RoLL NO. 9

105 | FOUNDATIONS OF ACL

CONCEPTS AND PRACTICES

ACL™

Data you can trust. Results you can see.

ACL Services Ltd.
1550 Alberni Street
Vancouver, BC
Canada V6G 1A5

Telephone: +1-604-669-4225
Fax: +1-604-669-4087
E-mail: info@acl.com
Web: www.acl.com

Printed in Canada

ITDE18025
Edition 4.3.2
May 1, 2008

Contents

Understanding Database Concepts

Data

Data is everywhere. Every time you use a credit card, make a telephone call, or log on to a website, you leave a digital footprint. Yet few of us understand what data is.

For example, consider the following string of numbers:

04092003

You might interpret this string of numbers as:

An account number	04092003
A transaction amount	$40,920.03
A date	April 9, 2003 or, depending on your date preference, September 4, 2003

To work with raw data such as this string of numbers, you must know how to interpret it and what to do with it. In the same way, a computer must first be told how to interpret raw data and how to manage it.

For example, assume an employee enters a duplicate employee number so that he or she receives two checks for the same pay period. To identify this problem, the computer must be programmed to interpret the raw data as employee numbers and to identify the duplicate employee numbers as irregularities.

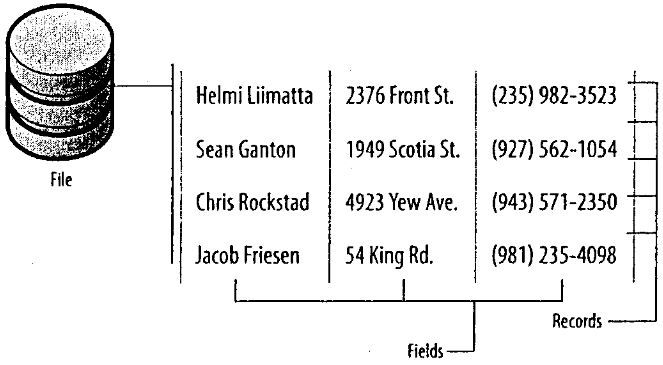
Files, Records, and Fields

Modern data structures are built from files. A file is a named collection of information stored or processed as an individual entity.

A file is composed of a number of smaller units, called records and fields. A record is a collection of related information grouped for processing. A field is an area in a record where a particular class of data is stored.

A useful analogy for understanding these concepts is the telephone book, which is a collection of information about the telephone users in a particular community.

A telephone book can be thought of as a file. Each entry for a telephone user is a record and each type of information repeated in every record—the name, address, and telephone number—is a field.



Helmi Liimatta	2376 Front St.	(235) 982-3523
Sean Ganton	1949 Scotia St.	(927) 562-1054
Chris Rockstad	4923 Yew Ave.	(943) 571-2350
Jacob Friesen	54 King Rd.	(981) 235-4098

fig. 1-1 Data structure sample

Fundamentals

1

About this Module

- Understanding Database Concepts
- Understanding ACL
- Viewing ACL
- Understanding Data Analysis

To use *ACL* effectively, it is necessary to understand key data analysis concepts and the data analysis cycle. You must learn how to navigate the *ACL* interface, how to store and organize your work in *ACL* projects, and how to work with the many analytical tools *ACL* provides. This module introduces you to data analysis and the basics of working with *ACL* software.

Understanding ACL

ACL is a tool that lets you read and analyze data. The data may be in different types of files scattered across numerous databases on different platforms. Regardless, *ACL* can help you access the data, ask intelligent questions about it, and distill the information needed to answer those questions.

Who Uses ACL?

ACL is used by data analysts, auditors, accountants, and other business professionals who need timely access to data and the means to analyze it efficiently and effectively. The information *ACL* provides allows confident management planning and decision-making.

What Can I Do with ACL?

Your use of *ACL* is limited only by your imagination. Because *ACL* provides such a huge range of options, you can accomplish almost any goal in the realm of data analysis.

Gather essential information for decision-making

ACL offers a full range of analytical tools, including functions, commands, and scripts. These tools let you find answers to your questions quickly and easily.

Retain the integrity of your data

Because *ACL* is a read-only application, it cannot alter the data source. This lets you access production data directly and safely.

Process different data types from different systems

ACL can combine data from modern databases, legacy systems, and report files. You can then analyze the data as if it was all from the same source.

Process files of any size rapidly

Unlimited file-size capability and high-speed processing make it possible to analyze millions of records in seconds with *ACL*.

Analyze large data sets in their entirety

The large capacity and high speed of *ACL* allow you to work with files in their entirety instead of through samples. Knowing that you have examined an entire file gives you greater confidence that your results are complete and correct.

Automate analytical procedures

Automating your analysis can save time and effort. *ACL* scripts can be programmed to be interactive or to run as continuous monitoring applications.

Maintain a record of your work

When you use *ACL*, every step you take is recorded in the log. This log, or audit trail, lets you review, document, and repeat the steps of your analysis.

Viewing ACL

ACL provides the means to access and analyze your data through:

- Multi-level organization within a project
- Easy creation and manipulation of project elements
- Analytical tools such as functions, commands, and scripts

The ACL Interface

When you open *ACL*, you are presented with the application window showing the Welcome Tab, the Project Navigator, and the Status Bar.

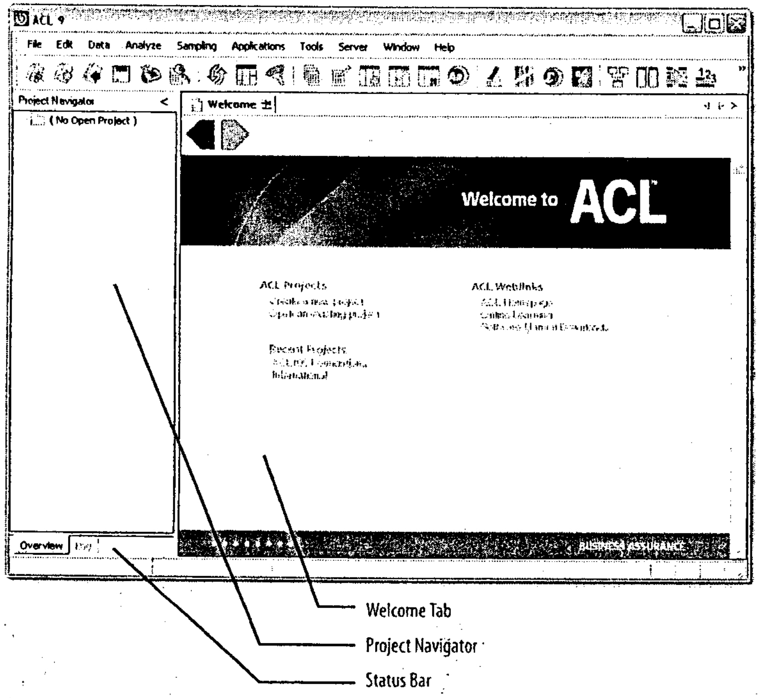


fig. 1-2 ACL interface

Welcome tab

The Welcome tab is a fast and easy way to navigate between your projects in *ACL*. By clicking the links provided, you can create a project, open a project, or access different parts of the ACL website. When online, you can download product enhancements, submit support requests, and search the knowledge base.

Project Navigator

The Project Navigator has an Overview tab and a Log tab. The Overview tab lets you organize items in your *ACL* project into folders. The Log tab lets you browse a chronological record of your past activities, view command results, and save portions of the log to another file.

Status bar

The status bar displays information about the active table, including the table name, the number of records, and any active filter criteria.

The ACL Project

Similar to a top-level folder in Windows, an *ACL* project is used to store and organize your analytical work. An *ACL* project can include tables, scripts, logs, workspaces, and folders.

Tables

A table is the combination of a table layout and the data source to which the table layout is linked. The data in a table is presented in a view.

Data source	A data source provides the data that populates a table. Every table layout is linked directly to an original data source or to a copy of the data. The data source does not reside in the project.
Table layout	A table layout contains everything needed to read a data file, including media and file type, record length, and field information.
View	A view is the visual presentation of a table. A table can have more than one view, letting you create different presentations by reformatting the data.

Scripts

A script is a series of *ACL* commands that can be executed repeatedly within your project. Almost all commands can be executed in a script. Scripts can be designed to run automatically or to prompt the user for input.

Logs

ACL automatically creates a log when you create an *ACL* project. The log is an audit trail, recording the commands and results from your analysis. This trail lets you retrace your steps and reproduce results without always having to save your work. The log is also an excellent resource for generating reports at the conclusion of your analysis.

Workspaces

A workspace is a set of field definitions that has been saved for reuse with other tables. When a workspace is activated, the field definitions within it are available for use with the active table. Workspaces let you maintain and reuse definitions of physical fields, computed fields, and filters, so you do not have to define these elements more than one time.

Folders

Folders are used to organize your work in an *ACL* project. *ACL* folders are similar to Windows folders.

Notes:

Understanding Data Analysis

Elements of Data Analysis

The main elements of data analysis in *ACL* are commands, expressions, functions, and variables.

Commands

ACL commands are means of executing standard analyses, such as statistical analysis and data profiling. Command results can be sent to file, screen, print, and in some cases, graphs.

Expressions

Expressions are statements used primarily to create filters and computed fields. They perform calculations, specify logical conditions, or create values that do not exist in the data file. Expressions can be named and saved as part of a project or else created just for immediate use.

Filters A filter is a logical expression that lets you select the kind of data you want to see and work with. For example, you can create a filter that selects only records that fall within a specified range of dates.

Computed fields A computed field, also known as a calculated field, is a virtual field that uses data derived from an expression or a specified constant. It does not contain physical data. For example, you can create a new field that is the product of the values in two other fields. You can also insert into a table a specific value such as an interest rate or a logical condition.

Functions

A function is a predefined routine that performs a computation on specified data. A function uses variables to accept input data values. *ACL* currently offers approximately 80 different functions.

Variables

A variable is a named memory space that stores data. The value stored in a variable can be either a character, numeric, date, or logical type. Variables are widely used in functions, expressions, and scripts.

The Data Analysis Cycle

Every data analysis project you undertake follows a regular cycle of activities. This helps to ensure that your work is thorough, accurate, and efficiently performed.

The data analysis cycle has five stages:

- Planning
- Data access
- Data integrity verification
- Data analysis
- Reporting results

Each stage is discussed in detail in this course.

Planning

Plan your work before you start a project. By formulating clear objectives before you begin your analysis, you can develop concise strategies and budget the right amount of time and resources.

Data Access

The next step is to access the data outlined in your strategic plans. This requires locating, requesting, and transferring the data prior to reading it with *ACL*.

Data Integrity Verification

Having received the data, it is necessary to test its integrity. If you start your project without having first verified the data's accuracy and completeness, your results may be unreliable or incorrect.

Data Analysis

In the analysis stage, you perform the tests necessary to achieve your objectives. You are likely to use a combination of commands, filters, and computed fields in your analysis.

Reporting Results

Depending on the project, you may need to create reports from your results. *ACL* can create different types of reports; including multiline, detail, and summary reports.

Notes:

2

Data Access

About this Module

Understanding the Data Access Process

Locating Data

Acquiring Data

Accessing Data

Creating Tables to Access Data

Reusing Table Layouts

Data access is a critical part of any data analysis project. You need to find and acquire necessary data in an efficient and effective manner. Because of the variety of data sources and technologies, data access methods vary. *ACL*, however, can access almost any kind of data. This module helps you understand the process of data access using *ACL*.

Understanding the Data Access Process

In order to prepare your analysis, you need to understand the data access process. To ensure the success of the entire process, pay careful attention to each step.

To prepare your analysis:

1. Locate the data.

   Locating data requires understanding the individual files available, the various file types and the technologies used to access them. A comprehensive understanding also helps you build a productive relationship with the data provider.

   - Formulate clear objectives.

     Before attempting to acquire data, use your analytical objectives to determine which data files you need.

   - Find out what data is available to you.
   - Find out the format in which the data is stored.

2. Acquire the data.

   - Create a request letter for the data provider that shows your specific requirements.

3. Access the data.

   - Use *ACL* to read almost any kind of data.

     ACL accesses some data directly at its source and other data through an imported copy. In some cases, *ACL* can create a table automatically. In other cases, you define aspects of the structure for it.

Notes:

Locating Data

If you are to make an informed request to the data provider, you need to learn what data is available to you and the format in which it is stored. Effectively communicating your data needs will save time for both you and your Information Systems (IS) department.

Creating Partnerships with Information Staff

You rely heavily upon IS to help you find data. To foster a positive working relationship, learn the language of technology and educate them about your needs. Find out what information they need from you to process your data request and develop a request form that conveys that information.

Educating Yourself and Your Staff

Understanding the fundamentals of data can help you to phrase your data request succinctly. Learn the characteristics of the different file formats available. Investigate data access and file transfer methods. Encourage your staff to educate themselves about data technology. Understanding the fundamentals of data will help your team build a productive relationship with Information staff.

Identifying Available Data

Find out what data exists and where it is stored. Ask to review the data and the data dictionary to assess what information you may require. Develop a map that documents data sources, file formats, and the contents of specific files and fields. Use the map as a reference for future projects. As your knowledge increases, your reliance on IS or a system administrator will decrease.

Determining Available Data Formats

ACL can create tables from many data formats with varying degrees of user intervention. When you have a choice, use data in its native format. Also, work with data formats that *ACL* can define with little or no help, such as ODBC-compliant sources.

ACL can access most file types, including the following:

ODBC-compliant data sources	Open Database Connectivity (ODBC) is a Microsoft technology that provides data access with minimal user input. Most modern database systems are ODBC-compliant.
dBASE files	dBASE is the fastest and easiest format for *ACL* to read. Few modern database systems store data in dBASE files, however, but most systems can easily export data to this format.
Flat files	Flat files contain only data. Unlike formats such as Excel and dBASE, flat files contain no field definition information. For *ACL* to create a table layout, you must define the file manually or use an external definition file.
Report (print-image) files	Sometimes you will not be able to acquire direct access to the data in one of the file formats discussed in this section. In these cases, you may be able to get an electronic copy of a report file. *ACL* can read report files, but you must define the table layout manually.
Delimited files	Delimited files are typically variable-length files with one record per line of text. Special characters such as commas or tabs are used to separate the fields in each record. CSV files (comma-separated values files) are the most common type of delimited file.

XML files	ACL has developed and patented technology used to process XML files so they can be analyzed like other data sources. XML is a general purpose markup language that combines data and the formatting information in the same document. XML is the base for other special-purpose markup languages, for example XBRL and XSLT.

These formats may not be the easiest for your data provider to supply, so you may have to compromise.

ⓘ *For more information on file formats, see the Data Access Guide, or select Help > Index and enter "data sources".*

Notes:

Acquiring Data

After you develop an overview of the available data, you need to acquire the data. To do that, you must do the following:

1. Determine your analytical objectives.
2. Request the data.
3. Transfer the data.

Determining Your Analytical Objectives

Before requesting data, formulate your analytical objectives. Your objectives determine the data that you require.

Consider the following example: You want to identify vendors that may be overcharging you for supplies than you have contracted. You have located the data sources and know that three relevant files exist:.

File	Contains
Purchases	Purchase transactions
Vendors	List of vendors from whom you purchase goods
Supplies	Master list of supplies, including the prices agreed upon by each vendor

To identify the vendors that may be overcharging, you must compare records in the Purchases file to records in the Supplies file. Therefore, you need to request these two files. Request the Vendors file only if you want to see vendor information for each instance of an overcharge.

Requesting Data

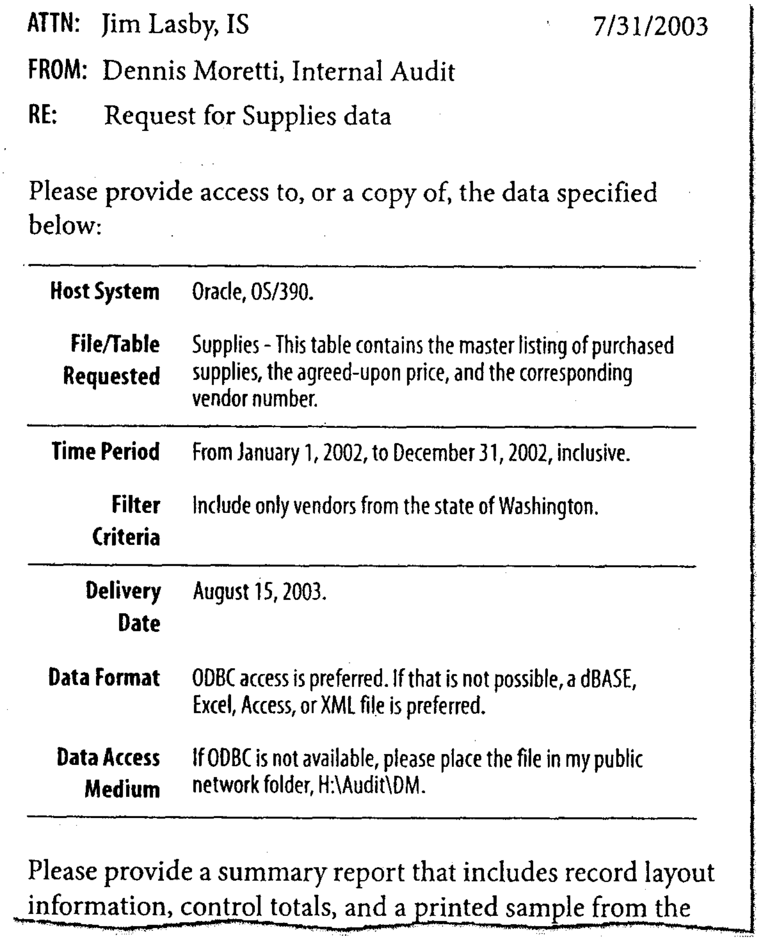
Requesting data is a critical step in a data analysis project. Complications with data access can seriously compromise the timeline of your analysis. It is essential that you be very specific about what data you need, where it comes from, and how you would like it delivered. To accomplish this, you need to submit a data request letter and request a summary report.

Data request letter

Specify what data you require in a formal request. Your request letter needs to have all the information that IS needs to provide the data.

Include the following in your data request letter:

- The date of your request
- The name of the system where the data resides
- The file name, table name, or report name
- Date bounds for the data. For example, you might specify from January 1, 2002, to December 31, 2002, inclusive of start and end dates.
- Any filter criteria. For example, you might specify to include only positive values in the balance field.
- When you would like the data delivered
- The file format in which you would like to receive the data
- The medium on which you would like the data transferred



ATTN: Jim Lasby, IS 7/31/2003

FROM: Dennis Moretti, Internal Audit

RE: Request for Supplies data

Please provide access to, or a copy of, the data specified below:

Host System	Oracle, OS/390.
File/Table Requested	Supplies - This table contains the master listing of purchased supplies, the agreed-upon price, and the corresponding vendor number.
Time Period	From January 1, 2002, to December 31, 2002, inclusive.
Filter Criteria	Include only vendors from the state of Washington.
Delivery Date	August 15, 2003.
Data Format	ODBC access is preferred. If that is not possible, a dBASE, Excel, Access, or XML file is preferred.
Data Access Medium	If ODBC is not available, please place the file in my public network folder, H:\Audit\DM.

Please provide a summary report that includes record layout information, control totals, and a printed sample from the

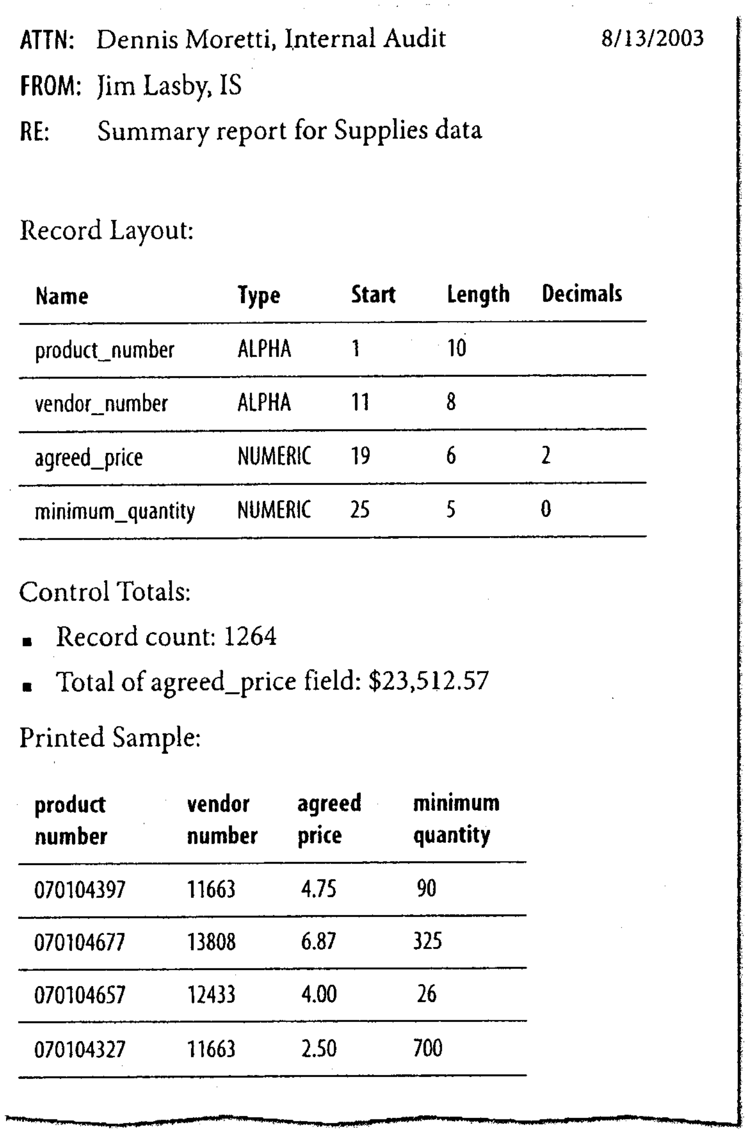
fig. 2-1 Sample data request letter

Summary report

You should also request a summary report from IS about the data you receive. If you are able to get a summary report, use the information in the report to confirm that you have received all the data and only the data that you requested. If possible, obtain the same details from an independent source, such as the department from which the data originates.

Request the following items in your summary report:

- Record layout information (field names, start positions, lengths, data types, and formatting details)
- Control totals (number of records, field totals)
- A printed sample of approximately fifty records from the original file. Compare the sample records against the table you create to confirm the accuracy of the table.



ATTN: Dennis Moretti, Internal Audit 8/13/2003

FROM: Jim Lasby, IS

RE: Summary report for Supplies data

Record Layout:

Name	Type	Start	Length	Decimals
product_number	ALPHA	1	10	
vendor_number	ALPHA	11	8	
agreed_price	NUMERIC	19	6	2
minimum_quantity	NUMERIC	25	5	0

Control Totals:

- Record count: 1264
- Total of agreed_price field: $23,512.57

Printed Sample:

product number	vendor number	agreed price	minimum quantity
070104397	11663	4.75	90
070104677	13808	6.87	325
070104657	12433	4.00	26
070104327	11663	2.50	700

fig. 2-2 Sample summary report

Transferring Data

The last step in acquiring data is transferring it. You must either obtain a copy of the data and store it on a drive where *ACL* can read it or you must be granted access rights to the data source.

Gaining access to data in a way that allows you to work with it can be difficult. The method that you use depends on file size, storage medium, and available technology. Discuss the options with IS to determine the method that works best for both of you.

The IS department can provide you with at least one of the following modes of transfer.

Access to a production database or data warehouse

You are granted access from *ACL* to either a production database or a regularly refreshed copy of a database in a data warehouse. *ACL* reads the data directly from the source using data access technologies such as ODBC.

Access to a copy of the data

You are provided with a copy of the data source in a form that *ACL* can read. Some file types contain metadata that *ACL* can use to structure the data. Other file types require manual definition.

The size of the file will dictate the medium on which the file is provided. You will typically receive files on a CD-RW, Zip disk, DVD-R, or via a network drive e-mail.

User access to the data source

You are provided with a user name, password, and instructions to access the host system. You must extract the data yourself. Many systems have an extraction tool you can use to create files that *ACL* can read.

Accessing Data

When the data has been made available in a usable format, you can begin the process of accessing it with *ACL*. *ACL* reads different file types in different ways and with varying degrees of automation. Data from newer technologies requires less manual processing than data from legacy systems.

There are two methods that *ACL* uses to accesses data:

Direct access	*ACL* reads the data directly from the source in its native format without creating a copy of the data. With direct access *ACL* creates tables with varying degrees of automation depending on how much layout information the source contains. *ACL* populates tables with data read directly from the source.
Import and copy	*ACL* imports the data as a flat file and copies it to a local drive. Details on the organization and structure of the data are maintained together with the information. *ACL* reads the structure to define the layout.

Notes:

Creating Tables to Access Data

Before you can analyze data with *ACL*, you must create tables to access the data. The following procedures describe how to use the Data Definition Wizard to access data from different data sources.

This section describes the following:

- Creating Tables from Flat Files
- Creating Tables from MS Excel Data
- Creating Tables from MS Access Data
- Creating Tables from ODBC-Compliant Data
- Creating Tables from dBASE Data
- Creating Tables from Report Files
- Creating Tables from Adobe PDF Data

Creating Tables from Flat Files

Creating Tables from Flat Files uses the direct access, manual layout method to access data from a flat file named Inventory.fil.

Direct access, manual layout

ACL can read any flat file, including data from older mainframe systems, and telephone and internet logs. These files do not contain table layout information, so you must provide the information to *ACL*.

To access data using direct access, manual layout:

1. Get a hard copy of the record layout information.
2. Locate the data source.
3. Input the record layout information in *ACL*.

ACL creates a table containing data read directly from the data source.

