additional Data Item List through System Administration: **System Utilities > Execute Program > CWS110**, and change any invalid data item names.

- Data item names cannot have a numeric character for the first character of the name.

**Error #:** 14  
**Error Message:** Invalid journal reference name  
**Error Description:** Invalid journal reference name  
**Resolution:** Run the Journal Controls List for General Ledger (through Report Manager) and change any invalid journal reference names.

- Journal reference names cannot have a numeric character for the first character of the name.

<b>Error #:</b>	15
<b>Error Message:</b>	Invalid data item date
<b>Error Description:</b>	Invalid data item date
<b>Resolution:</b>	If the Invalid date found in the data is "00/00/0000" then replace with NULL else replace invalid date with 12/31/9999. SQL Server 2000 does not allow a date 00/00/0000. FundWare allows you to store a data item date, such as 12/1999, which is an invalid date in SQL Server 2000. This may require some assistance from the FundWare Support Department to resolve this problem.
<b>Error #:</b>	16
<b>Error Message:</b>	Orphaned Data Item
<b>Error Description:</b>	Orphaned Data Item
<b>Resolution:</b>	There is an orphaned data item in the transaction files. The Data item values will not be displayed. This may require some assistance from the FundWare Support Department to resolve this problem.
<b>Error #:</b>	17
<b>Error Message:</b>	Unknown Attribute
<b>Error Description:</b>	Unknown Attribute
<b>Resolution:</b>	An unknown attribute was detected in the data. The Attribute Description is set to "Unknown".
<b>Error #:</b>	18
<b>Error Message:</b>	Invalid Journal ID Number
<b>Error Description:</b>	Invalid Journal ID Number = 0 found in the data.
<b>Resolution:</b>	If there is an invalid Journal ID = 0, G/L transactions are missing. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 19  
**Error Message:** Missing GL Date Periods for Balances  
**Error Description:** Missing GL Date Periods for Balances  
**Resolution:** If there are balances for date periods that do not exist, the Acct Type and Date Period will be logged. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 20  
**Error Message:** Missing data item short description  
**Error Description:** Missing data item because short description is blank  
**Resolution:** Run the Additional Data Item List through System Administration: **System Utilities > Execute Program > CWS110**, and add short descriptions for each data item where the description is blank.

- Data exist for this data item, but iData cannot create a column based on a blank short description.

**Error #:** 101  
**Error Message:** Invalid division  
**Error Description:** An invalid Division ID has been detected.  
**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 102  
**Error Message:** Error opening input file  
**Error Description:** Either directory or file does not exist, or is locked by another process  
**Resolution:** This error can occur when the FundWare data path has been incorrectly identified, or the path has changed. Another possibility is a locked file due to a User working in FundWare while the iData build process is running. Correct any of these possibilities. Otherwise, this may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 103  
**Error Message:** Input file failed record check  
**Error Description:** One or more bad records in the file  
**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 104  
**Error Message:** Error opening output file  
**Error Description:** Directory does not exists, or existing file is locked by another process  
**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 105  
**Error Message:** FileDef not found  
**Error Description:** Missing AFWExtract registry key  
**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 106  
**Error Message:** FileSet not found  
**Error Description:** Missing AFWExtract registry key  
**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 107  
**Error Message:** Invalid mode  
**Error Description:** An invalid mode was detected.