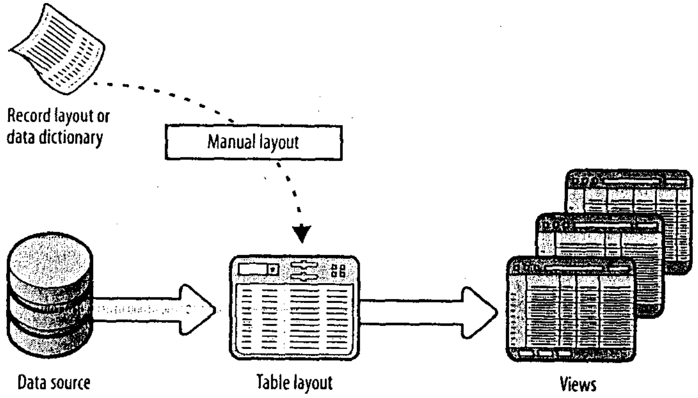


fig. 2-3 Direct access, manual layout

Select the flat file

1. Select **File** > **New** > **Table.**

   The Data Definition Wizard starts.

2. Click **Next.**

   By default, the Data Definition Wizard selects Disk as the data source.

3. Click **Next.**
4. In Select File to Define, select `Inventory.fil` and click **Open.**

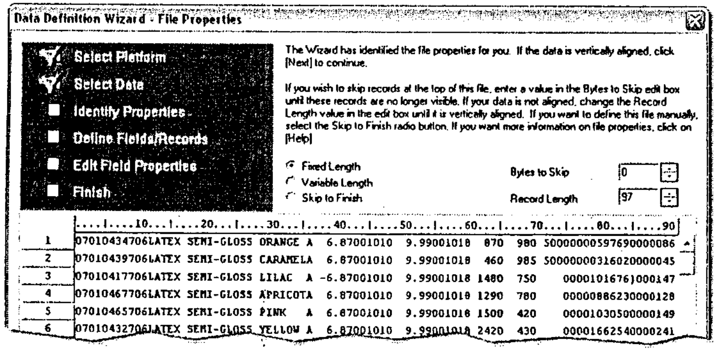
   The Data Definition Wizard correctly identifies the character set as ASCII.

5. Click **Next.**

The Data Definition Wizard correctly identifies the file format as Other.

6. Click **Next.**

   On the File Properties window, the Data Definition Wizard identifies the records as fixed length, with a length of 97 bytes each.



The Wizard has identified the file properties for you. If the data is vertically aligned, click [Next] to continue.

If you wish to skip records at the top of this file, enter a value in the Bytes to Skip edit box until these records are no longer visible. If your data is not aligned, change the Record Length value in the edit box until it is vertically aligned. If you want to define this file manually, select the Skip to Finish radio button. If you want more information on file properties, click on [Help]

● Fixed Length
○ Variable Length
○ Skip to Finish

Bytes to Skip 0

Record Length 97

```
    ....|....10...|....20...|....30...|....40...|....50...|....60...|....70...|....80...|....90
1   07010434706LATEX SEMI-GLOSS ORANGE A  6.87001010  9.99001018  870  980 50000000597690000086
2   07010439706LATEX SEMI-GLOSS CARAMELA  6.87001010  9.99001018  460  985 50000000316020000045
3   07010417706LATEX SEMI-GLOSS LILAC  A -6.87001010  9.99001018 1480  750     0000101676}000147
4   07010467706LATEX SEMI-GLOSS APRICOTA  6.87001010  9.99001018 1290  780     00000886230000128
5   07010465706LATEX SEMI-GLOSS PINK   A  6.87001010  9.99001018 1500  420     00001030500000149
6   07010432706LATEX SEMI-GLOSS YELLOW A  6.87001010  9.99001018 2420  430     00001662540000241
```

fig. 2-4 Data Definition Wizard - File Properties

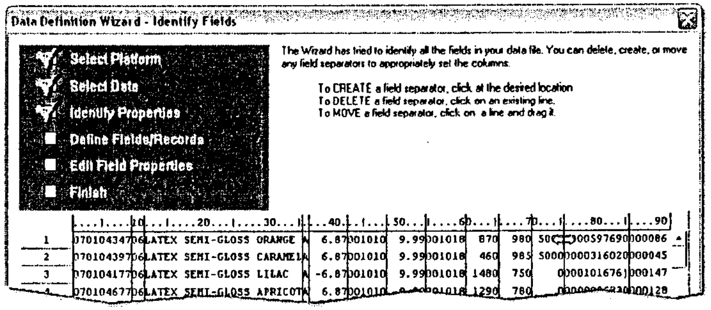
ⓘ *If you find the layout of the data is skewed diagonally, you must adjust the record length until the fields are aligned vertically.*

7. Click **Next.**

   The File Type window shows that the Data Definition Wizard correctly identified the file as a single-record-type data file.

8. Click **Next.**

   The Identify Fields window shows the Data Definition Wizard's placement of field separators. Confirm the Data Definition Wizard's choices and adjust if necessary.



The Wizard has tried to identify all the fields in your data file. You can delete, create, or move any field separators to appropriately set the columns.

To CREATE a field separator, click at the desired location
To DELETE a field separator, click on an existing line.
To MOVE a field separator, click on a line and drag it.

```
    ....|....|10...|....20...|....30...||...40.|..|...|50...|....60...|....70...|....80...|...90
1   070104347|06|LATEX SEMI-GLOSS ORANGE |A|  6.87|001010|  9.99|001018|  870|  980| 50C=|00597690|000086
2   070104397|06|LATEX SEMI-GLOSS CARAMEL|A|  6.87|001010|  9.99|001018|  460|  985| 5000|00031602|0000045
3   070104177|06|LATEX SEMI-GLOSS LILAC  |A| -6.87|001010|  9.99|001018| 1480|  750|     0|000101676}|000147
4   070104677|06|LATEX SEMI-GLOSS APRICOT|A|  6.87|001010|      |001018| 1290|  780|
```

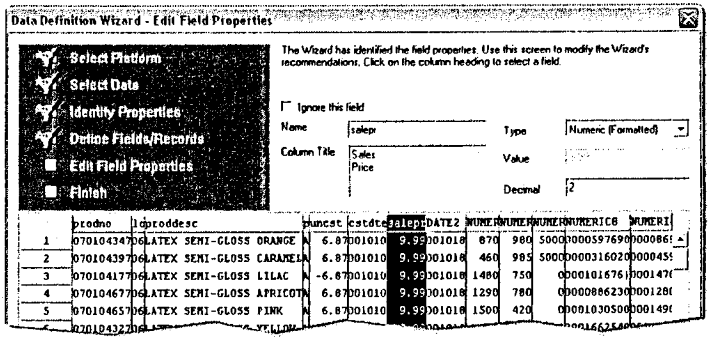
fig. 2-5 Data Definition Wizard - Identify Fields

9. As in *fig. 2-5*, shift the field separator from byte 75 to byte 76.

Name the fields and select data types

1. Click **Next.**

   The Edit Field Properties window shows the field properties.



The Wizard has identified the field properties. Use this screen to modify the Wizard's recommendations. Click on the column heading to select a field.

□ Ignore this field

Name salepr

Column Title Sales Price

Type Numeric (Formatted)

Value

Decimal 2

```
    prodno   |lo|proddesc                |p|uncst |cstdte|salepr|DATE2 |NUMER|NUMER|NUMER|NUMERIC8|NUMERI
1   070104347|06|LATEX SEMI-GLOSS ORANGE |A|  6.87|001010|  9.99|001018|  870|  980| 5000|00597690|000086
2   070104397|06|LATEX SEMI-GLOSS CARAMEL|A|  6.87|001010|  9.99|001018|  460|  985| 5000|00031602|000045
3   070104177|06|LATEX SEMI-GLOSS LILAC  |A| -6.87|001010|  9.99|001018| 1480|  750|    0|000101676}|000147
4   070104677|06|LATEX SEMI-GLOSS APRICOT|A|  6.87|001010|  9.99|001018| 1290|  780|    0|00008862|30001280
5   070104657|06|LATEX SEMI-GLOSS PINK   |A|  6.87|001010|  9.99|001018| 1500|  420|    0|00010305|00001490
```

fig. 2-6 Data Definition Wizard - Edit Field Properties

2. Click a column heading to select a field and do the following:

   a. Enter a name to identify the field in the table layout.

   b. Enter a column heading to identify the field in the view.

   c. Select the data type from the **Type** drop-down list and adjust the format if necessary.

Use the following table layout as a guide:

Start	Length	Name	Data Type	Decimals
1	9	prodno	ASCII	
1	2	prodcls	ASCII	
10	2	location	ASCII	
12	24	proddesc	ASCII	
36	1	prodstat	ASCII	
37	6	uncst	NUMERIC	2
43	6	costdate	Date	YYMMDD
49	12	Undefined area		
61	5	qtyoh	NUMERIC	0
66	10	Undefined area		
76	10	value	ZONED	2
86	10	Undefined area		

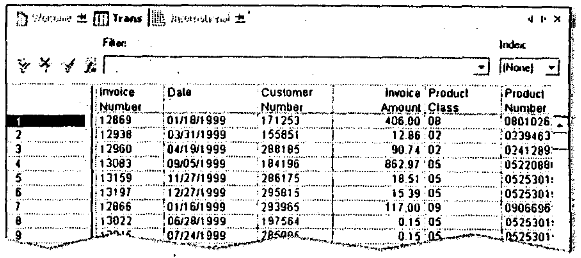
3. For all fields you want to define, repeat Step 2.
4. Click **Next**.

   The Final window shows a summary of table record and field properties.

Name and save the table

1. Click **Finish**.
2. Name the table `Inventory` and click **OK**.

   The Data Definition Wizard creates the table and displays the default view.



	Invoice Number	Date	Customer Number	Invoice Amount	Product Class	Product Number
1	12869	01/18/1999	171253	406.00	08	0801026
2	12938	03/31/1999	155851	12.86	02	0239463
3	12960	04/19/1999	288185	90.74	02	0241289
4	13083	09/05/1999	184196	862.97	05	0522088
5	13159	11/27/1999	286175	18.51	05	0525301
6	13197	12/27/1999	295815	15.39	05	0525301
7	12866	01/16/1999	293965	117.00	09	0906696
8	13022	06/28/1999	197564	0.15	05	0525301
9	[illegible]	07/24/1999	[illegible]	0.15	05	0525301

fig. 2-7 Table in default view

Defining overlapping fields

After you exit the Data Definition Wizard, you can define overlapping fields. This is, you can redefine a portion or all of an already defined field as a different field.

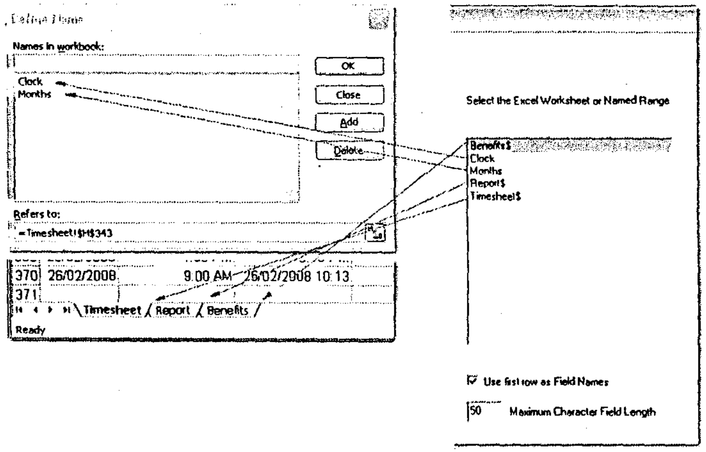
To define an overlapping field:

1. Select **Edit** > **Table Layout** and click the 🖫 (Add a New Data Field) button.
2. In the **Name** box, enter a name for the new field. Select the data type from the **Type** drop-down list, and enter the start position, field length, and any other appropriate information.
3. Click the ✓ (Accept Entry) button.

ⓘ *To add the new field to your view, right-click in the view, and select Add Columns. In the Available Fields list, double-click the new field and click* OK.

Creating Tables from MS Excel Data

ACL streamlines the creation of tables from Excel data. *ACL* can read multiple worksheets, named ranges, and pivot tables.



Refers to: =Timesheet!H343

370 26/02/2008 9:00 AM 26/02/2008 10:13

☑ Use first row as Field Names

50 Maximum Character Field Length

fig. 2-8 Excel to ACL

In *fig. 2-8*, *ACL* reads worksheets and named ranges in an Excel file. Worksheets appear with a dollar sign ($) after their name and named ranges appear without.

ⓘ *If the Excel file is using security features, you must open the Excel file separately and disable any password protected worksheets before proceeding in ACL.*

The following procedure uses the import and copy, automatic layout method to access a worksheet from a Microsoft Excel file named Credit_cards_metaphor.xls.

In this procedure, you complete the following:

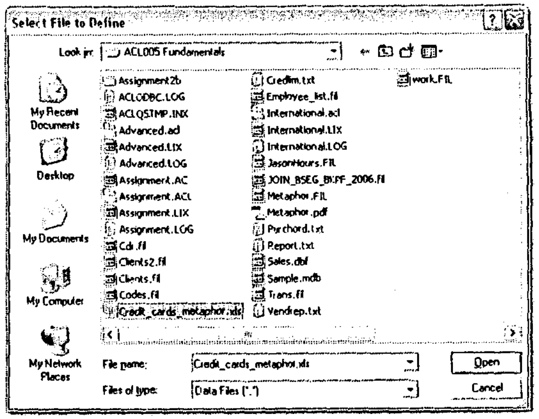
- Select the Excel file
- Select the worksheet or named range
- Create and save the table

Select the Excel file

1. Select **File** > **New** > **Table** to start the Data Definition Wizard.
2. Click **Next**.

   By default, the wizard selects Disk as the data source.
3. Click **Next**.

   The Select File to Define dialog box appears.



Look in: ACL005 Fundamentals

File name: Credit_cards_metaphor.xls

Files of type: Data Files (*.*)

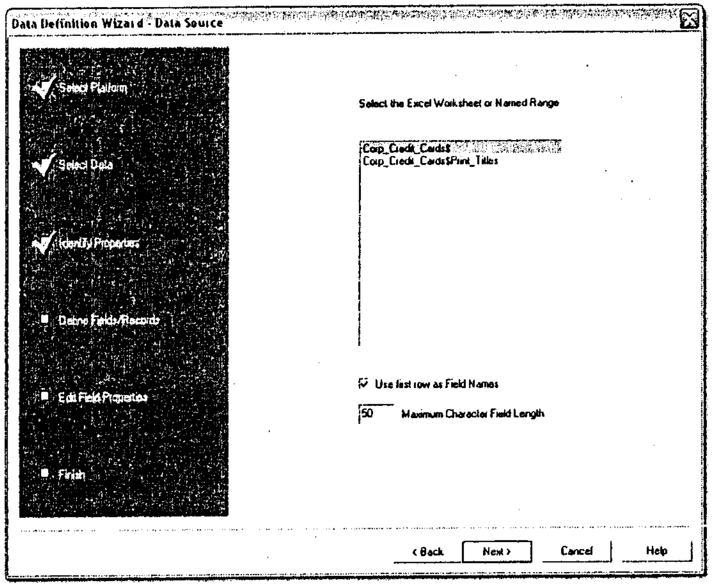
fig. 2-9 Select File to Define dialog box

4. Select Credit_cards_metaphor.xls and click **Open**.

   The wizard correctly identifies the character set as ASCII.
5. Click **Next**.

   The wizard correctly identifies the file as an Excel file.
6. Click **Next**.

The Data Definition Wizard - Data Source dialog box appears.



Data Definition Wizard - Data Source

Select the Excel Worksheet or Named Range

Corp_Credit_Cards$
Corp_Credit_Cards$Print_Titles

☑ Use first row as Field Names

50 Maximum Character Field Length

fig. 2-10 Data Definition Wizard - Data Source

Select the worksheet or named range

1. In the Data Source dialog box, select `Corp_Credit_Cards$`.

   If necessary, adjust the maximum character field length.

2. Click **Next**.

   The Save Data File As dialog box appears.

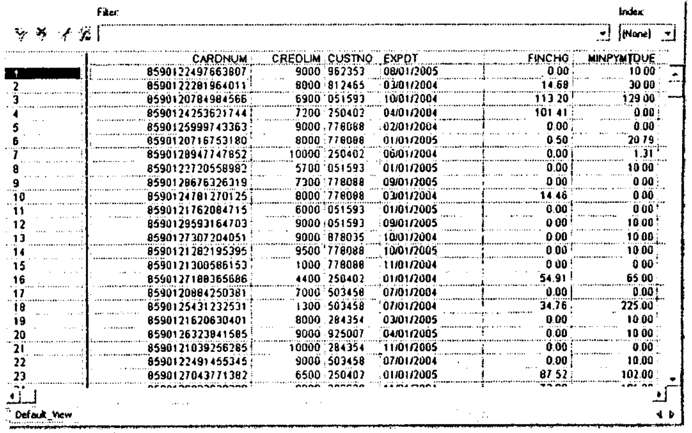
3. Name the file `Credit_cards_metaphor` and click Save.

   The wizard copies the data to an *ACL* data file and shows you a summary of the table's record and field properties.

Create and save the table

1. Click **Finish**.
2. Name the table `Credit_cards_metaphor` and click **OK**.

   The wizard creates the table and displays the default view. The Data Definition Wizard creates the table.



Filter:

Index: (None)

	CARDNUM	CREDLIM	CUSTNO	EXPDT	FINCHG	MINPYMTDUE
1	8590122497663807	9000	962353	08/01/2005	0.00	10 00
2	8590122281964011	8000	812465	03/01/2004	14.68	30 00
3	8590120784984566	6900	051593	10/01/2004	113 20	129 00
4	8590124253621744	7200	250402	04/01/2004	101 41	0.00
5	8590125999743363	9000	778088	02/01/2004	0.00	0.00
6	8590120716753180	8000	778088	01/01/2005	0.50	20 79
7	8590128947747852	10000	250402	06/01/2004	0.00	1.31
8	8590122720558982	5700	051593	01/01/2005	0.00	10 00
9	8590128676326319	7300	778088	09/01/2005	0 00	0 00
10	8590124781270125	8000	778088	03/01/2004	14.46	0 00
11	8590121762084715	6000	051593	01/01/2005	0.00	0 00
12	8590129593164703	9000	051593	09/01/2005	0 00	10.00
13	8590127307204051	9000	878035	10/01/2004	0.00	10 00
14	8590121282195395	9500	778088	10/01/2005	0.00	10.00
15	8590121300586153	1000	778088	11/01/2004	0 00	0 00
16	8590127188365686	4400	250402	01/01/2004	54.91	65 00
17	8590120884250381	7000	503458	07/01/2004	0.00	0.00
18	8590125431232531	1300	503458	07/01/2004	34.76	225.00
19	8590121620630401	8000	284354	03/01/2005	0.00	10.00
20	8590126323841585	9000	925007	04/01/2005	0.00	10 00
21	8590121039256285	10000	284354	11/01/2005	0.00	0.00
22	8590122491455345	9000	503458	07/01/2004	0.00	10.00
23	8590127043771382	6500	250402	01/01/2005	87 52	102.00

fig. 2-11 Table from Excel data

ⓘ *After import, you might have to adjust data types in the table layout.*

Creating Tables from MS Access Data

The following procedure uses the import and copy, automatic layout method to access a table named Customer from a Microsoft Access database named Sample.mdb.

ⓘ *If Access security is enabled, you are prompted to provide a password.*

To create a table from Access data:

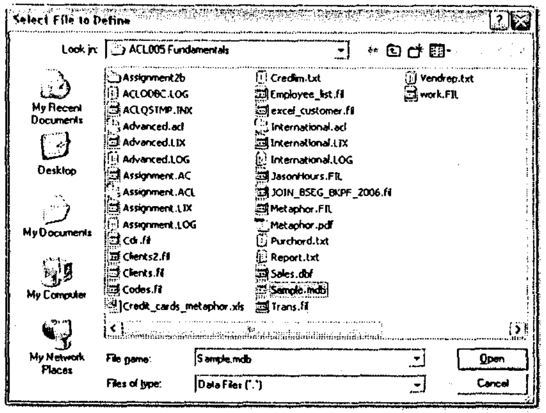
1. Select **File** > **New** > **Table.**

   The Data Definition Wizard appears.

2. Select **Disk** and click **Next.**

   The Select File to Define dialog box appears.

3. Choose `Sample.mdb` and click **Open.**



File name: Sample.mdb

Files of type: Data Files (*.*)

fig. 2-12 Select File to Define dialog box

   The Wizard selects Access database as the file type.

4. Click **Next.**

   ACL prompts you to select the Access Table or View.

5. Select the table `Customer` and click **Next.**

6. Name the file `access_customer` and click **Save.**

   The Wizard displays table information from the Access file.

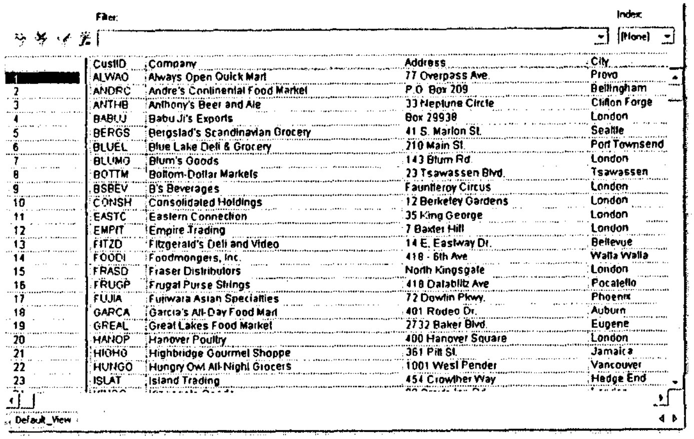
7. Click **Finish.**

8. Accept the table name or enter another and click **OK.**

   The Data Definition Wizard copies the data to an *ACL* data file.

9. Name the table `access_customer` and click **OK.**

   The Data Definition Wizard creates the table.



	CustID	Company	Address	City
1	ALWAO	Always Open Quick Mart	77 Overpass Ave.	Provo
2	ANDRC	Andre's Continental Food Market	P.O. Box 209	Bellingham
3	ANTHB	Anthony's Beer and Ale	33 Neptune Circle	Clifton Forge
4	BABUJ	Babu Ji's Exports	Box 29938	London
5	BERGS	Bergstad's Scandinavian Grocery	41 S. Marion St.	Seattle
6	BLUEL	Blue Lake Deli & Grocery	210 Main St.	Port Townsend
7	BLUMG	Blum's Goods	143 Blum Rd.	London
8	BOTTM	Bottom-Dollar Markets	23 Tsawassen Blvd.	Tsawassen
9	BSBEV	B's Beverages	Fauntleroy Circus	London
10	CONSH	Consolidated Holdings	12 Berkeley Gardens	London
11	EASTC	Eastern Connection	35 King George	London
12	EMPIT	Empire Trading	7 Baxter Hill	London
13	FITZD	Fitzgerald's Deli and Video	14 E. Eastway Dr.	Bellevue
14	FOODI	Foodmongers, Inc.	418 - 6th Ave	Walla Walla
15	FRASD	Fraser Distributors	North Kingsgate	London
16	FRUGP	Frugal Purse Strings	418 Datablitz Ave	Pocatello
17	FUJIA	Fujiwara Asian Specialties	72 Dowlin Pkwy.	Phoenix
18	GARCA	Garcia's All-Day Food Mart	401 Rodeo Dr.	Auburn
19	GREAL	Great Lakes Food Market	2732 Baker Blvd.	Eugene
20	HANOP	Hanover Poultry	400 Hanover Square	London
21	HIGHG	Highbridge Gourmet Shoppe	361 Pitt St.	Jamaica
22	HUNGO	Hungry Owl All-Night Grocers	1001 West Pender	Vancouver
23	ISLAT	Island Trading	454 Crowther Way	Hedge End

fig. 2-13 Table from Access data

Creating Tables from ODBC-Compliant Data

The following procedure uses the import and copy, automatic layout method to access a table named Customer from a Microsoft Access database named Sample.mdb.

ⓘ *You can import from any ODBC-compliant source such as Oracle, DB2, and Excel. The procedure varies with each type of data source.*

Import and copy, automatic layout

The information needed to build the table layout for the *ACL* data file is provided automatically with the data. Data sources in this category include SAP AIS files and data sources accessed by ODBC.

ⓘ *This data access method can be used to build ACL table layouts for Excel or Access files.*

To access data using the import and copy, automatic layout method:

▸ Locate the data source.

- *ACL* copies the data to an *ACL* data file.
- *ACL* creates a table containing the data from the *ACL* file.

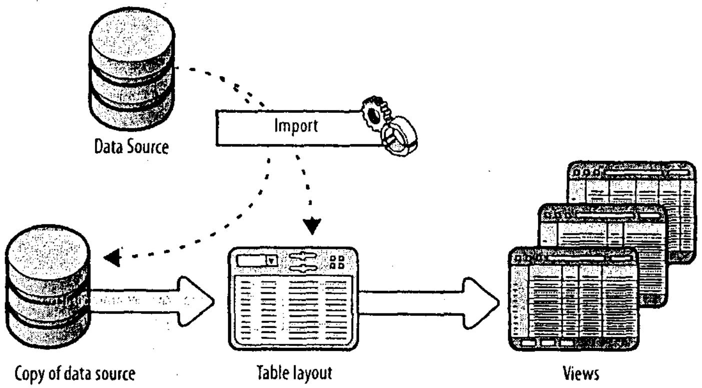


fig. 2-14 Import and copy, automatic layout

ⓘ *Using ODBC, you can refresh imported data from its source at any time.*

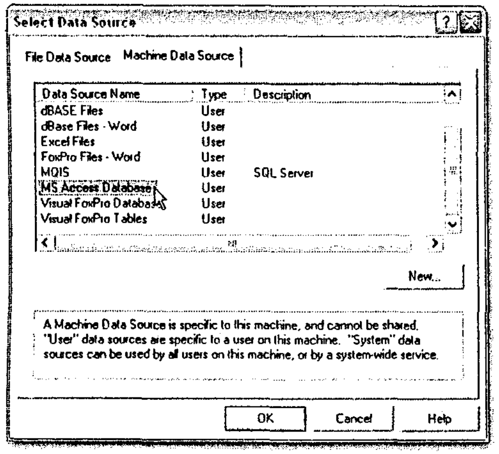
ACL creates a copy of the data source in a flat, fixed-length, *ACL* file with a .fil extension.

In this procedure, do the following:

- Select the Access file
- Select the table or view
- Select the fields and records
- Name and save the table

Select the Access file

1. Select **File** > **New** > **Table** to start the Data Definition Wizard and click **Next.**
2. Select **ODBC** and click **Next.**
3. In the Select Data Source dialog box, click the **Machine Data Source** tab, select MS Access Database, and click **OK.**



Select Data Source

File Data Source | Machine Data Source

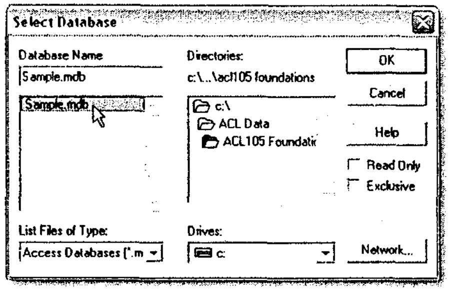
Data Source Name	Type	Description
dBASE Files	User	
dBase Files - Word	User	
Excel Files	User	
FoxPro Files - Word	User	
MQIS	User	SQL Server
MS Access Database	User	
Visual FoxPro Database	User	
Visual FoxPro Tables	User	

A Machine Data Source is specific to this machine, and cannot be shared. "User" data sources are specific to a user on this machine. "System" data sources can be used by all users on this machine, or by a system-wide service.

fig. 2-15 Select Data Source dialog box

ⓘ *Only data sources defined in your ODBC control panel will be listed. For help with adding ODBC data sources, contact your system administrator or the ACL Global Help Desk.*

4. In the Select Database dialog box, navigate to the ACL 105 Foundations folder, select Sample.mdb from the list of available files, and click **OK**.



Select Database

Database Name: Sample.mdb

Sample.mdb

Directories: c:\...\acl105 foundations

c:\
ACL Data
ACL105 Foundatic

☐ Read Only
☐ Exclusive

List Files of Type: Access Databases (*.m

Drives: c:

fig. 2-16 Select Database dialog box

ⓘ *If the selected ODBC data source is mapped to a specific database, the Select Database dialog box does not appear. If not, you are directed from the Select Data Source dialog box to the Select Table dialog box.*

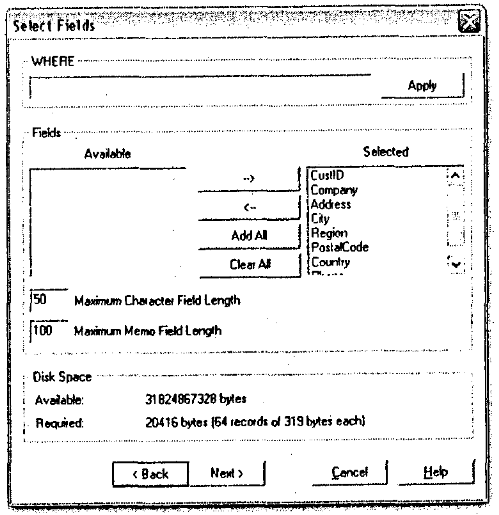
Select the table or view

1. In Select Table, select the Customer table and click **Next**.
2. Name the file `ODBC_customer` and click **Save**.

Select the fields and records

In the Select Fields dialog box, all fields are selected by default.

1. Double-click the fields that you do not want included in the table layout. If necessary, adjust the maximum character and memo field lengths.



Select Fields

WHERE

Fields

Available

Selected: CustID, Company, Address, City, Region, PostalCode, Country

50 Maximum Character Field Length

100 Maximum Memo Field Length

Disk Space

Available: 31824867328 bytes

Required: 20416 bytes (64 records of 319 bytes each)

fig. 2-17 Select Fields dialog box

ⓘ *In Where, you can enter SQL expressions to define the records to include. Leave the box empty to select all records.*

2. Under **Disk Space**, compare **Available** to **Required** to ensure you have sufficient disk space available for the data.

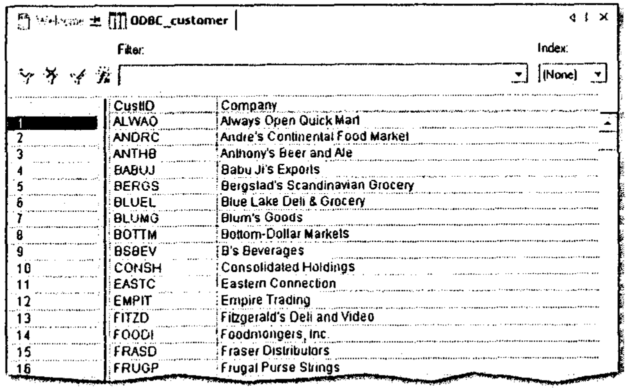
Name and save the table

1. Click **Next**.

   The Data Definition Wizard copies the data to an *ACL* data file.

2. Name the table `ODBC_customer` and click **OK**.

   The Data Definition Wizard creates the table and displays the default view.



	CustID	Company
1	ALWAO	Always Open Quick Mart
2	ANDRC	Andre's Continental Food Market
3	ANTHB	Anthony's Beer and Ale
4	BABUJ	Babu Ji's Exports
5	BERGS	Bergstad's Scandinavian Grocery
6	BLUEL	Blue Lake Deli & Grocery
7	BLUMG	Blum's Goods
8	BOTTM	Bottom-Dollar Markets
9	BSBEV	B's Beverages
10	CONSH	Consolidated Holdings
11	EASTC	Eastern Connection
12	EMPIT	Empire Trading
13	FITZD	Fitzgerald's Deli and Video
14	FOODI	Foodmongers, Inc.
15	FRASD	Fraser Distributors
16	FRUGP	Frugal Purse Strings

fig. 2-18 Table in default view

ⓘ *To quickly auto-resize a column width in the view, double-click the vertical line to the right of the column you want to adjust.*

Refresh your data

When you import a table or view from an ODBC-compliant database, the Data Definition Wizard makes a static copy of the data into an *ACL* data file. When you refresh from the source file, the refreshed data table reflects any subsequent changes in the data and overwrites the data file.

To refresh your data:

- Right-click the table in the Project Navigator, select **Refresh from Source**, and click **Yes**.

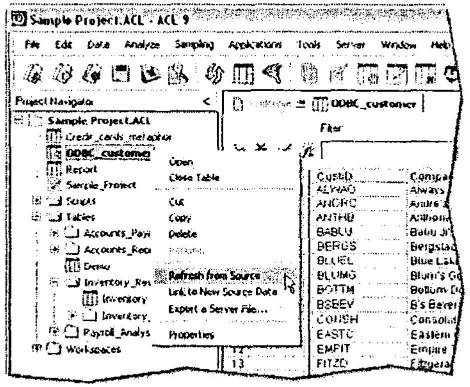


fig. 2-19 Refresh from Source

ⓘ *Prior to refreshing data, back up the original file. This precaution allows you to analyze the original file if need be.*

Creating Tables from dBASE Data

Using the Data Definition Wizard, you can access any data file created by xBASE software such as FoxPro or Clipper.

ⓘ *Memo and index files from xBASE software are not supported.*

Direct access, automatic layout

The information needed to build the table layout is provided automatically with the data. On the PC platform, *ACL* can access dBASE and ACCPAC files. On the OS/390 platform, the *ACL* Server Edition can access DB2, IMS, and VSAM.

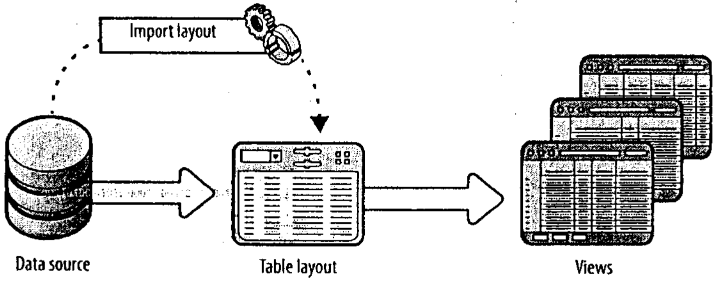


fig. 2-20 Direct access, automatic layout

To access data using this method:

▸ Locate the data source.

ACL creates a table containing data read directly from the data source.

The following procedure uses the direct access, automatic layout method to access a dBASE file named Arec.dbf.

In this procedure, do the following:

- Select the dBASE file
- Name and save the table

Notes:

Select the dBASE file

1. Select **File** > **New** > **Table** to start the Data Definition Wizard.
2. Click **Next**.

   By default, the Data Definition Wizard selects Disk as the data source.
3. Click **Next**.
4. In the Select File to Define dialog box, select `Arec.dbf` and click **Open**.

   The Data Definition Wizard correctly identifies the character set as ASCII.
5. Click **Next**.

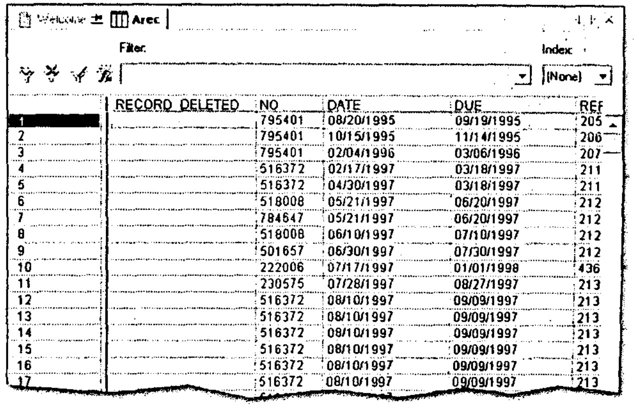
   The Data Definition Wizard correctly identifies the file as a dBASE-compatible file.
6. Click **Next**.

   The Data Definition Wizard shows you a summary of the table's record and field properties.

Name and save the table

1. Click **Finish**.
2. Name the table `Arec` and click **OK**.

   The Data Definition Wizard creates the table and displays the default view.



	RECORD_DELETED	NO	DATE	DUE	REF
1		795401	08/20/1995	09/19/1995	205
2		795401	10/15/1995	11/14/1995	206
3		795401	02/04/1996	03/06/1996	207
4		516372	02/17/1997	03/18/1997	211
5		516372	04/30/1997	03/18/1997	211
6		518008	05/21/1997	06/20/1997	212
7		784647	05/21/1997	06/20/1997	212
8		518008	06/10/1997	07/10/1997	212
9		501657	06/30/1997	07/30/1997	212
10		222006	07/17/1997	01/01/1998	436
11		230575	07/28/1997	08/27/1997	213
12		516372	08/10/1997	09/09/1997	213
13		516372	08/10/1997	09/09/1997	213
14		516372	08/10/1997	09/09/1997	213
15		516372	08/10/1997	09/09/1997	213
16		516372	08/10/1997	09/09/1997	213
17		516372	08/10/1997	09/09/1997	213

fig. 2-21 Table in default view

ⓘ *The first field in a dBASE file is record_deleted. If any record has an "*" in this field it means that the record has been marked for deletion in the source file. ACL does not automatically remove or ignore records that have been marked for deletion.*
To check for the occurrence of records marked for deletion, use the filter: `record_deleted = "*"`

Creating Tables from Report Files

The Data Definition Wizard can import data from report files. Often referred to as print image or print spool files, these are electronic copies of printed reports.

Report files often contain multiple record types, sometimes spanning more than one line. The most common record types are detail and header records, although some report files also include footer records. Each record type may contain information you want included in the table you create. If so, you may have to identify these record types for the Data Definition Wizard.

Import and copy, manual layout

Information required to build the table layout is in supplementary documentation. Use that documentation to help *ACL* build the table layout. Data sources in this category include delimited and report files.

To access data using this method:

1. Locate the data source.
2. Provide table layout information or, when *ACL* can determine the table layout, you are given the opportunity to confirm it.
   - *ACL* copies the data to an *ACL* data file.
   - *ACL* creates a table containing the data from the *ACL* file.

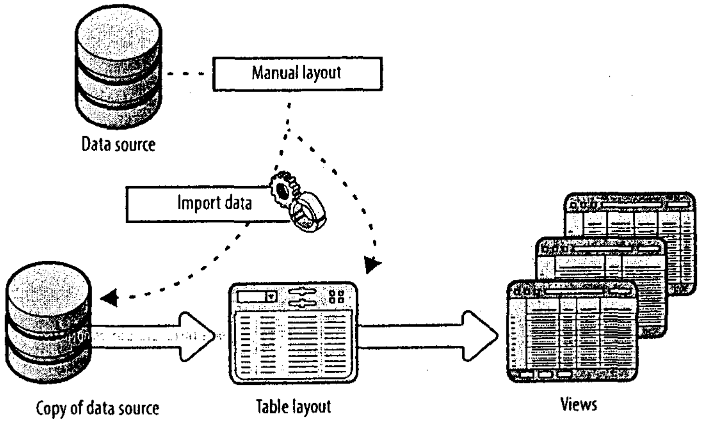


fig. 2-22 Import and copy, manual layout

The following procedure uses the Import and copy, manual layout method to access data from a report file named Report.txt. This file contains only detail and header records.

In the following exercise, you will use the Data Definition Wizard to create a table from a report file. To do this, you will complete the following main tasks:

- Select platform
- Identify detail information
- Identify header information
- Name the fields and select data types
- Name and save the table

To create a table from a report file:

Select platform

1. Select **File** > **New** > **Table**

   The Data Definition Wizard starts. By default, the Data Definition Wizard selects Disk as the data source.

2. Click **Next**.
3. In Select File to Define, select Report.txt and click **Open**.

   The Data Definition Wizard correctly identifies the character set as ASCII.

4. Click **Next**.

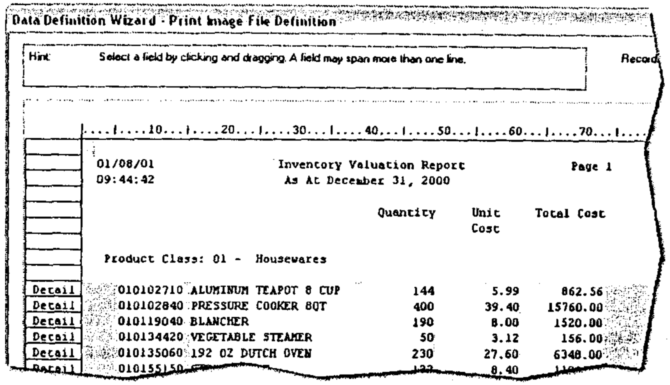
   The Data Definition Wizard correctly identifies the file as a Print Image (Report) file.

Identify detail information

▸ Click **Next**.

The Print Image File Definition window displays the report file.

The Data Definition Wizard attempts to identify detail records from patterns in the data. If it is unable to do so, you must identify the detail records.


Data Definition Wizard - Print Image File Definition
Hint: Select a field by clicking and dragging. A field may span more than one line.
Record

```
....|....10...|....20...|....30...|....40...|....50...|....60...|....70...|....

        01/08/01                   Inventory Valuation Report              Page 1
        09:44:42                    As At December 31, 2000

                                                       Quantity   Unit    Total Cost
                                                                  Cost

         Product Class: 01 -  Housewares

Detail    010102710 ALUMINUM TEAPOT 8 CUP                  144     5.99      862.56
Detail    010102840 PRESSURE COOKER 8QT                    400    39.40    15760.00
Detail    010119040 BLANCHER                               190     8.00     1520.00
Detail    010134420 VEGETABLE STEAMER                       50     3.12      156.00
Detail    010135060 192 OZ DUTCH OVEN                      230    27.60     6348.00
Detail    010155150                                                8.40
```

fig. 2-23 Data Definition Wizard - Print Image File Definition

In *fig. 2-23*, the Data Definition Wizard identifies:

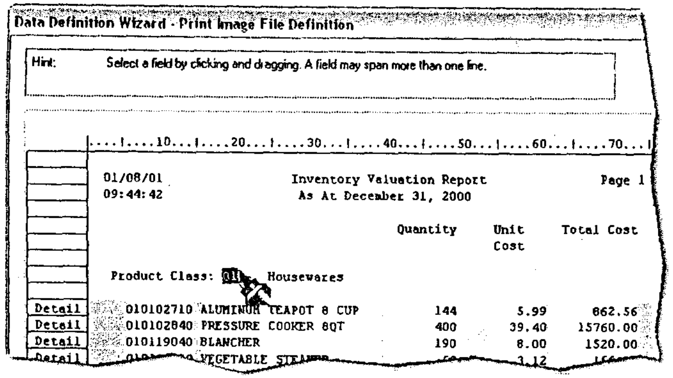
- Each detail record
- The start and end position of each field in the detail rows
- The record length

ⓘ *The Data Definition Wizard uses the decimal in the unit cost field to identify detail records.*

Each group of detail records is preceded by a header record containing a product class number and a class description.

Identify header information

1. On Print Image File Definition (*fig. 2-24*), define the first field in the header record (product class number) by highlighting the first instance, 01.

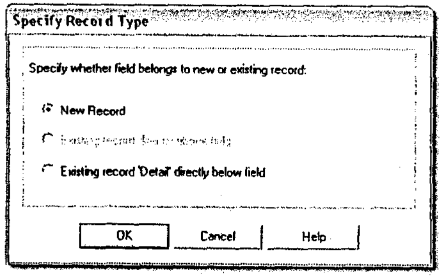


Hint: Select a field by clicking and dragging. A field may span more than one line.

```
|....|....10...|....20...|....30...|....40...|....50...|....60...|....70...|
        01/08/01                  Inventory Valuation Report           Page 1
        09:44:42                   As At December 31, 2000

                                                Quantity    Unit    Total Cost
                                                            Cost

         Product Class: 01 - Housewares

Detail    010102710 ALUMINUM TEAPOT 8 CUP           144      5.99      862.56
Detail    010102840 PRESSURE COOKER 8QT             400     39.40    15760.00
Detail    010119040 BLANCHER                        190      8.00     1520.00
```

fig. 2-24 Data Definition Wizard - Print Image File Definition

Specify Record Type (*fig. 2-25*) correctly shows that the product class number is part of a new record type.



Specify whether field belongs to new or existing record:

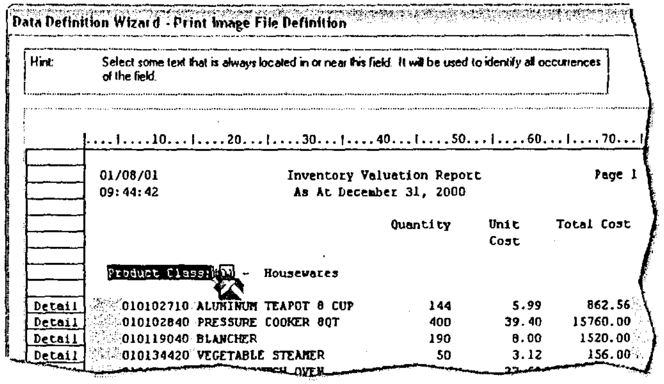
● New Record

○ Existing record [illegible]

○ Existing record 'Detail' directly below field

fig. 2-25 Specify Record Type dialog box

2. Click **OK**.
3. In Field Definition, name the new field `prodcls` and click **OK**.
4. To define the header record type as the data to identify all instances of a header record in this file, highlight `Product Class:`.



Hint: Select some text that is always located in or near this field. It will be used to identify all occurrences of the field.

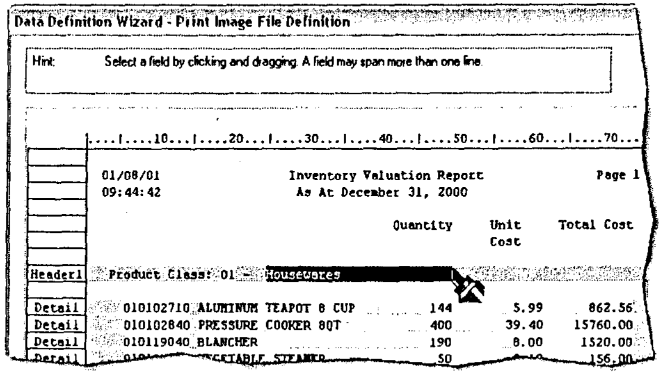
```
|....|....10...|....20...|....30...|....40...|....50...|....60...|....70...|
        01/08/01                  Inventory Valuation Report           Page 1
        09:44:42                   As At December 31, 2000

                                                Quantity    Unit    Total Cost
                                                            Cost

        Product Class: 01 - Housewares

Detail    010102710 ALUMINUM TEAPOT 8 CUP           144      5.99      862.56
Detail    010102840 PRESSURE COOKER 8QT             400     39.40    15760.00
Detail    010119040 BLANCHER                        190      8.00     1520.00
Detail    010134420 VEGETABLE STEAMER                50      3.12      156.00
```

fig. 2-26 Data Definition Wizard - Print Image File Definition

5. Click **OK**.

   In *fig. 2-26*, the header is identified as Header1. The product class number is also highlighted.
6. Identify other fields in the header record that you want to include in the detail records.
7. Highlight the product class description, `Housewares`.

   Ensure you select enough of a range to accommodate the longest product class description in the report.



Hint: Select a field by clicking and dragging. A field may span more than one line.

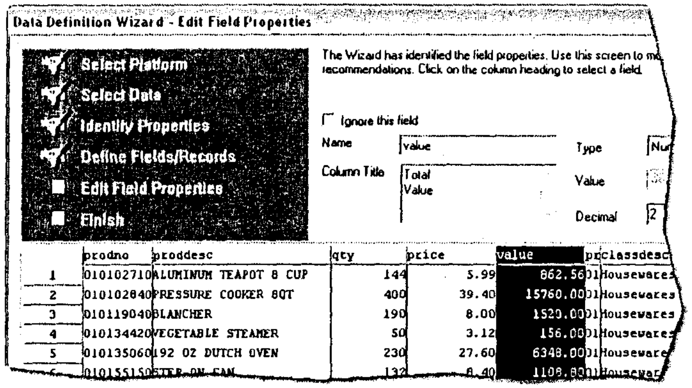
```
|....|....10...|....20...|....30...|....40...|....50...|....60...|....70...
        01/08/01                  Inventory Valuation Report           Page 1
        09:44:42                   As At December 31, 2000

                                                Quantity    Unit    Total Cost
                                                            Cost

Header1   Product Class: 01 - Housewares

Detail    010102710 ALUMINUM TEAPOT 8 CUP           144      5.99      862.56
Detail    010102840 PRESSURE COOKER 8QT             400     39.40    15760.00
Detail    010119040 BLANCHER                        190      8.00     1520.00
Detail              VEGETABLE STEAMER                50               156.00
```

fig. 2-27 Data Definition Wizard - Print Image File Definition

8. In Field Definition, name the field `class_desc` and click **OK.**
9. Click **Next.**

Name the fields and select data types

1. In Save Data File As, name the file `Report` and click **Save.**
2. On Edit Field Properties, click a column heading to select a field and:
    a. Enter a name to identify the field in the table layout.
    b. Enter a column heading to identify the field in the view.
    c. From the **Type** drop-down list, select the data type and adjust the format if necessary.
3. For all fields you want to define, repeat the previous step.



The Wizard has identified the field properties. Use this screen to m
recommendations. Click on the column heading to select a field.

□ Ignore this field

Name: value

Column Title: Total Value

Type: Nu

Decimal: 2

	prodno	proddesc	qty	price	value	pr	classdesc
1	010102710	ALUMINUM TEAPOT 8 CUP	144	5.99	862.56	01	Housewares
2	010102840	PRESSURE COOKER 8QT	400	39.40	15760.00	01	Housewares
3	010119040	BLANCHER	190	8.00	1520.00	01	Housewares
4	010134420	VEGETABLE STEAMER	50	3.12	156.00	01	Housewares
5	010135060	192 OZ DUTCH OVEN	230	27.60	6348.00	01	Housewares
6	010155150	STEP-ON CAN	132	8.40	1108.80	01	Housewar

fig. 2-28 Data Definition Wizard - Edit Field Properties

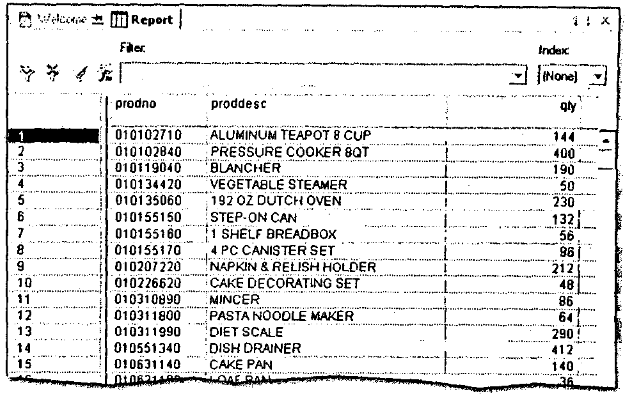
4. Click **Next.**

    The Data Definition Wizard copies the data to an *ACL* data file and shows you a summary of table record and field properties.

Name and save the table

1. Click **Finish.**
2. Name the table `Report` and click **OK.**

    The Data Definition Wizard creates the table and displays the default view.



	prodno	proddesc	qty
1	010102710	ALUMINUM TEAPOT 8 CUP	144
2	010102840	PRESSURE COOKER 8QT	400
3	010119040	BLANCHER	190
4	010134420	VEGETABLE STEAMER	50
5	010135060	192 OZ DUTCH OVEN	230
6	010155150	STEP-ON CAN	132
7	010155160	1 SHELF BREADBOX	56
8	010155170	4 PC CANISTER SET	96
9	010207220	NAPKIN & RELISH HOLDER	212
10	010226620	CAKE DECORATING SET	48
11	010310890	MINCER	86
12	010311800	PASTA NOODLE MAKER	64
13	010311990	DIET SCALE	290
14	010551340	DISH DRAINER	412
15	010631140	CAKE PAN	140

fig. 2-29 Example table in default view

Creating Tables from Adobe PDF Data

You can create tables for Adobe PDF (Portable Document Format) files. The procedure uses the import and copy, manual layout method to access PDF files.

PDF files can contain images and text that appear in paragraphs, tables, headers, footers, and other areas. *ACL* extracts text data only.

PDF files retain spacing and font information from the original application (for example, MS Word). Extra spaces or inconsistent fonts may cause data to be misaligned during the file definition. If possible, ensure that spaces and fonts in the original file are consistent before creating PDF files.