**Resolution:** This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 108

**Error Message:** Invalid FieldDef

**Error Description:** Missing AFWExtract registry key or value in FieldDef

**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 109

**Error Message:** Invalid FileDef

**Error Description:** Missing AFWExtract registry key or value in FileDef

**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 110

**Error Message:** Invalid FileSet

**Error Description:** Missing AFWExtract registry key or value in FileSet

**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 111

**Error Message:** Invalid Output Position for FieldDef

**Error Description:** Extract fields defined in registry must have output positions in a continuous range

**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.



**Error #:** 112  
**Error Message:** Did not process all records  
**Error Description:** Extract process halted before all records were processed, most likely due to file corruption  
**Resolution:** Rebuild file with FundWare or Microfocus file utilities. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 113  
**Error Message:** Error reading FileDefs key  
**Error Description:** Missing AFWExtract registry key or value in FileDef  
**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 114  
**Error Message:** Error reading Find/Replace info  
**Error Description:** Missing Fundware/AFWExtract/Replace registry key  
**Resolution:** Correct registry problem. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 115  
**Error Message:** Error opening log file  
**Error Description:** Directory does not exist, or existing file is locked by another process  
**Resolution:** Resolve file/directory conflict. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 116  
**Error Message:** Error writing to output file  
**Error Description:** Error writing to output file - Typically caused by being out of disk space.  
**Resolution:** Check return message. Resolve file/directory/disk conflict. This may require some assistance from the FundWare Support Department to resolve this problem.

**Error #:** 117  
**Error Message:** Error trying to allocate buffer space  
**Error Description:** Program could not allocate internal memory for processing.  
**Resolution:** Check system resources, other memory intensive applications.

# INDEX

- A**
- Account
  - Dimensions ..... 11
- Account Dimensions
  - Adaptable ..... 57
- Adaptable
  - Account Dimensions ..... 57
  - Data Items ..... 63
  - Journal References ..... 61
  - Period Information ..... 59
- Adding
  - Security settings ..... 33
  - SQL Server logins ..... 17
- Administration ..... 23, 29
  - Data Paths/Servers ..... 2
  - FundWare server ..... 25
  - General Setup ..... 2, 32
  - iData Server ..... 25
  - Process Log ..... 2, 39
  - Scheduling ..... 2
  - Scheduling a Build ..... 29
  - Security ..... 2, 33
- Assigning security settings ..... 33
- B**
- Business Office ..... 3
- C**
- Closing
  - iData Administration ..... 41
- Columns
  - Illegal names ..... 7
- Copying
  - Division security settings ... 36, 37
  - Users to security settings ..... 35
- D**
- Data Dictionaries
  - GL Accounts ..... 44
  - GL Balances ..... 51
  - GL Transactions ..... 46
- Data dictionary ..... 3
- Data Items ..... 12
  - Adaptable ..... 63
- Data scrubbing ..... 5
- Deleting
  - Security settings ..... 38
  - SQL Server logins ..... 21
- Dimensions
  - Account ..... 11
- Division security
  - Copying ..... 36, 37
- Divisions
  - FundWare ..... 3
- E**
- Extraction
  - Rules ..... 5
- F**
- FundWare
  - Server ..... 25
- FundWare divisions ..... 3
- G**
- Getting ..... 3

GL Account Component	
Account	44
Account Dimension	44, 45
Attribute	44, 45
Data Item	44, 46
GL Accounts	8
Smart View	44
GL Balance Component	
Account Dimension	51, 54
Balances	51, 52
Posted Units	51, 55
GL Balances	10
GL Transaction Component	
Account	46, 49
Data Item	46, 51
Period	46, 50
Transaction	46
Transaction Reference	46, 51
GL Transactions	9
<b>I</b>	
iData	
Server	25
ISAM	
Time component	6
<b>J</b>	
Journal References	
Adaptable	61
Journals	
References	11
<b>M</b>	
Modifying	
SQL Server logins	20
<b>N</b>	
Null	5
<b>O</b>	
Overriding	
Scheduled build	31
<b>P</b>	
Period Information	
Adaptable	59
Process Log	
Administration	39
<b>R</b>	
References	
Journal	11
Removing	
Security settings	38
Restarting SQL Server 2000	13
Rules	
Extraction	5
Run now	31
<b>S</b>	
Scheduling a Build	29
Security	
Administration	33
Assigning settings	33
Copying division settings	36, 37
Copying users	35
Removing settings	38
Smart View	
GL Accounts	44
Smart views	2
SQL Server 2000	
Adding logins	17
Date/time fields	6
Deleting logins	21
Modifying logins	20
Restarting	13

**T**  
Technical support .....4  
Training .....4

**U**  
User security  
    Assigning .....33

    Copying .....35

**V**  
vwGLAccount .....2  
vwGLBalance .....2  
vwGLTransaction .....2