ⓘ *ACL is able to read files created with Acrobat 5 and later.*

PDF files can be password protected requiring you to enter a password to open or to edit and print a file. Regardless of the type of security applied, you are required to enter a password. PDF files with Certificate Security (public key encryption) cannot be accessed by *ACL*.

This procedure uses the following sample file: Invoice_Number_Report.PDF

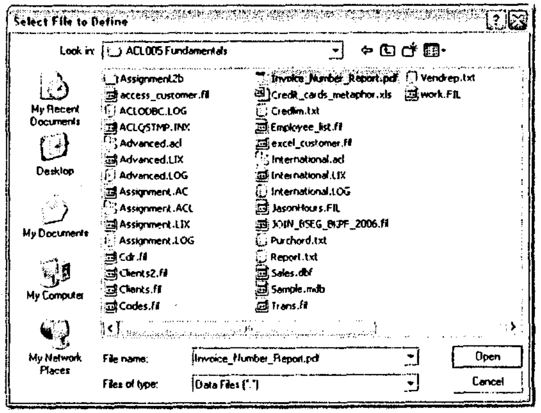
To create a table for a PDF data:

1. Select **File** > **New** > **Table.**

   The Data Definition Wizard starts. By default, the Data Definition Wizard selects Disk as the data source.

2. Click **Next**.

   The Select File to Define dialog box appears.



Look in: ACL005 Fundamentals

File name: Invoice_Number_Report.pdf

Files of type: Data Files (*.*)

fig. 2-30 Select File to Define dialog box

3. Choose a PDF file and click **Open**.

   ACL analyzes the file and lists its format as PDF Adobe Acrobat file.

4. Click **Next**.

5. If the PDF file is password-protected, enter the password in the text box and click **Next**.

6. Enter a page range or select All and click **Next**.

7. Complete the steps in “Import and copy, manual layout” on page 28, select regions and assign them to fields.

Reusing Table Layouts

You can reuse table layouts you created in other *ACL* projects or again in the same project. You can copy table layouts, rename them, and link them to new data sources. You can also import table layouts from other *ACL* projects or export table layouts to *ACL* Server Edition.

ⓘ *Reusing table layouts only works when the source data are flat files.*

Copying and Renaming Table Layouts

When you regularly receive data with the same structure, you can copy existing table layouts to use with new data. This way, you need to define files with the same structure only once.

To copy and rename a table layout:

1. In the **Overview** tab, right-click a table and select **Copy**.
2. Right-click the destination folder and select **Paste**.

   ACL places the copy in the specified folder. The original table name is appended with a 2.
3. Right-click the duplicated table layout, select **Rename**, enter a new name, and click **OK**.

Linking Table Layouts to New Data Sources

1. Right-click a table and select **Link to New Source Data**.
2. In the Select File dialog box, select the new data source and click **Open**.

   Your table layout is linked to the new data source.

Importing Table Layouts from Other ACL Projects

1. Right-click the project name and select **Copy from another Project > Table**.
2. Locate and select the project from which you want to import a table layout and click **Open**.
3. Double-click the table layout(s) that you want to import and click **OK**.

Exporting Table Layouts

You can export a table layout by creating a format (.fmt) file.

To export a table layout:

1. Right-click a table and select **Export a Server File**.
2. Select the platform to which you want to export the table layout.
3. Name the table layout file and click **Save**.

Expressions

3

About This Module

Understanding Expressions

Creating Filters

Creating Computed Fields

This module describes how to use expressions in filters and computed fields. Expressions are lines of code that return values. Using expressions, you can ask questions of the data to analyze and expand on it. Expressions are especially helpful when data needs to be normalized. This module shows you how to create different types of expressions.

Understanding Expressions

Expressions are sets of operators and values used to perform calculations, specify logical conditions, or create values that do not exist directly in the data. *ACL* expressions can be a combination of data fields, operators, constants, functions, and variables.

In *fig. 3-1*, expressions are divided into Filter and Computed Field. From there, you can choose sub-types.

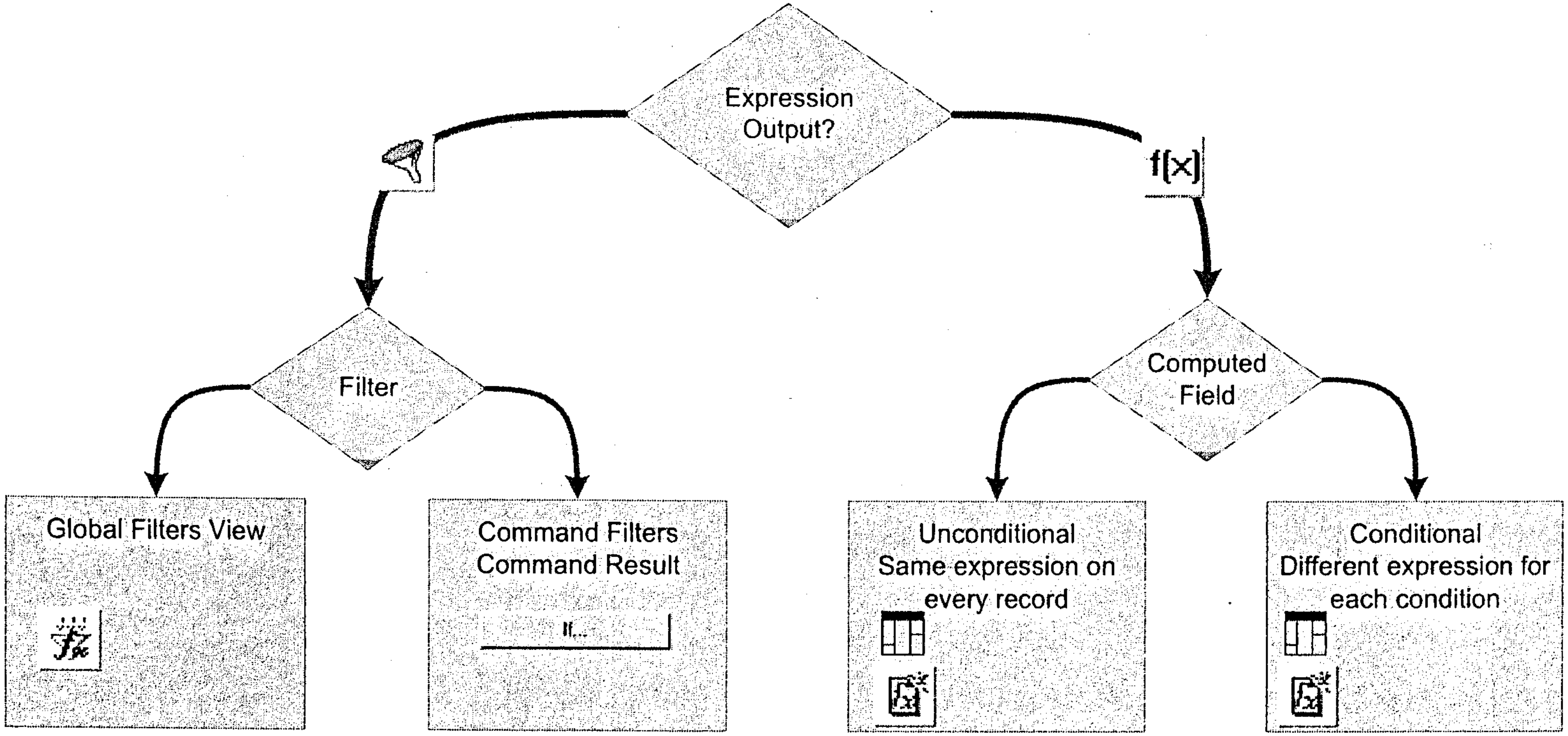


fig. 3-1 Expression decision tree

ⓘ *ACL Help provides more information on expressions. In ACL, select Help > Index and enter "expressions".*

Creating Filters

Expressions are used primarily to create filters or computed fields. In *ACL*, there are four types of expressions: logical, numeric, date, and character. Filters are logical expressions that let you select and work with particular data types.

Filter Elements

Filters consist of fields, operators, and values.

Fields

Use fields from your table to build filters. The fields are listed in, and selected from, the Expression Builder's Available Fields list.

Operators

Operators are used to combine simple expressions into more complex expression.

ACL uses the following operators:

Operator	Description
=	equal to
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
<>	not equal to

Values

Values contain the data you want to filter. Most data types must be enclosed with a delimiter. Depending on the data types that you are filtering, values must use different delimiters. The list below shows three data types, their appropriate delimiters, and an example of how to use them in a filter:

Data Type	Delimiter	Example
Character	Single or double quotes	"LATEX"
Date values	Reverse quotes	`20060930`
Numeric values	none required	25

Filter Examples

In the examples below, each field is followed by an operator and a value. Each value uses its appropriate field type.

Field	Operator	Value	Data Type
prod_desc	=	"LATEX"	Character
location	=	"04"	Character
quantity	>	25	Numeric
price_date	<=	`20060930`	Date

Creating Filters

There are two types of filters:

Global	Global filters restrict records that can be viewed and analyzed. When set, global filters remain active until removed. ▪ You can use Quick Filter to quickly create a global filter. To use this option: in the view, select one or more values, right-click, select Quick Filter, and select the type of logical operator to use in your filter.
Command	Command filters are applied to individual commands causing them to be executed only on the records that meet the filter criteria. When the command completes its processing, command filters are no longer active.

Global filters

Only one global filter can be in effect at a time. However, a global filter can consist of more than one existing filter and And/Or operators. *ACL* applies the conditions defined in the selected filter and displays the name of the global filter in effect on the status bar.

To create a global filter:

1. In the active table, click the (Edit View Filter) button.
2. In the **Expression** box, enter a logical expression and click **OK**.

ⓘ *To count the filtered records, run the Count command. ACL does not automatically count the records that meet the filter criteria.*

Command filters

A command filter is a condition applied to a command at the time you execute the command. You can make commands conditional, (dependant on a true or false condition) by specifying a filter in the command dialog box.

To create a command filter:

1. Select the command and click the **If** button.
2. In the **Expression** box, enter a logical expression and click **OK**.
3. Click **OK** to execute the command.

Naming and Saving Filters

When named, filters are saved in the table layout as logical computed fields. As with non-computed fields, you can edit and use logical computed fields.

You can name and save filters in the table layout for future use or reference.

To name and save a filter:

1. In the **Expression** box, enter a name in the **Save As** box.
2. Click **OK**.

   The filter is named and saved.

Creating Computed Fields

Computed fields are named expressions that calculate results to create additional data. Computed fields exists only in the table and are not actually a part of the data source. They are *virtual* fields that let you perform calculations based on information in the data file without affecting or changing the original data. You can use computed fields like *physical* fields.

Computed fields can return character, numeric, date, or logical values, and can be conditional or unconditional.

You create computed fields on the **Edit Fields/Expressions** tab of the Table Layout window. When you create a computed field, it is added to the table layout, but not to the original data file.

ⓘ *ACL does not automatically add new fields to the view. To add a field, right-click in the table, select Add Columns, select the field(s), and click* **OK**.

There are four main uses for computed fields:

- Performing unconditional computations
- Converting fields from one data type to another
- Making word substitutions
- Creating true/false logical tests (filters)

Unconditional Computed Fields

An unconditional expression is one that is applied to every record in the file.

To create an unconditional computed field:

1. Select **Edit** > **Table Layout** and click the (Add a New Expression) button.
2. In the **Name** box, enter a name for the computed field.
3. Click the f(x) button to open the Expression Builder.
4. In the **Expression** box, enter a numeric expression and click **OK**. For example, to calculate the extent to which people are over their credit limits, the expression might be:
   ```
   balance - limit
   ```
5. Click the ✓ (Accept Entry) button.

   ⓘ *ACL uses fixed-point arithmetic to increase processing efficiency. With fixed-point arithmetic, ACL may round decimals unexpectedly. For more information, select Help > Index and enter "fixed-point arithmetic".*

In "Using Other Data Integrity Tests" on page 56, computed fields are used to recalculate numeric fields.

Conversion Computed Fields

During your analysis, you may find fields that you want to use for a certain purpose, but find that those fields are a different data type than what is required for the command. In *ACL*, there are many functions that allow you to convert data from one type to another.

To Convert From	To	Use
Character	Numeric	VALUE( )
Character	Date	CTOD( )
Numeric	Date	CTOP( )
Numeric	Character	STRING( )
Date	Character	DATE( )

ⓘ *For more information, select Help > Index and enter "functions".*

To create a conversion computed field:

1. Select **Edit** > **Table Layout** and click the (Add a New Expression) button.
2. In the **Name** box, enter a name for the computed field.
3. Click the f(x) button to open the Expression Builder.
4. Browse the **Functions** list for the conversion function you want to use, double-click, and edit the function parameters in the **Expression** box.

   For example, to convert an inventory value field defined as numeric into a character field with a length of 12, your expression might be:

   ```
   STRING(value,12)
   ```
5. Click **OK** and click ✓ (Accept Entry).

Conditional Computed Fields

ACL populates a conditional computed field based on conditions and values that you specify. For example, you can create a computed field that contains:

- "Vancouver" for location codes 01 and 02
- "New York" for location codes 03 and 04
- "Chicago" for location codes 05 and 06
- "Unknown" for all other location codes

To create a conditional computed field:

1. Select **Edit** > **Table Layout** and click (Add a New Expression).
2. In the **Name** box, enter a name for the computed field.
3. In the **Default Value** box, if none of the conditions are met, enter a value to be used.

   For example, enter `"Unknown"`.
4. Click (Insert a Condition) and enter a condition and a value.

   For example, enter the condition `location = "01" OR location = "02"` and the value `"Vancouver"`.

   You can enter as many condition and value pairs as needed.
5. Click ✓ (Accept Entry).

   ⓘ *Multiple condition and value pairs are processed in the order that they are listed from top to bottom.*

Notes:

4

Data Integrity Verification

About This Module

- Understanding Data Integrity
- Checking Validity
- Confirming Control Totals
- Checking for Correct Bounds
- Finding Missing Items
- Finding Duplicates
- Testing for Reliability
- Using Other Data Integrity Tests

As a data analyst, it is important that you maintain an attitude of professional skepticism toward the data you receive. To have confidence that your conclusions are valid, you need to verify your data's integrity.

Understanding Data Integrity

Data integrity means that your table contains:

- All the data and only the data that you requested
- Calculated fields that deliver correct values
- Only numeric data in numeric fields
- Only unique records
- Only valid dates
- Data in fields where it is expected
- Field relations that are consistent and logical

If your table does not meet these standards, your analysis could be based on incorrect premises, possibly rendering your findings and recommendations invalid. It is necessary, therefore, to build your analysis upon data that has been verified by the processes outlined in this module.

Sources of Error

The integrity of your data can be compromised inadvertently (or with fraudulent intent) at any point in the data's life cycle.

Design

The data source may have undetected flaws in field definitions or expressions, resulting in unreliable data. For example, an incorrect expression for interest calculation generates incorrect interest expense values. Processing problems can also result in missing, duplicate, or invalid data.

Input

A data entry clerk might enter incorrect data, omit items, include unwanted items, or enter invalid data such as non-existent dates. If the system does not have input validation controls, these errors remain in the data.

Extraction

When receiving a data request, a programmer or business analyst responsible for extracting data may misinterpret the parameters, resulting in missing or extra records being included. For example, a request for all transactions between January 1, 2002, and January 31, 2002, might be interpreted as exclusive of the beginning and end dates instead of inclusive.

Conversion

The character set for data on an IBM mainframe system is EBCDIC, which many software applications cannot read. Programmers often convert data into ASCII, which in the case of EBCDIC, can create errors in some data types, especially packed fields.

ⓘ *ACL for Windows can read and process EBCDIC, so conversion is unnecessary.*

Transmission

The mode of transmission, whether by network or other media, may corrupt the data.

Definition

The table layout created in *ACL* to accept the data source may not be correctly defined, resulting in mismatches between fields and data.

Testing Data Integrity

ACL provides many tools for testing data integrity. Tools include functions and commands.

To test the integrity of your data:

1. Examine the table layout for validity by checking data types and field definitions.
2. Check that all requested records are present, there are no unrequested records, and all numeric field totals are correct.
3. Check that data falls within expected bounds.
4. Perform other tests as required using the following:

Test	Tool	Ensure
Validity	Verify	Data types and field definitions match.
Control Totals	Count	Number of records matches control totals.
	Total	Numeric field totals match control totals.
	Statistics	Record and field totals match control totals.
Bounds Values	Statistics	Data falls within correct bounds.
	BETWEEN( )	Data falls within correct bounds.
	Classify	Data falls within correct bounds.
Missing Items	Gaps	Values are not missing from sequences.
	ISBLANK( )	Fields do not contain inappropriate blanks.
Duplicates	Duplicates	Tables do not contain duplicate records or inappropriate duplicate values.
Reliability	Computed fields	Calculations in the data source are correct.

These tests and tools are explained in the sections that follow.

Notes:

Checking Validity

Checking data integrity ensures that tables are valid. Tables are valid if the data is correct and the field definitions conform to the data. Checking validity is especially important when working with data that was defined without the Data Definition Wizard.

Using the Verify Command

The Verify command can identify data source, transmission, and definition errors by checking that:

- Character fields contain only valid printable characters (letters, numbers, symbols, and international characters)
- Numeric fields contain only valid numeric characters (numbers, decimals, and currency symbols)
- Date fields contain only valid dates

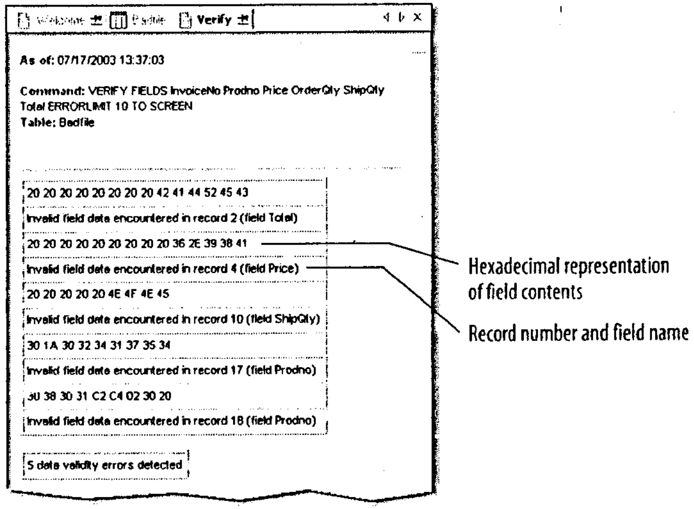
To find fields with validity errors, include all fields when running the Verify command.

To run the Verify command:

1. Select **Data** > **Verify.**
2. Click **Verify Fields,** add all the fields to the **Selected Fields** list, and click **OK.**
3. Click **OK.**

ⓘ *ACL does not verify computed fields—it assumes that the data in these fields is valid.*

The results shown below are produced by running the Verify command on all the fields in the Badfile table.


Welcome
Badfile
Verify
Hexadecimal representation of field contents
Record number and field name

```
As of: 07/17/2003 13:37:03

Command: VERIFY FIELDS InvoiceNo Prodno Price OrderQty ShipQty
Total ERRORLIMIT 10 TO SCREEN
Table: Badfile

20 20 20 20 20 20 20 20 42 41 44 52 45 43
Invalid field data encountered in record 2 (field Total)
20 20 20 20 20 20 20 20 20 36 2E 39 38 41
Invalid field data encountered in record 4 (field Price)
20 20 20 20 20 4E 4F 4E 45
Invalid field data encountered in record 10 (field ShipQty)
30 1A 30 32 34 31 37 35 34
Invalid field data encountered in record 17 (field Prodno)
30 38 30 31 C2 C4 02 30 20
Invalid field data encountered in record 18 (field Prodno)

5 data validity errors detected
```

fig. 4-1 Verify command results

For each of the five errors detected, *ACL* displays two items: the hexadecimal representation of the field contents and the record number and field name where the error was found.

What if you find validity errors in your table?

You need to address validity errors before proceeding with your analysis. If the errors are concentrated in particular fields, the problem may lie either in the definition of the fields in the *ACL* table layout or in the data source. Examine the *ACL* layout first. If the layout is correct, contact the data provider, as there may be errors in the data source.

If the errors appear random, the problem may be input or transmission errors. If a field is critical to your analysis, try to fix the errors or, if that is not possible, have the data re-sent. In either case, notify the data provider.

To check the table layout:

Date fields	Ensure that the date format specified in the field definition is correct. If you have data that reads 12312002 and you have specified the format as DDMMYYYY, *ACL* identifies this as a validity error. Also, check that all dates are valid calendar dates.
Numeric fields	Ensure that numeric fields contain only numbers, decimals, and currency symbols.
Character fields	Ensure that ASCII and EBCDIC characters are stored in the field.

You must use the hexadecimal output included in the Verify command results to determine the problem. Validity errors arise from unprintable characters that are not visible in the data.

What if ACL reports the error limit is exceeded?

In *ACL*, any command that reports errors has an error limit setting. *ACL* processes a table until the error limit is reached. The default value for the error limit is 10. You can change the error limit on the More tab in the Verify dialog box.

Notes:

Confirming Control Totals

After you confirm the validity of table layout and data, you can check that data meets the specifications in the summary report. Compare the control totals generated in *ACL* against those provided in the summary report.

If you do not confirm control totals, you may be working with a table that has more or fewer records than you requested or that contains corrupt data.

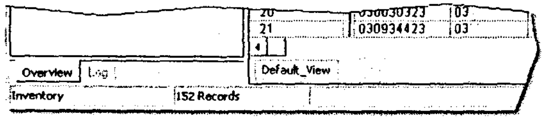
Three commands in *ACL* generate control totals: Count, Total, and Statistics.

The Count Command

Use the Count command to count records. Unlike other commands, the results of the Count command are displayed in the status bar. The only time the results of the Count command are displayed in a results tab is when running the command with a command filter.

To run the Count command:

- Select **Analyze** > **Count Records** and click **OK**.



```
20        030030323   03
21        030934423   03
```

Inventory 152 Records

fig. 4-2 Count command results

In *fig. 4-2*, results are produced by running the Count command on the Inventory table.

ⓘ *The results of the Count command are recorded in the log.*

The Total Command

You can use the Total command to sum numeric fields in your data. Use the Total command to generate control totals.

To run the Total command:

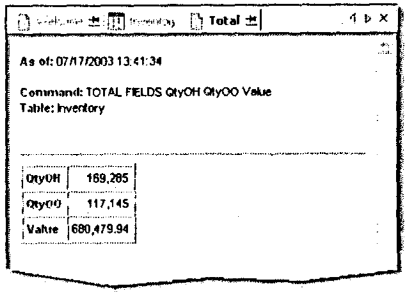
1. Select **Analyze** > **Total Fields**.

   The Total dialog box opens.

2. Click **Total Fields....**

   Add available fields to the **Selected Fields** list, and click **OK**.

3. To see your results, click **OK**.



```
As of: 07/17/2003 13:41:34

Command: TOTAL FIELDS QtyOH QtyOO Value
Table: Inventory

QtyOH      169,285
QtyOO      117,145
Value   680,479.94
```

fig. 4-3 Total command

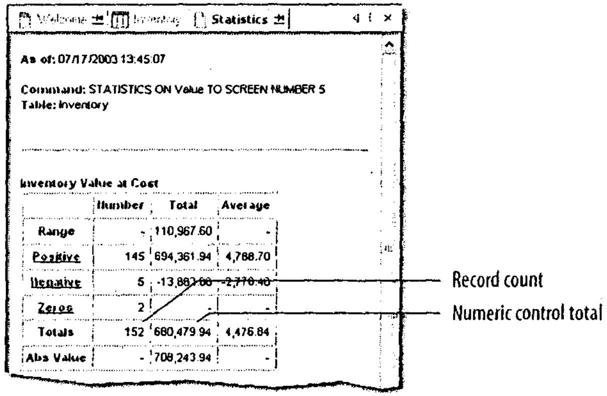
In *fig. 4-3*, results are produced by running the Total command on the quantity on hand, quantity on order, and inventory value fields in the Inventory table.

The Statistics Command

You can use the Statistics command to generate control totals. However, unlike the Total command, the Statistics command also works with date fields.

To run the Statistics command:

1. Select **Analyze** > **Statistical** > **Statistics.**
2. Click **Statistics On**, add the fields to the **Selected Fields** list for the fields you want control totals, and click **OK.**
3. Click **OK.**



As of: 07/17/2003 13:45:07

Command: STATISTICS ON Value TO SCREEN NUMBER 5
Table: Inventory

Inventory Value at Cost

	Number	Total	Average
Range	-	110,967.60	-
Positive	145	694,361.94	4,788.70
Negative	5	-13,882.00	-2,776.40
Zeros	2	-	-
Totals	152	680,479.94	4,476.84
Abs Value	-	708,243.94	-

fig. 4-4 Statistics command

In *fig. 4-4*, results are produced by running the Statistics command on the value field in the Inventory table. The Totals row contains the values for the record count (in the Number column) and the numeric control total (in the Total column).

What If Control Totals Do Not Match?

When control totals do not match your request or the totals in the summary report, it usually means the requested data was not extracted properly from the data source.

If you find that you have too many records, use a filter to extract the required data. If you find you have too few records or that field totals do not match the summary report, contact the data provider and have the source data file re-sent.

Notes:

Checking for Correct Bounds

Having confirmed that your table layout is valid and control totals match independent sources, you can check that the upper and lower bounds match your request.

For example:

- In a table with debit transactions only, the lower bound is greater than or equal to 0.00 and the upper bound is a positive number.
- In a table for the calendar year 2002, the lower bound is not earlier than January 1, 2002 and the upper bound is no later than December 31, 2002.

You can use the Statistics command and the BETWEEN( ) function to test if data meets your expectations.

The Statistics Command

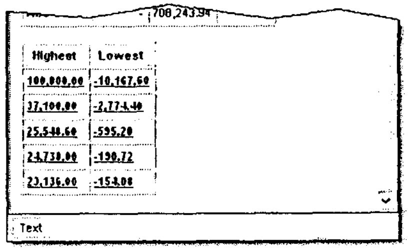
Besides control totals, the Statistics command also generates the bounds values of a field for both numeric and date data types. These values can be checked against known values to ensure that the data is correct.

Using the Statistics command to check for numeric bounds

Use the highest value and lowest value in the results to determine if the records in your table are within expected upper and lower bounds.

To run the Statistics command to check for numeric bounds:

1. Select **Analyze** > **Statistical** > **Statistics.**
2. Click **Statistics On**, add the numeric or date fields to the **Selected Fields** list for the fields you want bounds information, and click **OK.**
3. Click **OK.**



708,243.94

Highest	Lowest
100,000.00	-10,167.60
37,100.00	-2,774.40
25,540.60	-595.20
24,730.00	-190.72
23,136.00	-154.08

fig. 4-5 Statistics command results

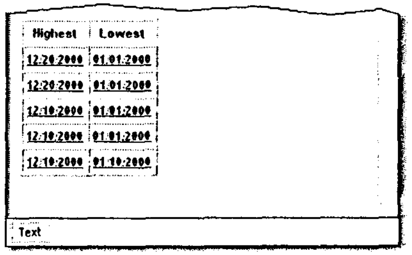
In *fig. 4-5*, results are produced by running the Statistics command on the value field in the Inventory table.

Use the Statistics command to check that dates fall within bounds

Use the highest value and lowest value in the results to determine if the records in your table are within specified bounds.

To run the Statistics command to check dates fall within bounds:

1. Select **Analyze** > **Statistical** > **Statistics.**
2. From the **Statistics On** list, select a date field, and click **OK.**



Highest	Lowest
12:20:2000	01.01:2000
12:20:2000	01.01:2000
12:10:2000	01.01:2000
12:10:2000	01.01.2000
12:10:2000	01.10:2000

fig. 4-6 Statistics command results

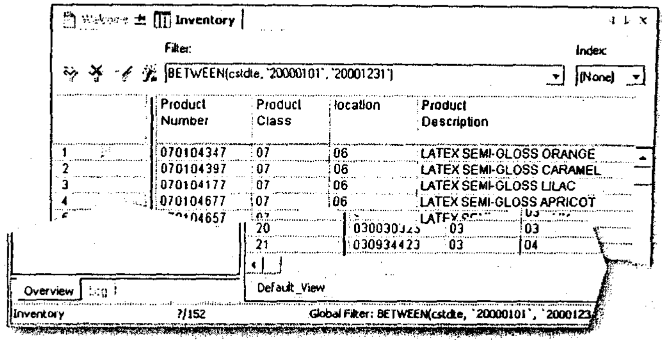
In *fig. 4-6*, results are produced by running the Statistics command on the cost date field in the Inventory table.

The Between( ) Function

An alternative to the Statistics command, the BETWEEN( ) function can be used in a global or command filter to isolate records where numeric, date, or character data are within specified bounds. You can count the number of records in the resulting table to ensure that you have the data requested.

Use the BETWEEN( ) function to check that dates fall within bounds

1. In the active table, click (Edit View Filter).
2. In the **Expression** box, enter an expression such as:
   `BETWEEN(Costdate, `20000101`, `20001231`)`
   in which:
   - The first parameter shows the name of the date field being examined.
   - The second and third parameters show:
     - first and last days of the time period requested
     - correct format
     - enclosed in reverse quotes
3. Click **OK**.
4. Select **Analyze** > **Count Records** and click **OK**.
   The number of records falling within the specified bounds appears in the status bar and is recorded in the log.



Filter: BETWEEN(cstdte, `20000101`, `20001231`)
Index: (None)

	Product Number	Product Class	location	Product Description
1	070104347	07	06	LATEX SEMI-GLOSS ORANGE
2	070104397	07	06	LATEX SEMI-GLOSS CARAMEL
3	070104177	07	06	LATEX SEMI-GLOSS LILAC
4	070104677	07	06	LATEX SEMI-GLOSS APRICOT

20	03003032	03	03
21	030934423	03	04

Inventory 7/152 Global Filter: BETWEEN(cstdte, `20000101`, `2000123

fig. 4-7 Between function results

In *fig. 4-7*, results are produced using the BETWEEN( ) function and the Count command with the cost date field in the Inventory table. If the number of records counted matches the number of records in your table, all of your records are within the specified bounds.

What If Your Records Are Not within Specified Bounds?

If you find that your table contains extraneous records, you can usually extract the valid records into a new table using a filter and analyze the new table. If there are insufficient records, have the data re-sent.

ⓘ *If you determine that a mistake was made in the preparation of your data, take special care to ensure that no other errors are present in your table.*

Finding Missing Items

Though your control totals may match the figures from your data provider, it is not a guarantee that your data is complete. It is possible that records or values were omitted on the source system. You can test for completeness using the Gaps command and the ISBLANK( ) function.

The Gaps Command

The Gaps command can find missing items from a list of sequential numbers. The Gaps command works on Character, Numeric, and Date field types although running the Gaps command on Date fields requires that you output the results to a table.

You can use the Gaps command to create a list of missing items in the following ways:

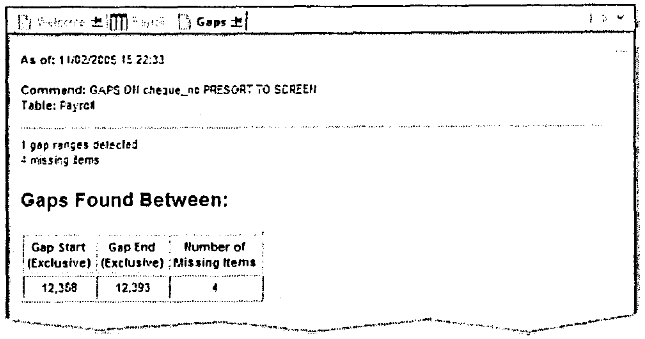
- Find ranges of missing items
- Find individual missing items

Find ranges of missing items

To find ranges of missing items:

1. Select **Analyze** > **Look for Gaps**.
2. Click **Gaps On....**
3. Choose an available field and add it to the **Selected Fields** list.

   You can only choose one field.
4. Click **OK**.
5. Ensure that **List Gap Ranges** is selected and click **OK**.



```
As of: 11/02/2005 15:22:33

Command: GAPS ON cheque_no PRESORT TO SCREEN
Table: Payroll

1 gap ranges detected
4 missing items
```

Gaps Found Between:

Gap Start (Exclusive)	Gap End (Exclusive)	Number of Missing Items
12,358	12,393	4

fig. 4-8 Gaps command results

In *fig. 4-8*, results are produced by running the Gaps command, with List Gap Ranges selected, on the check number field in the Payroll table.

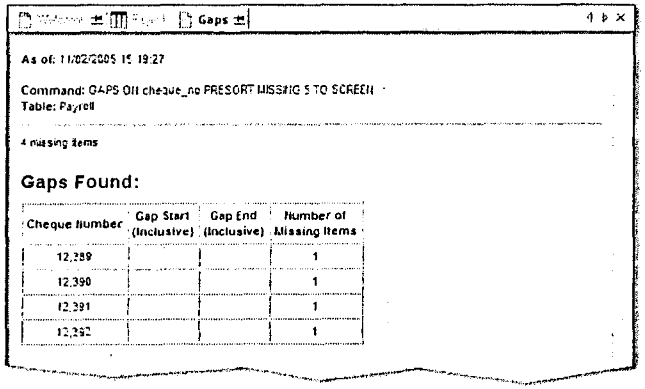
ⓘ *Note that the missing values are exclusive of the identified gap start and end.*

Finding individual missing items

To find individual missing items:

1. Select **Analyze** > **Look for Gaps**.
2. From the **Gaps On** list, select the field that you want to search for gaps, and click **OK**.
3. Select the **List Missing Items** option and click **OK**.

   The missing items are shown individually.



As of: 11/02/2005 15 19:27

Command: GAPS ON cheque_no PRESORT MISSING 5 TO SCREEN
Table: Payroll

4 missing items

Gaps Found:

Cheque Number	Gap Start (Inclusive)	Gap End (Inclusive)	Number of Missing Items
12,389			1
12,390			1
12,391			1
12,392			1

fig. 4-9 Gaps command results

In *fig. 4-9*, results were produced by running the Gaps command, with List Missing Items selected, on the check number field in the Payroll table.

ⓘ *The Maximum Missing Items option controls how missing items are presented. If there are gaps with more missing items than the value of that option, they appear in the gap start and end columns.*

The ISBLANK( ) Function

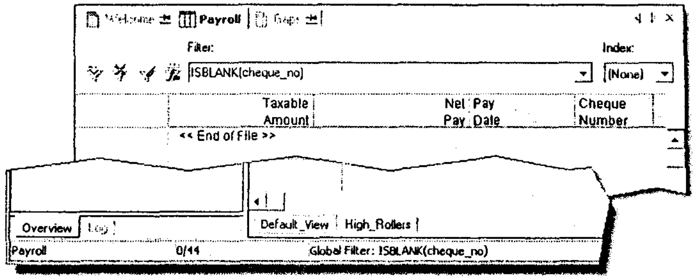
Like the Gaps command, the ISBLANK( ) function also checks for completeness. The ISBLANK( ) function is used in a global or command filter to find records with blank fields.

ⓘ *The ISBLANK( ) function works on Character fields only.*

To run the ISBLANK( ) function:

1. In the active table, click 𝑓ₓ (Edit View Filter).
2. In the **Expression** box, enter an expression:

   For example, `ISBLANK(cheque_no)` finds blanks in the `cheque_no` field.
3. Click **OK**.
4. Choose **Analyze** > **Count Records** and click **OK**.



Filter: ISBLANK(cheque_no)

Index: (None)

Taxable Amount	Net Pay	Pay Date	Cheque Number
<< End of File >>			

Payroll   0/44   Global Filter: ISBLANK(cheque_no)

fig. 4-10 ISBLANK( ) function results

The number of records with information missing appears in the status bar and is recorded in the log. None of the records in this table are missing information in the check number field.

In *fig. 4-10*, results were produced by using the ISBLANK( ) function with the check number field in the Payroll table.

Using filters to detect invalid or blank dates

To identify transactions with an invalid or blank date, apply a filter to the data.

For example: `Cost_Date='19000101'`

Using filters to detect zero value transactions

To identify transactions with numeric data equal to zero, apply a filter to the data.

For example: `Amount=0`

What If You Find Missing Items in Your Table?

Before proceeding with your analysis, you need to address missing items that you find.

Results from the Gaps command shows missing items in a sequence. Depending on the field, missing records may or may not be critical to your analysis. For example, you must not have gaps in a series of transaction numbers, but you may have gaps in a series of check numbers because of void checks. In either case, send your findings to the data provider.

The ISBLANK( ) function finds records where a field is blank. Certain fields in a table are always populated and others are not. For example, in a customer table, the customer number is never blank, but the second address is often left blank. If you find missing items in critical fields, contact the data provider before proceeding.

Notes:

Finding Duplicates

While duplicate records can show invalid data entry or even fraudulent activity, they can also occur in a valid context. For example, a transaction table might have duplicate customer numbers that show repeat transactions from the same customers. Fields that must never contain duplicates are ones where the values uniquely identify records. For example, an employee table must never have duplicate employee numbers because each value uniquely identifies each employee.

The Duplicates Command

The Duplicates command identifies records that have the same values. You can use it to identify records where all fields are duplicate or where duplicate values appear only in selected fields.

You can use the Duplicates command in the following ways:

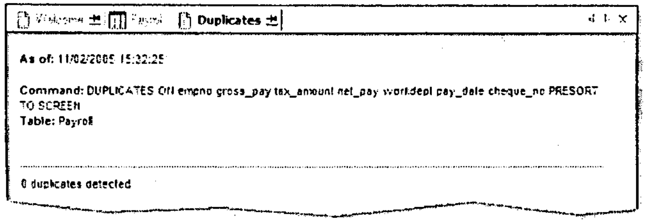
- Find duplicate records
- Find duplicate values

Finding duplicate records

To run the Duplicates command to find duplicate records:

1. Select **Analyze** > **Look for Duplicates.**
2. Click **Duplicates On.**
3. Add all the fields to the **Selected Fields** list and click **OK.**
4. Click the **Output** tab, select the **Screen** option, and click **OK.**

Notes:



As of: 11/02/2005 15:22:25

Command: DUPLICATES ON empno gross_pay tax_amount net_pay workdept pay_date cheque_no PRESORT TO SCREEN
Table: Payroll

0 duplicates detected

fig. 4-11 Duplicates command results

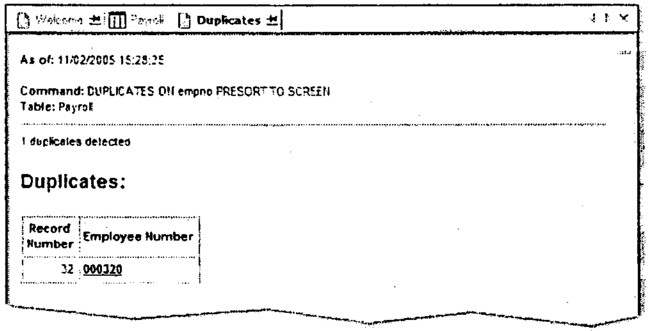
No duplicate records are found.

In *fig. 4-11*, results are produced by running the Duplicates command on all fields in the Payroll table.

Finding duplicate values

To run the Duplicates command to find duplicate values:

1. Select **Analyze** > **Look for Duplicates.**
2. Click **Duplicates On.**
3. Add the fields you want to search for duplicate values to the **Selected Fields** list and click **OK.**
4. Click the **Output** tab, select the **Screen** option, and click **OK.**



As of: 11/02/2005 15:28:25

Command: DUPLICATES ON empno PRESORT TO SCREEN
Table: Payroll

1 duplicates detected

Duplicates:

Record Number	Employee Number
32	000320

fig. 4-12 Duplicates command results

In *fig. 4-12*, results of the Duplicates command show how many duplicates were found and list the duplicated values. Each record number given is for the first duplicate of the value in the presorted table, not for the first occurrence of the value.

5. Click any value to see a complete list of the records that contain that value.

ⓘ *Narrow the scope of searches for duplicates to limit the results. For example, if you are trying to find duplicates of an employee's name, include the first name, initial, and last name fields in the search.*

What If You Find Duplicate Entries?

Duplicate entries must be examined in context to determine their validity. If you determine they are invalid, contact the data provider. If you want to continue working with this data, but do not want duplicates in your table, you can use the Summarize command to create a new table that does not contain duplicates.

ⓘ *For more information on the Summarize command, see "The Summarize Command" on page 58, or select Help > Index and enter "Summarize command".*

Testing for Reliability

Your data source can have values derived from calculations, usually based on other fields. For example, an invoice total field is the product of a unit cost field multiplied by a number of units field. It is necessary to recalculate these fields to ensure that they are reliable for use in your analysis.

Recalculating Numeric Values

Use computed fields to recalculate numeric values.

To recalculate numeric values:

1. Select **Edit** > **Table Layout** and click (Add a New Expression).
2. In **Name**, enter a name for the new computed field and click f(x).

   For example, to recalculate an invoice total, name the new computed field `c_inv_total`.
3. Open the **Expression Builder** and build an expression to recalculate the values.

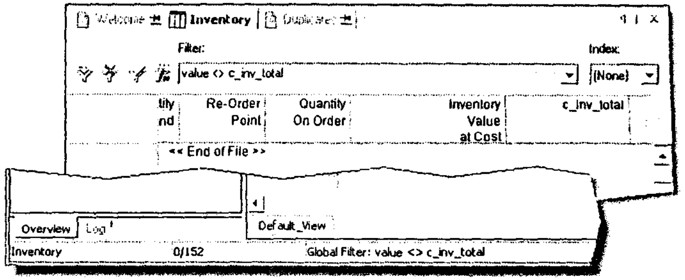
   For example, to calculate `c_inv_total`, the expression might be:

   ```
   unit_cost * quantity
   ```
4. Click **OK**.
5. Click (Accept Entry), click **OK**, and close the Table Layout window.

   ⓘ *To add the new field to your view, right-click in the view, and select Add Columns. In the Available Fields list, double-click the new field and click* **OK**.
6. To build a filter that isolates records with new computed values that do not match the original values, click (Edit View Filter).

   For example, an expression that compares new values to original values might be: `invoice_total <> c_inv_total`
7. Click OK.
8. Select **Analyze** > **Count Records** and click **OK**.

   The total number of records isolated by the filter appears in the status bar and is recorded in the log.



Filter: value <> c_inv_total

Index: (None)

lity nd	Re-Order Point	Quantity On Order	Inventory Value at Cost	c_inv_total
<< End of File >>				

Inventory 0/152 Global Filter: value <> c_inv_total

fig. 4-13 Count command results

In *fig. 4-13*, results are produced by recalculating the inventory value field in the Inventory table. A global filter and the Count command are used to show records whose computed values do not match the original values.

What If Your Computed Values Do Not Match?

If the errors are pervasive or significant, contact the data provider and have the file re-sent. Depending on how serious the errors are, you may be able to continue working with the data. In fact, by testing for the presence of errors, you have already created a new computed field with correct data.

Using Other Data Integrity Tests

There are many other ways of testing data integrity. For example, you can test for reasonableness and relations. While checking your data, you can use different *ACL* tools to verify data integrity.

Testing Reasonableness

Sometimes common sense assumptions can be made about what data must be found in a table. To test these assumptions, you can use various commands. For example:

- An online retailer makes an average of 47 transactions per day. If you receive a file containing three months of data, you expect to have approximately 4,320 records. You can use the Count command to confirm this estimate.
- The same retailer makes more small transactions ($0.00 to $1,000.00) than large transactions (over $1,000.00). You can use the Stratify command to confirm the distribution of transaction amounts.

ⓘ *For more information on the Stratify command, see "The Stratify Command" on page 61, or select Help > Index and enter "Stratify command".*

Testing Relations

Relation testing allows you to confirm that the relation between the data in two fields is consistent. You can use filters to compare two fields or you can create computed fields that calculate the difference between fields.

For example, in a transaction table, the units shipped must never be greater than the units ordered. Therefore, subtracting the units shipped from the units ordered must never yield a negative result.

Notes:

Data Analysis

About This Module

Data analysis is the most important phase in a project. It is the means by which you find answers to the questions defined in the planning phase. The bulk of the work on any project is usually spent performing data analysis.

The many features in *ACL* are designed to help you achieve fast, efficient, and thorough analyses. This module describes how to use *ACL* to achieve your objectives.

Profiling Data

Profiling data gives you an overview that can help identify trends and irregularities. The commands described below show you how to retrieve results to a file of any size.

ACL has five commands to help you profile your data:

- Classify
- Summarize
- Cross-tabulate
- Stratify
- Age

The Classify Command

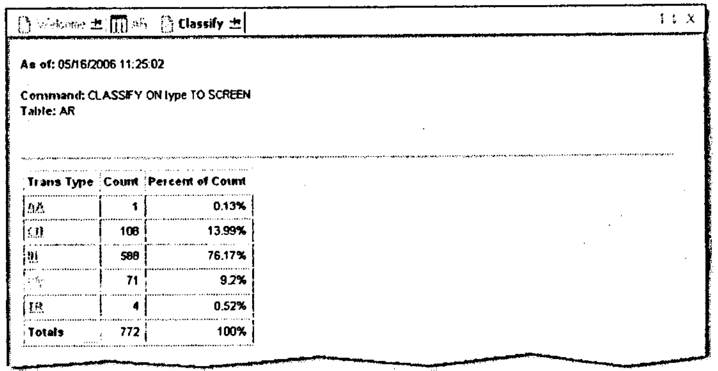
The Classify command creates a profile of your table by grouping unique character field values.

Use the Classify command to:

- Specify the character field upon which you want to base your analysis
- Specify the fields for which you want subtotals
- Send output to screen, graph, print, or file

To run the Classify command:

1. Select **Analyze** > **Classify.**
2. From the **Classify On** drop-down list, select the character field.
3. Click **OK.**



As of: 05/16/2006 11:25:02

Command: CLASSIFY ON type TO SCREEN
Table: AR

Trans Type	Count	Percent of Count
AA	1	0.13%
CR	108	13.99%
IN	588	76.17%
PY	71	9.2%
TR	4	0.52%
Totals	772	100%

fig. 5-1 Classify command

In *fig. 5-1*, results are produced by running the Classify command on the transaction type field in the AR table.

The Summarize Command

The Summarize command performs the same type of analysis as the Classify command, but it allows you to select one or more character or date fields for your analysis.

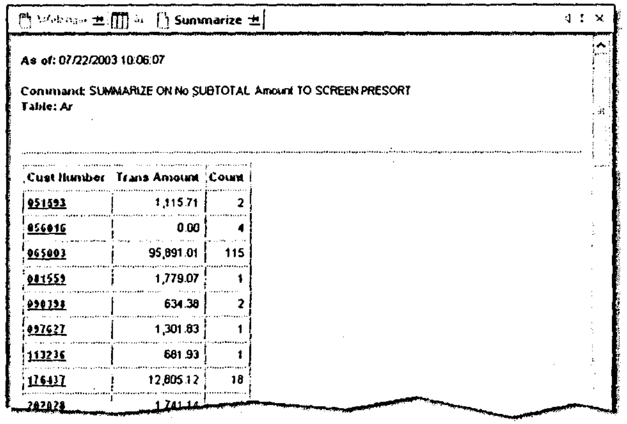
Use the Summarize command to:

- Specify the fields you want to summarize
- Specify the fields you want to subtotal
- Send output to screen, print, or file

To run the Summarize command:

1. Select **Analyze** > **Summarize.**
2. Click **Summarize On**, add the fields you want to summarize to the **Selected Fields** list, and click **OK.**

3. Click **Subtotal Fields**, add the fields for which you want to create subtotals to the **Selected Fields** list, and click **OK**.
4. Click the **Output** tab, select the **Screen** option, and click **OK**.



As of: 07/22/2003 10:06:07

Command: SUMMARIZE ON No SUBTOTAL Amount TO SCREEN PRESORT
Table: Ar

Cust Number	Trans Amount	Count
051593	1,115.71	2
056016	0.00	4
065003	95,891.01	115
081559	1,779.07	1
090398	634.38	2
097627	1,301.83	1
113236	681.93	1
176437	12,805.12	18
202028	1,741.14	

fig. 5-2 Summarize command results

In *fig. 5-2*, results are produced by running the Summarize command on the customer number field in the AR table.

Comparing Classify and Summarize

The Classify and Summarize commands are similar, but they have different options available and differences in the way they are processed. The following table compares Classify and Summarize.

Functional Specification	Classify	Summarize
Calculates and reports the number of times a key field appears in the table	Yes	Yes
Computes and displays subtotals on selected numeric fields	Yes	Yes
Computes and displays percentages	Yes	No
Key field can be Character	Yes	Yes
Key field can be Date	No	Yes
Number of key fields allowed	One	One or more
Allows the display of other fields	No	Yes
Data needs to be sorted on key field prior to running command	No	Yes (presort option available)
Primary location utilized to handle sequence process, subtotaling of numeric fields, and compression of details	RAM	Hard disk
Can send results to Screen	Yes	Yes
Can send results to Table	Yes	Yes
Can send results to Graph	Yes	No

The Cross-Tabulate Command

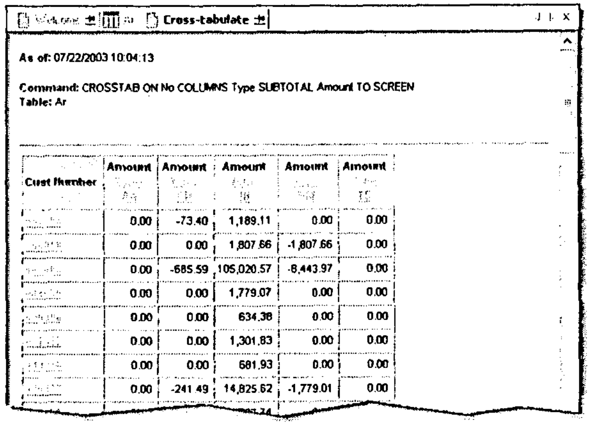
The Cross-tabulate command lets you analyze character fields by setting them in rows and columns. By cross-tabulating character fields, you can do the following:

- Specify a character column to be cross-tabulated with the key field. For example, there may be several different transaction types associated with each unique customer number.
- Specify a character row upon which the analysis of the columns is based. The row is usually a key field such as customer number.
- Subtotal numeric fields for each intersection of a row and column value. For example, total the numeric values associated with the intersection of each unique customer number with each unique transaction type.
- Send output to screen, graph, print, or file.

To run the Cross-tabulate command:

1. Select **Analyze** > **Cross-tabulate.**
2. From the **Rows** list, select a character field.
3. From the **Columns** drop-down list, select a character field.
4. Click **Subtotal Fields.**

   The Selected Fields dialog box opens.
5. Add numeric fields to the **Selected Fields** list by double clicking them, or by selecting them in the **Available Fields** list and clicking the right arrow button.
6. When you have selected the fields you want, click **OK.**
7. Click the **Output** tab, select the **Screen** option, and click **OK.**



As of: 07/22/2003 10:04:13

Command: CROSSTAB ON No COLUMNS Type SUBTOTAL Amount TO SCREEN
Table: Ar

Cust Number	Amount	Amount	Amount	Amount	Amount
[illegible]	0.00	-73.40	1,189.11	0.00	0.00
[illegible]	0.00	0.00	1,807.66	-1,807.66	0.00
[illegible]	0.00	-685.59	105,020.57	-8,443.97	0.00
[illegible]	0.00	0.00	1,779.07	0.00	0.00
[illegible]	0.00	0.00	634.38	0.00	0.00
[illegible]	0.00	0.00	1,301.83	0.00	0.00
[illegible]	0.00	0.00	681.93	0.00	0.00
[illegible]	0.00	-241.49	14,825.62	-1,779.01	0.00

fig. 5-3 Cross-tabulate command results

In *fig. 5-3*, results are produced by running the Cross-tabulate command on the customer number and the invoice type fields in the AR table.

Notes:

The Stratify Command

The Stratify command creates a profile of your table by grouping records into ranges based on a numeric field. For example, you can use the Stratify command to identify ranges with an unusually high number of transactions, such as multiple credit card transactions just below a spending authorization limit.

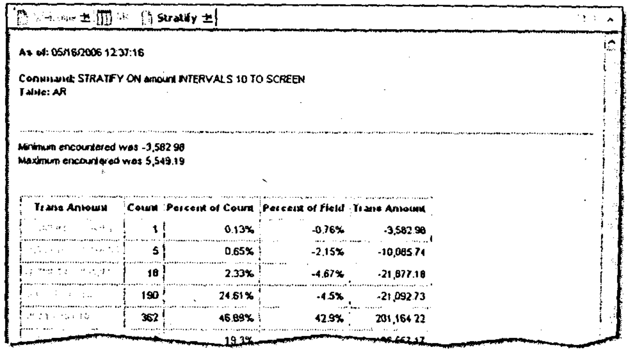
Use the Stratify command to:

- Specify the numeric field that you want to stratify
- Select the minimum and maximum values of the range upon which the command is to be executed
- Specify how many intervals you want to divide the data into
- Specify which fields are to have subtotals
- Send output to screen, graph, print, or file

To run the Stratify command:

1. Select **Analyze** > **Statistical** > **Statistics.**
2. From the **Statistics On** list, select the numeric field you want to run the Stratify command on and click **OK**.
3. Select **Analyze** > **Stratify.**
4. From the **Stratify On** drop-down list, select the field selected in Step 2.

   The values generated by the Statistics command appear in the **Minimum** and **Maximum** boxes.
5. Click **Subtotal Fields**, add the numeric fields for which you want subtotals to the **Selected Fields** list, and click **OK**.
6. Click OK.



As of: 05/16/2006 12:37:16

Command: STRATIFY ON amount INTERVALS 10 TO SCREEN
Table: AR

Minimum encountered was -3,582.98
Maximum encountered was 5,549.19

Trans Amount	Count	Percent of Count	Percent of Field	Trans Amount
[illegible]	1	0.13%	-0.76%	-3,582.98
[illegible]	5	0.65%	-2.15%	-10,085.74
[illegible]	18	2.33%	-4.67%	-21,877.18
[illegible]	190	24.61%	-4.5%	-21,092.73
[illegible]	362	46.89%	42.9%	201,164.22
[illegible]	[illegible]	19.3%	[illegible]	[illegible]

fig. 5-4 Stratify command results

In *fig. 5-4*, results are produced by running the Stratify command on the transaction amount field in the AR table.

Use the Stratify command with custom ranges

Custom ranges can be manually entered in the Stratify command by selecting the **Free** option and entering each range. Each value must be entered on a separate line, beginning with the smallest value.

For example:

-4000

-3000

-2000

-1000

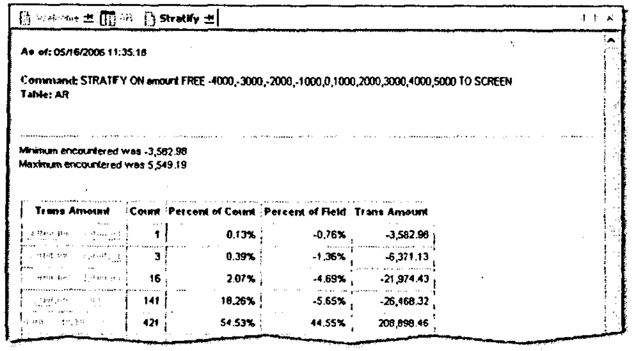
0

1000

2000

3000

4000



```
As of: 05/16/2006 11:35:18

Command: STRATIFY ON amount FREE -4000,-3000,-2000,-1000,0,1000,2000,3000,4000,5000 TO SCREEN
Table: AR

Minimum encountered was -3,582.98
Maximum encountered was 5,549.19

 Trans Amount    Count  Percent of Count  Percent of Field  Trans Amount
 [illegible]         1             0.13%            -0.76%     -3,582.98
 [illegible]         3             0.39%            -1.36%     -6,371.13
 [illegible]        16             2.07%            -4.69%    -21,974.43
 [illegible]       141            18.26%            -5.65%    -26,468.32
 [illegible]       421            54.53%            44.55%    208,898.46
```

fig. 5-5 Stratify command results

In *fig. 5-5*, results are produced by running the Stratify command with the **Free** option on the transaction amount field in the AR table.

Notes:

The Age Command

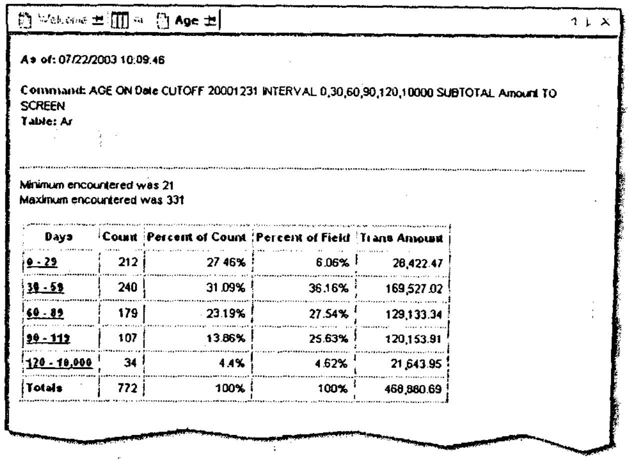
The Age command creates a profile of your table by grouping records into specified ranges or aging periods. The most common use of the Age command is to age accounts receivable data, displaying the records in 0-30, 31-60, and 61-90 day ranges.

Use the Age command to:

- Specify the date field that you want to age
- Specify a cutoff date or use the current system date
- Specify the aging periods or use the default values
- Specify which fields have subtotals
- Send output to screen, graph, print, or file

To run the Age command:

1. Select **Analyze** > **Age**.
2. From the **Age On** drop-down list, select the date field which you want to age and click **OK**.
3. In the **Cutoff Date** drop-down calendar, navigate to the cutoff date you want.
4. Click **Subtotal Fields,** add the numeric fields for which you want subtotals to the **Selected Fields** list, and click **OK**.
5. Click **OK**.



As of: 07/22/2003 10:09:46

Command: AGE ON Date CUTOFF 20001231 INTERVAL 0,30,60,90,120,10000 SUBTOTAL Amount TO SCREEN
Table: Ar

Minimum encountered was 21
Maximum encountered was 331

Days	Count	Percent of Count	Percent of Field	Trans Amount
0 - 29	212	27.46%	6.06%	28,422.47
30 - 59	240	31.09%	36.16%	169,527.02
60 - 89	179	23.19%	27.54%	129,133.34
90 - 119	107	13.86%	25.63%	120,153.91
120 - 10,000	34	4.4%	4.62%	21,643.95
Totals	772	100%	100%	468,880.69

fig. 5-6 Age command results

In *fig. 5-6*, results are produced by running the Age command on the invoice date field in the AR table. A cutoff date of December 31, 2000 is used.

Isolating Data

When you analyze a table, you are likely to encounter data that is irrelevant to your analytic objectives. *ACL* offers the following ways to isolate data in a table:

- Filters
- The Extract command
- The Export command

Filters

Global and command filters let you isolate records from a table without physically removing them. However, a filter can be inefficient when it is returning only a few records out of a large table. To avoid this situation, extract the target records to a new table using a command filter.

The Extract Command

You can use the Extract command to create a new table from the records and fields of an existing table. This can significantly reduce processing time by reducing the table to just those records and fields of concern. And, because the command automatically creates a data file and table layout, you can immediately proceed with your analysis on the new table.

When using the Extract command, choose from the following options:

Record	Creates a table with an identical structure to the table from which you are extracting the data. The new table includes all defined and undefined areas of the input record and computed field definitions.
Fields	Creates a table containing selected fields only. If computed fields are included in the Fields list, they are first calculated before being extracted to the new table. They are given one of the following data types: • Logical • Date • ASCII • *ACL* (a numeric data type) • EBCDIC • PACKED

To run the Extract command:

1. Select **Data** > **Extract Data.**
2. Select **Record.**

   Alternatively, you can select **Fields** and choose the fields from the **Extract Fields** list.
3. Click **If** and enter an expression to filter records.
4. In the **To** box, enter a name and click **OK.**

The Export Command

The Export command allows you to use *ACL* as a data conversion tool. *ACL* can export files in formats readable by other applications. You can export to the following formats:

- Access
- Clipboard - for pasting into other applications
- dBASE III PLUS
- Delimited text
- Excel
- Lotus
- Text
- Word merge files
- WordPerfect 4.1 or 6.0 merge files
- XML

To run the Export command:

1. Select **Data** > **Export To Other Application.**

   The Export dialog box opens. The Export command lets you select fields or views as the data elements to be processed by the other application

2. Do one of the following:
   - With **Fields** selected, click **Export Fields** and choose the fields in the order they need to appear in the exported file.
   - With **View** selected, export data based on the fields in the active view. The exported file has the same fields, in the same order, as your view.

3. Click **If** and enter an expression to filter records.

4. From the **Export As** drop-down list, select the file format to which you want to export the data.

5. In the **To** box, enter the file name and click **OK.**

Notes:

Organizing Data

It is often necessary to reorder data, both to clarify the meaning of the data and to prepare for the requirements of subsequent commands.

You can use the following commands to organize data:

- The Sequence command
- The Sort command
- The Index command
- Quick Sort

The Sequence Command

The Sequence command tests that the selected fields are in sequence.

ⓘ *The Sequence command does not modify the order of the original table.*

To run the Sequence command:

1. Select **Analyze** > **Examine Sequence.**
2. Click **Sequence On** and add the fields you want to sequence to the **Selected Fields** list.
3. In the **Selected Fields** list, click the arrow beside each item to select ascending or descending order and click **OK.**
4. Click OK.

The Sort Command

Use the Sort command to create a new table in which records are ordered based on specified key fields.

ⓘ *To sort a table, ACL requires free disk space at least 2.5 times the size of the table being sorted.*

To run the Sort command:

1. Select **Data** > **Sort Records.**
2. Click **Sort On** and add the fields you want to sort to the **Selected fields** list in the order of priority.
3. In the **Selected Fields** list, click the arrow beside each item to select ascending or descending order and click **OK.**
4. In the **To** box, enter a name and click **OK.**

The Index Command

An index allows you to work with a table in a specified order without creating a new sorted table. The Index command creates an index file with the extension .inx. An index provides direct access to the records of a table in a logical rather than a physical order.

The status bar shows when the records in a table are indexed. You activate and deactivate an index using the Index drop-down list in the view.

To run the Index command:

1. Select **Data** > **Create Index.**
2. Click **Index On** and add the fields you want to sequence to the **Selected fields** list in the order of priority.
3. In the **Selected Fields** list, click the arrow beside each item to select ascending or descending order and click **OK.**
4. In the **To** box, enter a name and click **OK.**

Comparing Sort and Index

The Sort and Index commands are used for similar results although there are significant differences in how they operate. As a result, the processing time and storage space required differ.

The following table compares the Sort and Index commands against given requirements:

Requirements	Sort	Index
Execution speed	Slower	Faster
Resulting file size	Larger	Smaller
Required disk space	More	Less
Subsequent processing of an entire table	Much faster	Much slower
Subsequent processing searching for a few records	Much Slower	Much faster

Quick Sort

Another feature for organizing records is Quick Sort. Use this feature to temporarily sort a table based on a single field in a view without creating a new sorted table.

To run Quick Sort:

- Right-click the column heading and select **Quick Sort Ascending** or **Quick Sort Descending.**

To turn off Quick Sort:

- Right-click the column heading and select **Quick Sort Off.**

Notes:

Combining Data

In many cases, your analysis requires data from two or more tables. To access and analyze data in different tables as if it were a single table, you need to combine the tables.

ACL provides the following methods to combine tables:

- The Extract command with the Append option
- The Join command
- The Relations command

The method you choose depends on the structure of the tables you want to combine and the objectives you are trying to achieve.

The Extract Command with the Append Option

When you receive multiple tables that have the same structure and contain the same type of information, it is often advantageous to extract the data from each original table into a new master table. For example, use this technique to combine transactions from multiple periods. Notice that the tables you are appending must have the same structure as the master table.

ⓘ *By appending records to a new table, you preserve the integrity of your original source tables.*

To run the Extract command with the Append option:

1. Select **Data** > **Extract Data.**
2. If you want to restrict the records you extract, in the **If** box, enter an expression.
3. In the **To** box, enter the new table name and click **OK.**
4. In the **Overview** tab, double-click the second table from which you want to extract records.
5. Select **Data** > **Extract Data.**
6. If you want to restrict the records you extract, in the **If** box, enter an expression.
7. In the **To** box, enter the same table name to which you sent the records from the previous table, click **OK**, and select **Append.**

Notes:

The Join Command

The Join command combines fields from two different tables to create a third table consisting of matched or unmatched records from each table, or a combination of the two. The records written to the third table are based on the comparison of common "key fields" in both original tables. The fields contained in the third table can be any combination of fields from the two source tables.

When to use the Join command

The Join command is typically used to match records in a transaction table with those in a master table.

Use the Join command to:

- Compare customer account balances to customer credit limits to identify if customers have exceeded their credit limit.
- Compare Employee records to a vendor listing to check for employees who are doubling as vendors.
- Compare Physician billing records to insurance claims to ensure that claim amounts are accurate.

Join command rules

The following rules must be followed when running the Join command:

- The primary and secondary tables must exist in the same *ACL* project.
- Files that are accessed using the *ACL* Client/Server System must be on the same server and must be accessed using the same server profile.
- Both tables must share at least one common key field. The field names do not have to be the same, but they must describe the same data element.
- The key field must be the same data type in both tables.
- The key field must be the same length in both tables. If key fields are not the same length, use functions to harmonize them. For example, the SUBSTRING( ) function lets you adjust the length and content of a character field.

Join command guidelines

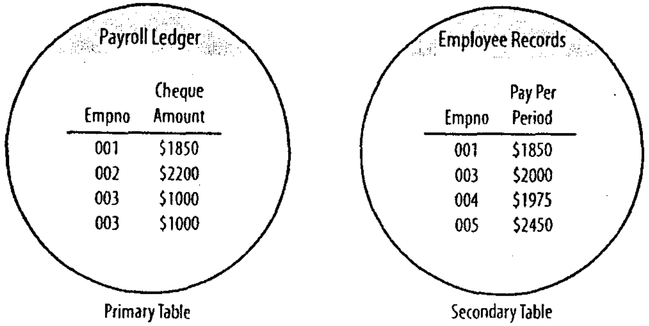
When running the Join command using character key fields, follow these guidelines to ensure the best results:

- The key fields must both be in the same case: UPPER, lower, or Proper. To harmonize the case, use one of these functions: UPPER( ), LOWER( ), or PROPER( ).
- The key fields must both have the same justification. To harmonize the justification, use the LTRIM( ) function to remove leading blanks.

ⓘ *If you apply a filter when running the Join command, it references and restricts records from the primary table only.*

The Join command requires a primary and a secondary input table. A transactional table is typically used as the primary table, and a master table as the secondary table. There are six options for combining the two tables using the Join command. The first five options, which are illustrated in the following sections, are examples of many-to-one joins. The sixth option, a many-to-many join, is taught in advanced *ACL* courses only.

ⓘ *To learn more about many-to-one joins, see "Duplicate keys and Join command results" on page 73.*



Empno	Cheque Amount
001	$1850
002	$2200
003	$1000
003	$1000

Empno	Pay Per Period
001	$1850
003	$2000
004	$1975
005	$2450

fig. 5-7 Join command example

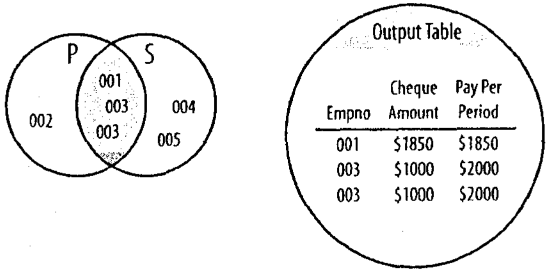
In *fig. 5-7*, the example shows the following:

- The Payroll Ledger table has a single pay period and includes all payroll disbursements. One employee, 003, was paid twice. Note how duplicate records in the primary table are handled by different join types.
- The Employee Records table is maintained by the human resources department. Employee records consist of a complete list of valid employees and the amount that they are paid each period. One employee, 002, is missing from the sequence.

In each of the five following join-type examples, the objective is to test for employees who have been paid incorrectly.

Matched primary records

Creates one output record for every record in the primary table that has a match in the secondary table.



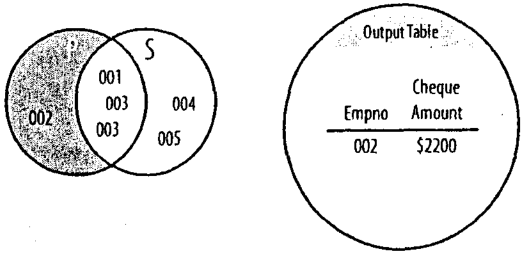
Empno	Cheque Amount	Pay Per Period
001	$1850	$1850
003	$1000	$2000
003	$1000	$2000

fig. 5-8 Matched primary records example

In *fig. 5-8*, the results of this join show all employees who have been paid and who are also listed in the employee records table. Analysis of this table allows you to determine if any valid employees were paid incorrectly.

Unmatched primary records

Creates one output record for every record in the primary table that does not have a match in the secondary table.



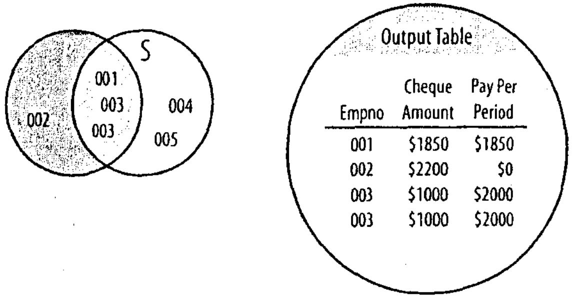
Empno	Cheque Amount
002	$2200

fig. 5-9 Unmatched primary records

In *fig. 5-9*, the results of this join show all employees who have been paid but who are not listed in the employee records table. This allows you to identify checks issued to invalid employee numbers.

Matched primary records, include all primary records

Creates one output record for every record in the primary table that has a match in the secondary table, and one additional record for every unmatched record in the primary table.



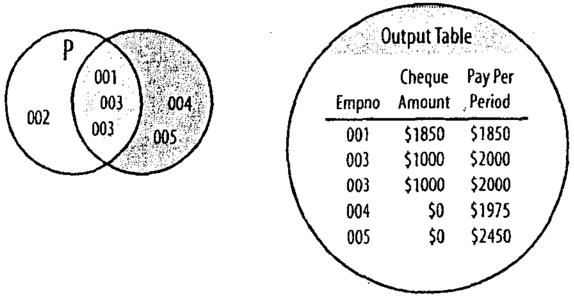
Empno	Cheque Amount	Pay Per Period
001	$1850	$1850
002	$2200	$0
003	$1000	$2000
003	$1000	$2000

fig. 5-10 Matched primary records, include all primary records

In *fig. 5-10*, the results of this join are the equivalent to the combined results of the two preceding tests. You can account for all checks issued.

Matched primary records, include all secondary records

Creates one output record for every record in the primary table that has a match in the secondary table, and one additional record for every unmatched record in the secondary table.



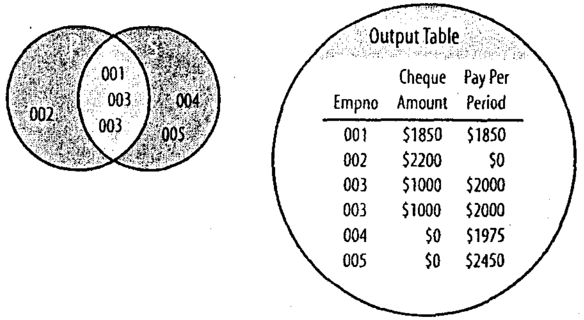
Empno	Cheque Amount	Pay Per Period
001	$1850	$1850
003	$1000	$2000
003	$1000	$2000
004	$0	$1975
005	$0	$2450

fig. 5-11 Matched primary records, include all secondary records

In *fig. 5-11*, this join lets you account for all employees listed in the employee records table. You can identify valid employees who were paid incorrectly and those who were not paid at all.

Matched primary records, include all primary and secondary records

Creates one output record for every record in the primary table that has a match in the secondary table, and one additional record for every unmatched record in the primary and secondary tables.



Empno	Cheque Amount	Pay Per Period
001	$1850	$1850
002	$2200	$0
003	$1000	$2000
003	$1000	$2000
004	$0	$1975
005	$0	$2450

fig. 5-12 Matched primary records, include all primary and secondary records

In *fig. 5-12*, the results of this join account for all payroll checks issued and all employees in the employee records table. You can identify incorrect payments (amounts and duplicates), payments to invalid employee numbers, and employees who were not paid at all.

Many-to-many matched records

Creates one output record for every match between records in the primary and secondary tables.

Many-to-many is the sixth and most complex type of join. It is used in a limited number of scenarios. Use this option cautiously because it is capable of producing a table that is much larger than the combined size of both original tables. This type of join is taught in advanced *ACL* courses only.

How to join tables

Plan the join

1. Determine the objective: what results are you looking for?
2. Identify the two tables to be joined.
3. Identify the key field(s) common to both tables.
4. Examine the key field(s) in both tables to ensure that they:
   - Are the same data type
   - Are the same length
   - Have the same case and justification
5. Identify and select the table to be the primary table. Usually the primary table has data that is supplemented with data from the secondary table.
6. When you have selected your primary and secondary tables, determine the type of join to use. Which type you choose may depend on which table you have selected as the primary.

Test and prepare the table

1. Test the key fields in the tables for duplicates. If you encounter duplicates, you might consider removing them with the Classify or Summarize commands.

   ⓘ *For more information, see "Duplicate keys and Join command results" on page 73.*
2. Test the key fields in both tables for blanks. If blanks exist in both tables, they are matched together, but they might not represent comparable data.
3. Test both tables for sequence on the key fields. If tables are not in sequential order, they must be sorted prior to the Join or sorted during the Join process using the Presort option. If the tables were previously classified or summarized on the same key fields as used in the join, they are already in sequential order based on those fields.

Use the Join command

1. In the **Overview** tab, double-click the primary table.
2. Select **Data** > **Join Tables**.
3. From the **Secondary Table** drop-down list, select the secondary table.
4. From the **Primary** and **Secondary Keys** lists, select the key fields.
5. From the **Primary** and **Secondary Fields** lists, select the fields you want to include in the final table.

   ⓘ *Primary and secondary key fields are only included in the final table if also selected as primary or secondary fields.*
6. Select the **Presort** check box as necessary.
7. In the **If** box, enter any filter you want applied.
8. Click the **More** tab, select the type of join you want, and click the **Main** tab
9. In the **To** box, name the new table.
10. Click **OK**.
11. Check the log to see information regarding the outcome of the join.

ⓘ *ACL resolves duplicate field names by adding a "2" to the name from the secondary table.*

Duplicate keys and Join command results

When you execute a many-to-one join using a secondary table in which duplicate keys exist, primary records are matched only to the first occurrence of the duplicated key in the secondary table. This is shown in the diagram below:

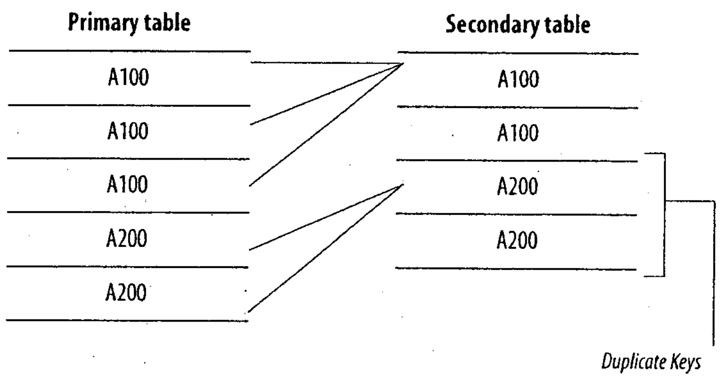


fig. 5-13 Duplicate results

ⓘ *Duplicate keys can be removed from the secondary table by using the Summarize command.*

Notes:

The Relations Command

The Relations command can access data in multiple tables simultaneously. This lets you recreate the environment of a relational database, even including tables from different systems.

Before you start to use the Relations command, however, it is important to have a good understanding of your tables. The illustration below identifies the common key fields that are the basis of the relations between the tables.

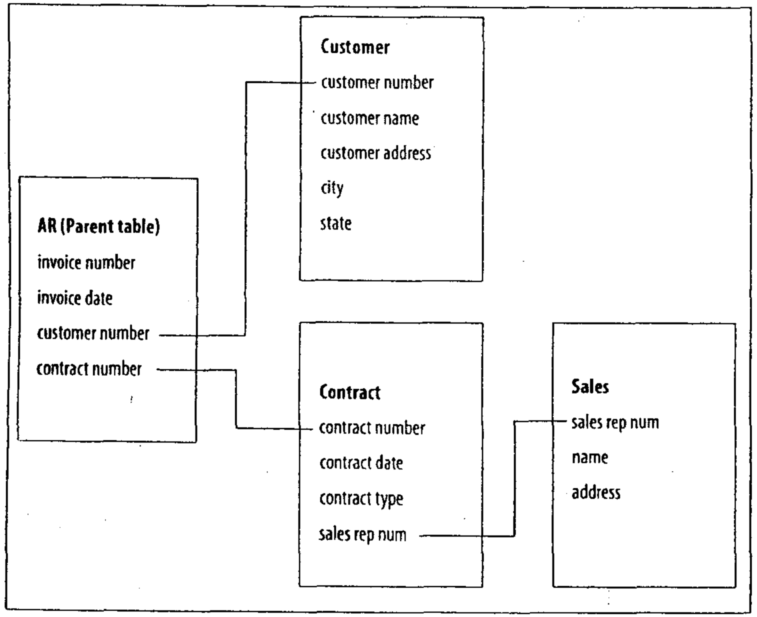


fig. 5-14 Relations command example

ⓘ *The relation between two tables created by the Relations command must be based on a single key field in each table.*

The Relations command is the logical equivalent to the Join command using a Matched Records, All Primary join. However, the results of the Relations command are virtual and do not result in the creation of a new table. Instead, the fields of the related or child table(s) become available in the parent table in which the relation was created.

In the Join command, you are able to match data from just two tables: the primary and secondary tables. The Relations command lets you relate a single parent table (equivalent to the primary table) to many child tables (equivalent to secondary tables).

There is no limit to how many relations can exist in a parent table. However, when displaying fields in a view, specifying fields in a command, or when creating computed fields, you can only refer to fields in a maximum of 18 tables.

The logic used to match records in the Relations command is similar to the many-to-one logic of the Join command. If duplicate keys exist in the child table, *ACL* matches the first key encountered. If no matching key is found in the related table, the related fields are given blank or zero values in the view of the parent table.

Relations command rules

The following rules must be followed when running the Relations command:

- All tables must exist within the same project. You can link and reference up to 18 tables at one time, including the parent table.
- Files that are accessed using the *ACL* Client/Server System must be on the same server and accessed using the same server profile.

- All tables must share one common key field. The field names do not have to be the same, but they must describe the same data element.
- The key field must be the same data type in the tables you are relating.
- You must define and store relations within the parent table. You can only view relations when the parent table is open.
- To delete or modify an existing relation, do so from within the parent table.
- When a relation is defined, the type of the key fields from either table cannot be modified.

Relations command guidelines

When running the Relations command using character key fields, follow these guidelines to ensure the best results:

- The key fields must be the same length. If key fields are not the same length, use functions to harmonize them. For example, the SUBSTRING( ) function allows you to adjust the length and content of a character field.
- The key fields must both be in the same case: UPPER, lower, or Proper. To harmonize the case, use one of these functions: UPPER( ), LOWER( ), or PROPER( ).
- Key fields must have the same justification. Use the LTRIM( ) function to remove leading blanks from the key fields.

Relating Tables

Before you can relate tables, ensure you do the following:

- Plan the relation
- Test and prepare the tables

Plan the relation

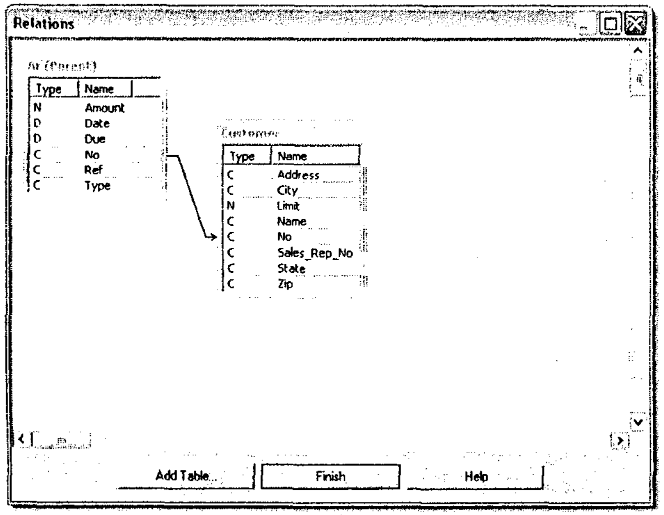
1. Determine the objective: what results are you looking for?
2. Identify the two tables to be related.
3. Identify the key field that is common to both tables.
4. Examine the key field in both tables to ensure that they:
    - Are the same data type
    - Are the same length
    - Have the same case and justification
5. Determine which table is to be the parent table. Usually the parent table contains data that is complemented with data from the child table.

Test and prepare the tables

1. Test the key field in each child table for duplicates. If duplicate keys are encountered in a child table, only the first key is selected. If the duplicate keys are representative of a larger item, such as transactions composing an account balance, the Summarize command can eliminate the duplicates and total the balances.
2. Test the key field in each table for blanks. If blanks exist in both tables, they are matched together, but they might not represent comparable data.

To relate a table:

1. In the **Overview** tab, double-click the parent table.
2. Select **Data** > **Relate Tables**.
3. Click **Add Table** and add the child table to the Relations dialog box.
4. Click and drag the key field from the parent table to the key field in the child table.
5. Click **Finish**.



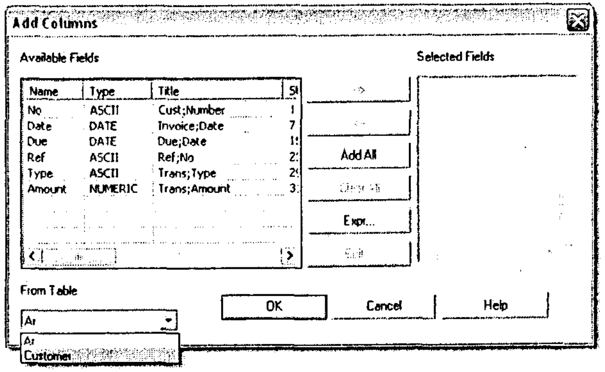
Type	Name
N	Amount
D	Date
D	Due
C	No
C	Ref
C	Type

Type	Name
C	Address
C	City
N	Limit
C	Name
C	No
C	Sales_Rep_No
C	State
C	Zip

fig. 5-15 Relations dialog box

Add related fields to a view

1. Right-click in the parent table and select **Add Columns**.
2. From the **From Table** drop-down list, select the child table.
3. Add the field(s) you want in the view to the **Selected Fields** list and click **OK**.



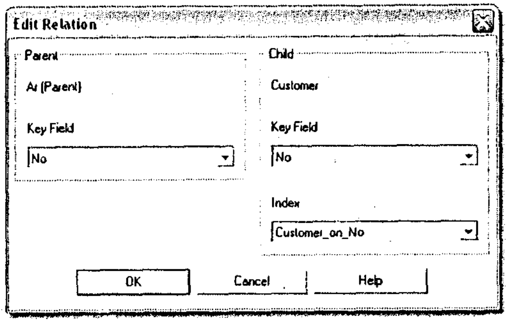
Name	Type	Title	St
No	ASCII	Cust;Number	1
Date	DATE	Invoice;Date	7
Due	DATE	Due;Date	1
Ref	ASCII	Ref;No	2
Type	ASCII	Trans;Type	2
Amount	NUMERIC	Trans;Amount	3

fig. 5-16 Add Columns dialog box

ⓘ *When related fields are referred to in a parent table, their names are prefaced with the name of the table to which they belong. For this reason, naming conflicts are not possible when using the Relations command.*

Edit a relation

1. With the parent table open, select **Data** > **Relate Tables**.
2. Right-click the link between two tables and select **Edit Relation**.
3. In the **Edit Relation** dialog box, change the key field of the parent or child table.



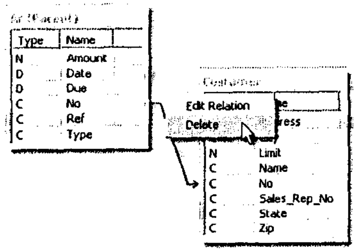
Edit Relation

Parent	Child
Ar (Parent)	Customer
Key Field: No	Key Field: No
	Index: Customer_on_No

fig. 5-17 Edit Relation dialog box

Delete a relation

1. With the parent table open, select **Data** > **Relate Tables**.
2. Right-click the link between two tables and select **Delete**.



Ar (Parent)

Type	Name
N	Amount
D	Date
D	Due
C	No
C	Ref
C	Type

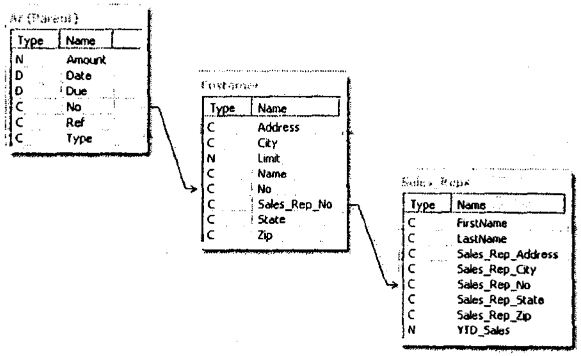
Customer

Type	Name
	ress
N	Limit
C	Name
C	No
C	Sales_Rep_No
C	State
C	Zip

fig. 5-18 Delete relations

Create an indirect relation

In *ACL*, all relations must be created in the parent table. However, relations can be established indirectly between tables, as shown in the example below, in which the AR table is related indirectly to the Sales_Reps table. Through this indirect relation, the AR table can be populated with fields from the Sales_Reps table. Child-child relations are created in the same manner as parent-child relations.



Ar (Parent)

Type	Name
N	Amount
D	Date
D	Due
C	No
C	Ref
C	Type

Customer

Type	Name
C	Address
C	City
N	Limit
C	Name
C	No
C	Sales_Rep_No
C	State
C	Zip

Sales_Reps

Type	Name
C	FirstName
C	LastName
C	Sales_Rep_Address
C	Sales_Rep_City
C	Sales_Rep_No
C	Sales_Rep_State
C	Sales_Rep_Zip
N	YTD_Sales

fig. 5-19 Indirect relations

Comparing Joins and Relations

The Join and Relations commands allow you to work with data from more than one table at one time. However, one command might be better suited to a task than the other. The table below suggests in what situations each might be used and how to use them.

Objective	Join Command	Relations Command
List primary records that have a match in the secondary table.	Matched Records option	After relating the tables, use filters to identify and display the matched records.
List primary records that do not have a match in the secondary table.	Unmatched Records option	Use filters after relating the tables to identify and display the unmatched records.
List primary records that have a match in the secondary table plus records found only in the primary table.	Matched Records, All Primary option	Equivalent to Join Matched Records, All Primary. Use filters to identify unmatched items.
List primary records that have a match in the secondary table plus records found only in the secondary table.	Matched Records, All Secondary option	No equivalent.
List primary records that have a match in the secondary table plus records found only in either the primary or secondary tables.	Matched Records, All Primary and All Secondary option	No equivalent.

Technical differences

There are also technical differences between the Join and Relations commands. The table below outlines these differences.

Requirement	Join Command	Relations Command
Disk space required for results	The Join command creates a third table containing join results. It is possible for this table to be larger than both original tables combined.	Minimal disk space is required to create an index for the child table(s).
Time to produce results	The duration of the execution of the Join command can vary depending on the complexity of the join, and whether or not the primary table is sorted.	No actual record matching is done at the time the Relations command is executed. For this reason, it takes considerably less time than the Join command.
Time to process results	The results of the Join command are stored in a flat table. Flat files can be processed very quickly.	Matching of records is done at the time of command processing and takes longer than with Join command.
Presort or Index	Sort or Presort required for secondary tables (optional for primary tables).	Index is required for child tables (created automatically by the Relations command).
Number of key fields	One or more key fields can be selected from each table.	Limited to one key field per table. Note: If more than one key is required to establish the relation, create a computed field to concatenate all the fields. Apply this process to both tables.

Reporting Results

6

About this Module

Reporting with ACL

Creating a Report from a View

Advanced Reporting

Generating Reports with Crystal Reports

Creating Graphs

Using the Log

Documenting Your Analysis

The presentation of your analysis is important. You require flexible reporting tools that let you present your results in a variety of ways, in formats that are easy to review and understand. You also need tools to help you track and document your analysis.

ACL has numerous reporting features that offer you efficiency and versatility in the creation of your reports. This module introduces you to these features.

Reporting with ACL

When you complete your analysis, you must present your findings. You may also need to provide supporting documentation that details the analysis you performed.

ACL has numerous features that let you easily generate what you need for your reports and supporting documentation.

Views

You can format a view to produce a professional-looking report that can be sent to print or exported to other applications. If you want to enhance and standardize reporting in your organization, you can take advantage of Crystal Reports® integration and generate reports using customized Crystal Reports templates.

Graphs

For a visual presentation of your findings, you can generate graphs from your results and customize the graphs with a variety of formatting options.

Results tab

You can easily copy and paste results from a results tab to other applications. Because the results are already formatted for easier viewing, you do not have to spend time formatting them yourself. If you want to access results from previously run commands, you can do so from the log.

Logs

You can also use the log as a reporting tool. The log records and displays the commands issued, the tables used, and the results obtained during your data analysis. You can easily copy or export log items to other applications.

Project contents

The Print Project Contents option is a valuable tool for compiling supporting documentation. It lets you print logs, tables, views, indexes, workspaces, and preferences, all from the same dialog box.

Record notes

Record notes allow you to attach notes to individual records from within *ACL*. These notes can be edited, and can be used to store whatever information you find useful.

Notes:

Creating a Report from a View

All *ACL* reports are based on views. More than one view can be created for each table, letting you create different presentations of the data.

This section shows you how to:

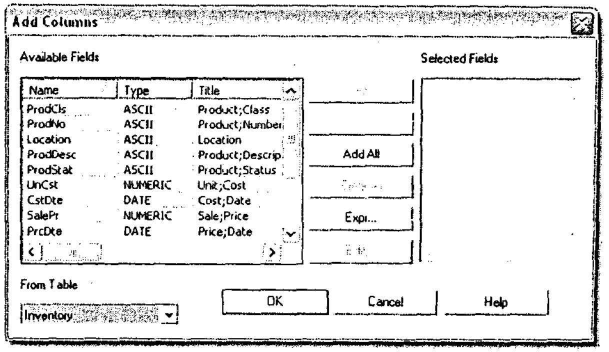
- Create a new view
- Format a view
- Design the report layout
- Print a report

Creating New Views

The first time you open a table, *ACL* creates a default view that contains all the fields in the table layout. You may, however, want to base a report on a new view containing only a selection of the fields pertinent to the report you are creating.

To create a new view:

1. Right-click the **Default_View** tab at the bottom of the view and select **New**.
2. Name the new view and click **OK**.
3. In the **Available Fields** list, double-click the fields you want to add and click **OK**.



Name	Type	Title
ProdCls	ASCII	Product;Class
ProdNo	ASCII	Product;Number
Location	ASCII	Location
ProdDesc	ASCII	Product;Descrip.
ProdStat	ASCII	Product;Status
UnCst	NUMERIC	Unit;Cost
CstDte	DATE	Cost;Date
SalePr	NUMERIC	Sale;Price
PrcDte	DATE	Price;Date

From Table
Inventory

fig. 6-1 Add Columns dialog box

The new view opens in the display area.

Formatting Views

You can change a view to produce a report that contains just the information you want, presented in the way you want it.

ⓘ *Each column in a view represents a defined field or expression.*

Add a column

1. Click the column heading to the right of where you want to add a column, right-click, and select **Add Columns**.
2. Add the fields you want to include to the **Selected Fields** list and click **OK**.

   If you are adding more than one column, add them to the **Selected Fields** list in the order you want them to appear in the view.

Delete a column

- Right-click the column heading, select **Remove Selected Columns**, and click **Remove**.

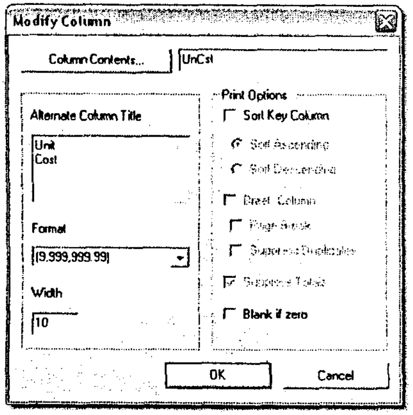
ⓘ *Removing a column from a view does not delete the field from the table.*

Rearrange columns

- Click the column heading and drag it to its new location.

Format columns

- Right-click the column heading and select **Modify Column**.



Column Contents: UnCst

Alternate Column Title: Unit Cost

Format: (9,999,999.99)

Width: 10

Print Options

☐ Sort Key Column
● Sort Ascending
○ Sort Descending
☐ Break Column
☐ Page Break
☐ Suppress Duplicates
☑ Suppress Totals
☐ Blank if zero

fig. 6-2 Modify Column dialog box

The Modify Column dialog box gives you several options for formatting a column. For example, you can change the column heading, modify the column width, or suppress totals.

Change the font

1. Click the Ⓐ (Change Font) button.
2. Select the part of the view you want to change, make the changes, and click **OK**.
3. Click **OK**.

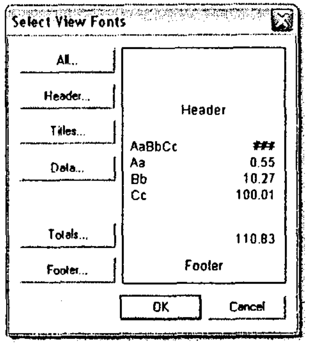


fig. 6-3 Select View Fonts dialog box

ⓘ *You cannot change the font for only a single column, column heading, record, or total. If you change, for example, the font for a column heading, you change the font for all column headings in the view.*

Designing Report Layouts

When you finish formatting a view, you can use the Report dialog box to complete the design of your report.

To open the Report dialog box:

- Select **Data** > **Report**.

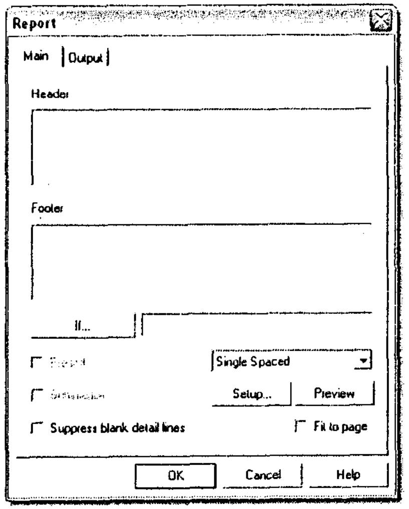


fig. 6-4 Report dialog box

The Report dialog box provides the following formatting options:

Header	If you want a header at the top of your printed report, enter text in the Header box. *ACL* automatically centers the header on the report page.
Footer	If you want a footer at the bottom of your printed report, enter text in the Footer box. *ACL* automatically centers the footer on the report page.
If	You can enter a filter in the If box to limit the records that appear in the report. Because it is a command filter, the filter affects just the contents of the report, not the view.
Presort	If you want to ensure that your report is sorted correctly before printing, select this option. ⓘ For more information see "Creating Sorted Reports" on page 85.
Summarize	Select this option to create a report that contains only subtotals generated on the basis of the break fields selected in the view.
Detail line spacing	By default, reports are single-spaced. Use the drop-down list provided to specify two or three spaces between each detail line.
Suppress blank detail lines	This option lets you create more compact reports by leaving out blank lines.
Fit to page	By default, *ACL* prints data only to the left of the page width indicator (vertical dotted line) in the view. Select this option if you want data appearing to the right of the page width indicator to be included when you print your report. ⓘ Another way to make the data fit the page width is to modify the view.

Printing Reports

Before printing your report, you must specify print and output options. You need to also preview the report so that you can make any necessary adjustments before printing.

Page setup

To specify page orientation, page size, and paper source:

- Select **Data** > **Report** and click **Setup**.

Page margins

ACL uses default margin settings for all printed views and reports.

To change the default margins:

- Select **Tools** > **Options** and click the **Print** tab.

Output

You can send a report to print, to the screen, or to a file.

If you select the File option, the report is sent by default to an ASCII text file. Alternatively, you can send the report to an HTML text file by selecting the option from the File Type drop-down list. This lets you prepare your *ACL* report to be distributed or published via the Internet.

To select an output option:

- In the Report dialog box, select the **Output** tab.

Print preview

Before printing, you can preview a report:

- In the Report dialog box, click **Preview.**

ⓘ *Print preview is not available when you select the Screen output option.*

Print

When you are ready to print:

- In the Report dialog box, click **OK.**

ⓘ *By default, ACL includes the report history when you print a report. To change this default setting, select Tools > Options and click the Print tab.*

Notes:

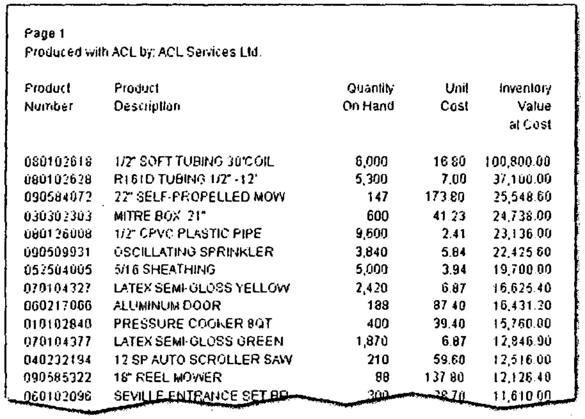
Advanced Reporting

This section shows you how to take advantage of advanced reporting features to create sorted reports, reports that contain subtotals, and multiline reports.

Creating Sorted Reports

A sorted report lets you display records in a particular order based on key fields that you specify.

The following procedure shows you how to create a report sorted by inventory value in descending order. The report appears as follows:



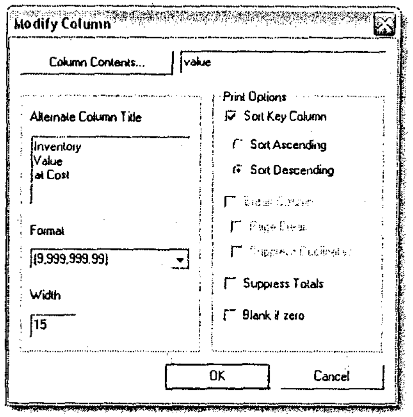
Page 1
Produced with ACL by: ACL Services Ltd.

Product Number	Product Description	Quantity On Hand	Unit Cost	Inventory Value at Cost
080102618	1/2" SOFT TUBING 30'COIL	6,000	16.80	100,800.00
080102628	RIGID TUBING 1/2" -12'	5,300	7.00	37,100.00
090584072	22" SELF-PROPELLED MOW	147	173.80	25,548.60
030302303	MITRE BOX 21"	600	41.23	24,738.00
080126008	1/2" CPVC PLASTIC PIPE	9,600	2.41	23,136.00
090509931	OSCILLATING SPRINKLER	3,840	5.84	22,425.60
052504005	5/16 SHEATHING	5,000	3.94	19,700.00
070104327	LATEX SEMI-GLOSS YELLOW	2,420	6.87	16,625.40
060217066	ALUMINUM DOOR	188	87.40	16,431.20
010102840	PRESSURE COOKER 8QT	400	39.40	15,760.00
070104377	LATEX SEMI-GLOSS GREEN	1,870	6.87	12,846.90
040232194	12 SP AUTO SCROLLER SAW	210	59.60	12,516.00
090585322	16" REEL MOWER	88	137.80	12,126.40
060102096	SEVILLE ENTRANCE SET BR	300	38.70	11,610.00

fig. 6-5 Sorted report example - inventory value descending

To create a sorted report:

1. Click the Inventory Value at Cost column heading, right-click, and select **Modify Columns**.
2. Select the **Sort Key Column** check box, select the **Sort Ascending** option, and click **OK**.



Modify Column

Column Contents... value

Alternate Column Title: Inventory Value at Cost

Format: (9,999,999.99)

Width: 15

Print Options

☑ Sort Key Column
○ Sort Ascending
● Sort Descending
□ Break Column
□ Page Break
□ Suppress Duplicates
□ Suppress Totals
□ Blank if zero

fig. 6-6 Modify Column dialog box

3. Select **Data** > **Report**, select the **Presort** check box and click **OK**.

ⓘ *If you specify more than one sort key column, the order of the fields from left to right determines the order in which they are sorted.*

Creating Reports That Contain Subtotals

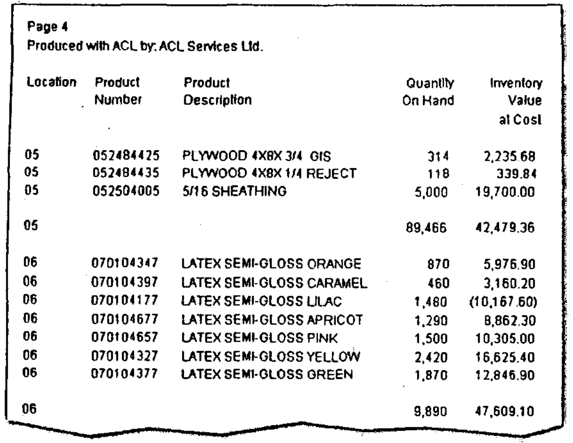
If you format a column as a break column, *ACL* creates a subtotal line for all numeric columns in a report each time the value in the break column changes. Any sort key column containing character data can be formatted as a break column. The column, however, must first be moved to the left-most position in the view.

When you select the Break Column option, you are given two further options:

Page Break	Select this option to insert a page break each time the field value in a break column changes.
Suppress Duplicates	Select this option to suppress the printing of duplicate key fields.

The following procedure shows you how to specify the Location column as a break column to produce a report with subtotals.

The report appears as follows:



```
Page 4
Produced with ACL by: ACL Services Ltd.

Location  Product     Product                       Quantity   Inventory
          Number      Description                   On Hand        Value
                                                                 at Cost

05        052484425   PLYWOOD 4X8X 3/4  GIS              314    2,235.68
05        052484435   PLYWOOD 4X8X 1/4 REJECT            118      339.84
05        052504005   5/16 SHEATHING                   5,000   19,700.00

05                                                    89,466   42,479.36

06        070104347   LATEX SEMI-GLOSS ORANGE            870    5,976.90
06        070104397   LATEX SEMI-GLOSS CARAMEL           460    3,160.20
06        070104177   LATEX SEMI-GLOSS LILAC           1,480  (10,167.60)
06        070104677   LATEX SEMI-GLOSS APRICOT         1,290    8,862.30
06        070104657   LATEX SEMI-GLOSS PINK            1,500   10,305.00
06        070104327   LATEX SEMI-GLOSS YELLOW          2,420   16,625.40
06        070104377   LATEX SEMI-GLOSS GREEN           1,870   12,846.90

06                                                     9,890   47,609.10
```

fig. 6-7 Subtotal Columns example

Specify a break column

1. Place the Location column to the left-most position in the view.
2. Drag the break bar to the right side of the Location column.
3. Select **Data** > **Report**, select the **Presort** check box, and click **OK**.

ⓘ *You can specify more than one column as a break column to produce a report similar to the results of the Cross-tabulate command.*

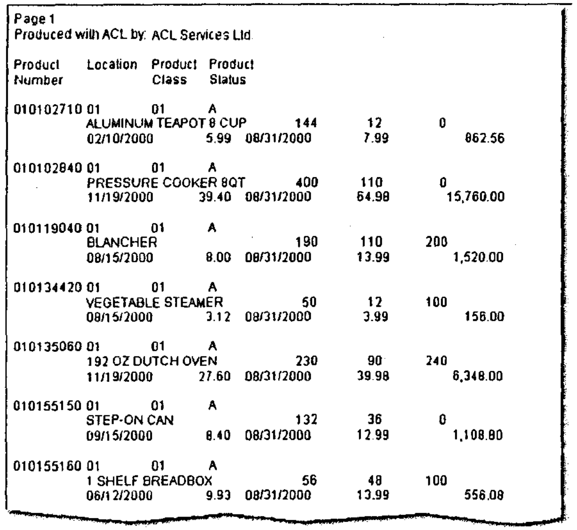
Notes:

Creating Multiline Reports

ACL lets you set up multiline views for your report. The following procedures show you how to create a multiline inventory report. In this procedure, you:

- Add or delete rows below existing records
- Add or delete rows above existing records
- Move column data to new rows
- Suppress column headings

The report appears as follows:



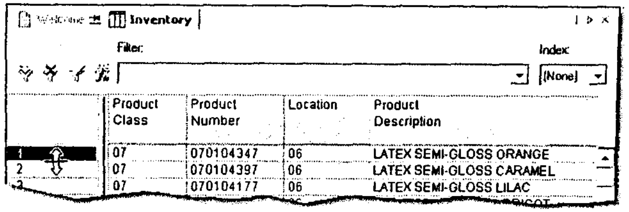
```
Page 1
Produced with ACL by: ACL Services Ltd

Product    Location Product Product
Number              Class   Status

010102710 01        01      A
          ALUMINUM TEAPOT 8 CUP      144       12        0
          02/10/2000       5.99  08/31/2000    7.99               862.56

010102840 01        01      A
          PRESSURE COOKER 8QT        400      110        0
          11/19/2000      39.40  08/31/2000   64.98             15,760.00

010119040 01        01      A
          BLANCHER                   190      110      200
          08/15/2000       8.00  08/31/2000   13.99              1,520.00

010134420 01        01      A
          VEGETABLE STEAMER           50       12      100
          08/15/2000       3.12  08/31/2000    3.99               156.00

010135060 01        01      A
          192 OZ DUTCH OVEN          230       90      240
          11/19/2000      27.60  08/31/2000   39.98              6,348.00

010155150 01        01      A
          STEP-ON CAN                132       36        0
          09/15/2000       8.40  08/31/2000   12.99              1,108.80

010155160 01        01      A
          1 SHELF BREADBOX            56       48      100
          06/12/2000       9.93  08/31/2000   13.99               556.08
```

fig. 6-8 Break Column example

Add or delete rows below existing records

To add or delete rows below the existing ones, click and drag any horizontal line in the record number column as follows:

- To add rows below the column heading and data cell areas, drag downwards.
- To delete rows you added previously, drag upwards. You can delete only blank rows.



	Product Class	Product Number	Location	Product Description
1	07	070104347	06	LATEX SEMI-GLOSS ORANGE
2	07	070104397	06	LATEX SEMI-GLOSS CARAMEL
3	07	070104177	06	LATEX SEMI-GLOSS LILAC

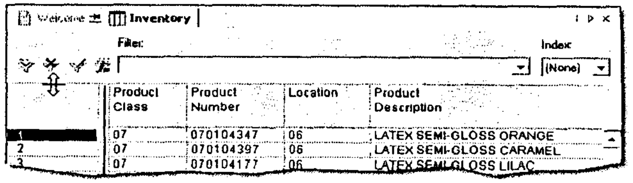
fig. 6-9 Adding and deleting rows below

ⓘ *You cannot add or delete rows separately to heading and data cell areas, but you can suppress the printing of column heading rows as described later in this section.*

Add or delete rows above existing records

To add or delete rows above the existing ones, click and drag the top line in the record number column as follows:

- Drag downwards to add rows above the column heading and data cell areas.
- Drag upwards to delete rows you added previously. You can delete only blank rows.



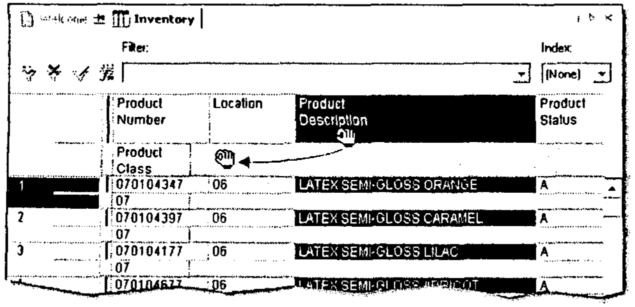
	Product Class	Product Number	Location	Product Description
1	07	070104347	06	LATEX SEMI-GLOSS ORANGE
2	07	070104397	06	LATEX SEMI-GLOSS CARAMEL
3	07	070104177	06	LATEX SEMI-GLOSS LILAC

fig. 6-10 Adding and deleting rows above

Move column data to new rows

- Click a column heading and drag it to a new row.

  When you release the mouse button, the column heading snaps to the left-most side of the row.

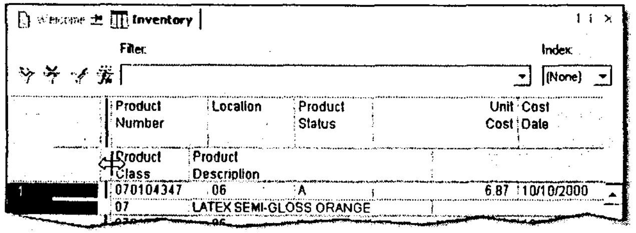


	Product Number	Location	Product Description	Product Status
	Product Class			
1	070104347	06	LATEX SEMI-GLOSS ORANGE	A
	07			
2	070104397	06	LATEX SEMI-GLOSS CARAMEL	A
	07			
3	070104177	06	LATEX SEMI-GLOSS LILAC	A
	07			
	070104677	06	LATEX SEMI-GLOSS [illegible]	A

fig. 6-11 Moving column data to new rows

Indent column rows

- Drag the column divider on the left side of the column to the right.



	Product Number	Location	Product Status	Unit Cost	Cost Date
	Product Class	Product Description			
1	070104347	06	A	6.87	10/10/2000
	07	LATEX SEMI-GLOSS ORANGE			

fig. 6-12 Indenting column rows

Suppress headings

When you first create multiline reports, you create multiline column headings as well. If you do not want multiline column headings, you can suppress them as follows:

- Drag the horizontal line between the column headings and the vertical scroll bar upwards to cover the column headings that you do not want to print.

  Note that you can suppress all but one of the column heading rows. The remaining row can be blank.

Notes:

Generating Reports with Crystal Reports

You can take advantage of the reporting capabilities of Crystal Reports if you install ACL Runtime Components. You can generate, view, and print standardized reports in *ACL* using custom templates designed in Crystal Reports.

ⓘ *Install ACL Runtime Components from the ACL CD-ROM or from the Utilities drop-down list at www.acl.com/downloads.*

Creating Crystal Report Templates

A report template is created in two parts:

- Create a blank template using ACL
- Complete the template using Crystal Reports

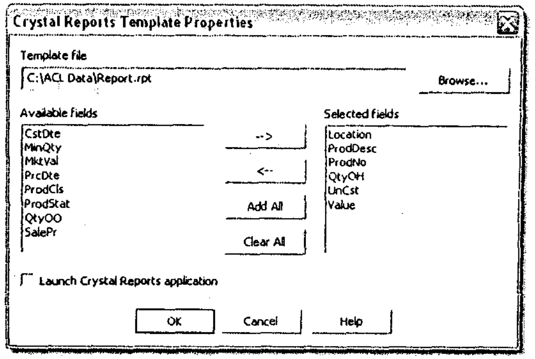
If you have Crystal Reports installed on your workstation, you can customize the template yourself. If you do not have Crystal Reports, find a Crystal Reports user who can complete the template for you.

Create a blank template using ACL

Using *ACL*, you create a template based on an open *ACL* table.

To create a blank Crystal Reports template:

1. Open the *ACL* table for which you want to create a report.
2. Select **Data** > **Crystal Reports** > **Create Template**.



Crystal Reports Template Properties

Template file

C:\ACL Data\Report.rpt

Available fields

CstDte
MinQty
MktVal
PrcDte
ProdCls
ProdStat
QtyOO
SalePr

Selected fields

Location
ProdDesc
ProdNo
QtyOH
UnCst
Value

□ Launch Crystal Reports application

fig. 6-13 Crystal Reports Template Properties dialog box

The Crystal Reports Template Properties dialog box opens.

3. Click **Browse.**

   The Crystal Report dialog box opens.

4. Navigate to the location where you want to save the report template, name the template, and click **Open**.
5. Add the fields you want in the report to the **Selected fields** list and click **OK**.

   ACL creates a template with an .rpt extension. The template is blank, but contains embedded field information. The Crystal Reports application is required to complete the template.

Complete the template using Crystal Reports

In order to complete the report you created, you must have Crystal Reports version XI or later.

To complete the template in Crystal Reports:

1. In Crystal Reports, open the report template.

2. Insert the fields from the **Database Fields** list into the template.
3. Format the template.

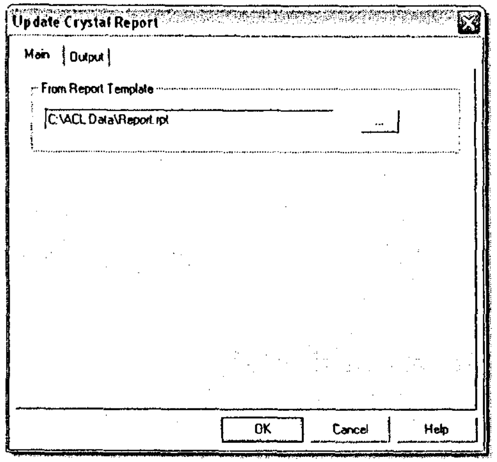
ⓘ *Use the template only with tables where field names and field types match those in the template. There can, however, be more fields in the table than in the template.*

Generating Reports Using Custom Templates

When the report template is complete, you can update the template with *ACL* data.

Update the template

1. Open a table and select **Data** > **Crystal Reports** > **Update Template.**


Update Crystal Report

From Report Template: C:\ACL Data\Report.rpt

fig. 6-14 Update Crystal Report dialog box

2. Browse to the report template file you want and click **OK.**
3. Click the **Output** tab, select an output option, and click **OK.**

View the report

1. Select **Data** > **Crystal Reports** > **View Report.**
2. Browse to the report file and click **Open.**

The Crystal Reports viewer lets you view and print reports. You can also save reports to other file formats, including PDF.

ⓘ *If you send your results to screen after updating a template, the report automatically opens in the Crystal Reports viewer.*

Notes:

Creating Graphs

Graphs are a good way to enhance your reports. They add visual appeal and provide an easily accessible presentation of your results.

In *ACL*, six commands generate graphs automatically. You can also graph data directly from a view. In either case, you can customize a graph using the many formatting options in *ACL*.

Viewing Graphs Created by a Command

The following commands generate graphs automatically:

- Stratify
- Classify
- Histogram
- Age
- Cross-tabulate
- Benford

To view a graph:

▸ Click the **Graph** tab at the bottom of the display area.

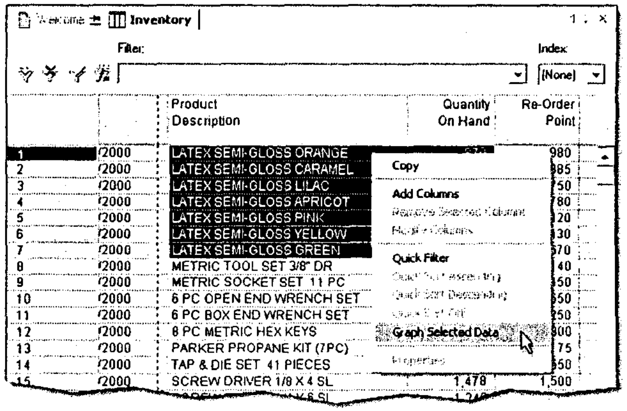
Creating Graphs from a View

You can create a graph from any range of data in a view as long as at least one field is numeric. For ease of comprehension, it is best to restrict the fields selected to a numeric field and a second field that is related to the numeric data.

For example, in the Inventory table, there is a quantity on hand field and a product description field. You can create a graph showing the quantities on hand for the first seven products listed in the product description field.

To create a graph from a view:

1. Drag the quantity on hand field until it is next to the product description field.
2. Highlight records 1–7 in both fields.
3. Right-click the table and select **Graph Selected Data.**



		Product Description	Quantity On Hand	Re-Order Point
1	/2000	LATEX SEMI-GLOSS ORANGE		980
2	/2000	LATEX SEMI-GLOSS CARAMEL		985
3	/2000	LATEX SEMI-GLOSS LILAC		750
4	/2000	LATEX SEMI-GLOSS APRICOT		780
5	/2000	LATEX SEMI-GLOSS PINK		120
6	/2000	LATEX SEMI-GLOSS YELLOW		130
7	/2000	LATEX SEMI-GLOSS GREEN		570
8	/2000	METRIC TOOL SET 3/8" DR		40
9	/2000	METRIC SOCKET SET 11 PC		150
10	/2000	6 PC OPEN END WRENCH SET		550
11	/2000	6 PC BOX END WRENCH SET		250
12	/2000	8 PC METRIC HEX KEYS		300
13	/2000	PARKER PROPANE KIT (7PC)		75
14	/2000	TAP & DIE SET 41 PIECES		550
15	/2000	SCREW DRIVER 1/8 X 4 SL	1,478	1,500

fig. 6-15 Graphing selected data

The resulting graph shows the quantity on hand on the vertical axis and the product description on the horizontal axis. The default graph type is a 3-D bar graph.

Using Graph Options

When creating graphs, *ACL* includes a button bar above the graph for easy access to formatting, printing, storage, and analysis options.

ⓘ *You can also right-click any element in a graph to access a shortcut menu with the same formatting options as the button bar.*

Formatting graphs

The following options let you change the appearance of your graphs.

Option	Icon	Description
Graph Type		While the default format is a 3-D bar graph, you can change the graph to any of the many options available. Options not suitable for the data upon which the graph is based are grayed out. **Note:** You can separate the sections of a pie graph by right-clicking on a section and selecting Explode Pie.
Graph Properties		Format the font, background, and frame, and select the elements you want displayed in the graph.
Legend Properties		Format the attributes and font, and select which data series is shown in the legend.
Axis Properties		Format the style, scale, and font of the vertical axis.
Format Data		Format the horizontal axis labels and the appearance of the data content, and select the fields you want displayed in the graph.
Label Properties	aB	Format the orientation, font, and other attributes of labels on the graph. A label must be selected for the button to be active. **Note:** To add a label, click where you want the label to appear, right-click, select Add Label, enter the label, and click OK.
Show/Hide Legend		Make the legend visible or invisible.
Show/Hide Axis		Make the vertical axis visible or invisible.

Incorporating a graph into a report

The following options let you print, save, and copy your graphs.

Option	Icon	Description
Print Graph		To print with maximum resolution, make the graph window as large as possible before you print.
Save Graph as Bitmap		A bitmap is a file format that can be imported into almost any document.
Copy Graph to Clipboard		Copy the bitmap image to the clipboard for easy pasting into documents.

Exploring the data in a graph

The following options let you change and examine the data that underlies any portion of your graphs.

Option	Icon	Description
Edit Command		If the graph was produced automatically by a command, the Edit Command button opens the dialog box for that command. You can change the parameters underlying the graph.
Drill-down		Select any section of a bar, pie, or line graph and double-click it to view the records upon which that section is based. A filter selecting the relevant records is imposed on the data.

Using the Log

The log is a chronological record of your sessions on a project. As described in “Logs” on page 5, the log automatically records the steps you take in your project, letting you retrace your work and retrieve results for a report.

To view the log:

- In the Project Navigator, click the **Log** tab.

Adding and Viewing Log Sessions

A session is a record of your work on a project. When you open a project, a new session is created. By separating your analysis into discrete parts, sessions help you organize your work.

During a session, *ACL* automatically records all executed commands and results in the log. To help you track the purpose of your actions and the meaning of your results, you can also add comments.

By default, sessions are named with a time stamp. You can start a new session at any time and give it a different name.

To add a new session:

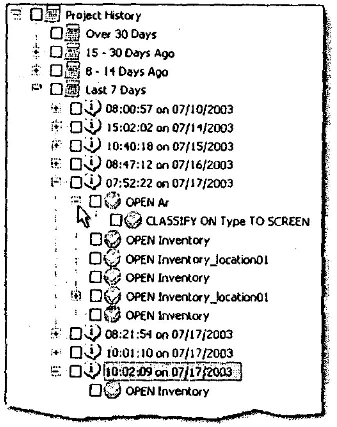
1. Select **Tools > Add New Session.**
2. Enter a name.
3. Click **OK.**

   A new session is created.

To view a session:

1. Click the **Log** tab.

   The log opens.



```
Project History
    Over 30 Days
    15 - 30 Days Ago
    8 - 14 Days Ago
    Last 7 Days
        08:00:57 on 07/10/2003
        15:02:02 on 07/14/2003
        10:40:18 on 07/15/2003
        08:47:12 on 07/16/2003
        07:52:22 on 07/17/2003
            OPEN Ar
                CLASSIFY ON Type TO SCREEN
            OPEN Inventory
            OPEN Inventory_location01
            OPEN Inventory
            OPEN Inventory_location01
            OPEN Inventory
        08:21:54 on 07/17/2003
        10:01:10 on 07/17/2003
        10:02:09 on 07/17/2003
            OPEN Inventory
```

fig. 6-16 Log tab

2. Choose from the following:
   - To view a list of the tables that were opened during the session, click the plus sign beside a session.
   - To view a list of the commands that were executed on the table, click the plus sign beside an OPEN <table> entry.
   - To view a command’s details in the display area, double-click the command.

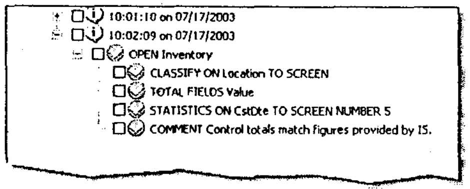
Adding Comments

Comments help you and others track the purpose of the steps you take and the meaning of the results obtained.

ⓘ *Add a comment immediately after you run a command. This ensures that the comment appears directly below the command.*

To add a comment:

1. Select **Tools** > **Add Comment**.
2. Type your comment.
3. Click **OK**.



```
□ 10:01:10 on 07/17/2003
□ 10:02:09 on 07/17/2003
  □ OPEN Inventory
    □ CLASSIFY ON Location TO SCREEN
    □ TOTAL FIELDS Value
    □ STATISTICS ON CstDte TO SCREEN NUMBER 5
    □ COMMENT Control totals match figures provided by IS.
```

fig. 6-17 Adding comments

ⓘ *Double-click a comment to view it in the display area.*

Exporting from the Log

You can use the log as a source for your reports. You can export selected entries from the log to:

- HTML files
- Log files
- Scripts
- WordPad
- Text files

To export from the log:

1. Click the **Log** tab.
2. Select the check boxes of the sessions and commands that you want to export.
3. Right-click in the Project Navigator, select **Save Selected Items**, and select a format.
4. Name the file and click **Save**.

Copying and Pasting Results

You can copy and paste results into Word and Excel, and most other text and spreadsheet programs. Word and Excel retain the formatting of the content.

Copy and paste results from the log

1. Click the **Log** tab.
2. Select a command by clicking the box to its left, right-click, and select **Copy**.

   The command can now be pasted into another application.

Copy and paste results from the results tab

1. Click the **Log** tab.
2. Double-click the command whose results you want to copy.
3. Right-click in the display area and select **Select All**.
4. Right-click the display area and select **Copy**.

   The command can now be pasted into another application.

ⓘ *To select just a portion of the results tab, highlight the portion you want, right-click, and select Copy.*

Searching the Log

In long or complex analyses, particular commands or results may be difficult to locate. In these situations, you can search the log.

To search the log:

1. Click the **Log** tab, right-click in the Project Navigator, and select **Find**.
2. Enter key words from the content you seek, select the appropriate options, and click **Find Next**.

Documenting Your Analysis

Documenting your analysis is an integral part of any project. In addition to the log, *ACL* lets you keep notes within a project, print any part of a project, and view and print the history of a table.

You can create notes for the following items:

- Projects
- Tables
- Views
- Fields
- Records
- Scripts
- Workspaces

Create Project Notes

Each project has its own area for making notes.

To create a project note:

1. In the **Overview** tab, right-click the project name, and select **Properties**.
2. Click the **Notes** tab, enter a note, and click **OK**.

To print a project's notes, use the Print Project Contents feature described following.

Print Project Contents

ACL makes it easy to print any aspect of a project, including:

- Table layouts
- View definitions
- Script definitions
- Index definitions
- Workspace definitions
- Program options, notes, and the log

To print project contents:

1. Select **File > Print Project Contents.**
2. Select the items that you want to print and click **Print.**

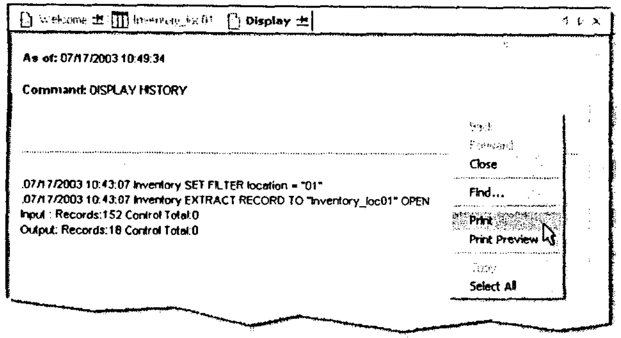
View and Print a Table History

A table history is created when an *ACL* command creates a new table.

For example, when you extract selected fields to a new table, a history is created for that table. This history lists the Extract command, the name of the table the fields were extracted from, control totals, and the date and time the table was created.

To view and print a table history:

1. In the **Overview** tab, double-click the table whose history you want to view.
2. Select **Tools > Table History**.
3. Right-click in the display area and select **Print.**



```
As of: 07/17/2003 10:49:34

Command: DISPLAY HISTORY

.07/17/2003 10:43:07 Inventory SET FILTER location = "01"
.07/17/2003 10:43:07 Inventory EXTRACT RECORD TO "Inventory_loc01" OPEN
Input : Records:152 Control Total:0
Output: Records:18 Control Total:0
```

fig. 6-18 Printing history

Record Notes

ACL allows you to attach a note to a single record. Use this feature for documenting specifics about a particular record. You can create, edit, view, display, and delete record notes.

ⓘ *Notes are not supported in the Client/Server environment when the table is maintained on a server.*

To create a record note:

1. Click the record number of the record you want to add a note to.
2. Select **Edit** > **Notes** > **Edit Note.**
3. Enter a note and click **OK.**

   The record is marked with a pink dot to show a note is associated with the record.

To edit or view a record note:

1. Right-click the appropriate record number.
2. Select **Edit Note.**
3. If required, edit the note and click **OK.**

To display record notes in a view:

1. Right-click a column heading in the view, and select **Add Columns.**

   ACL inserts new column(s) to the left of the highlighted column.

2. In the **Available Fields** list, double-click the RecordNote field and click **OK.**

ⓘ *The RecordNote field is only available if you have already created at least one record note.*

To hide record notes in a view:

1. Right-click the RecordNote column heading and select **Remove Selected Columns.**
2. Click **Remove.**

   Record notes are no longer displayed in the view, but they can still be viewed by selecting **Edit Note.**

To delete a single record note:

1. Right-click the appropriate record number.
2. Select **Edit Note.**
3. Select all the content in the note and press **Delete.**
4. Click **OK.**

To delete all record notes from a table:

1. Select **Edit** > **Notes** > **Delete All Notes from Table.**
2. Click **OK.**

   ⓘ *Use caution when deleting all notes from a table. You are prevented from deleting all notes from a table if the RecordNote column is currently being displayed.*

Activities

In this Section

To master *ACL* it is necessary to practice with *ACL*. These activities are designed to help you learn to use *ACL* effectively and efficiently.

Fundamentals

Activity 1.1 Navigating in ACL

Project: ACL105 Foundations.acl

Table: Customer

Purpose

This exercise provides a review of the *ACL* user interface including the menu bar, toolbar, and status bar.

Background

You are a new auditor at the ABC Department Store. You have received an *ACL* project entitled ACL105 Foundations.acl that was created by a team member. As a new *ACL* user, your first assignment is to become familiar with this project and to learn how to navigate in *ACL*.

Your tasks

1. What steps are required to open the Customer table in the ACL105 Foundations.acl project?
2. Where do you access commands in *ACL?*
3. Under which menu do you find the Count command?
4. How do you quickly access frequently used commands?
5. What two pieces of information does the status bar always display?
6. What information can be accessed from the **Overview** tab?

Activity 1.2 Fundamentals Review

Project: ACL105 Foundations.acl

Table: Customer

Purpose

This exercise provides a review of *ACL* fundamentals, including *ACL* projects, tables, and views. It is necessary to develop a solid understanding of these concepts because they are the foundation upon which all *ACL* analyses are built.

Background

After reviewing the contents of ACL105 Foundations.acl, you determine that there are a number of tables and views stored in this project. Your audit team has assigned you to focus on the Customer table. Before you begin your analysis of this table, you must review the table layout and understand what types of fields are stored there.

Your tasks

1. What information does the Customer table contain?
2. Where are the field definitions for the Customer table stored?
3. Is the Customer.fil data file stored in the project?
4. How are the data file and table layout related?
5. What information is stored in a table layout?
6. The log file ACL105_Foundations.log is activated when the project ACL105 Foundations.acl is opened. What information is stored in a log file?

Activity 1.3 ACL Projects

Project: ACL105 Foundations.acl

Purpose

This exercise provides practice using *ACL* projects. It shows how to create a new project and import table layouts from an existing project.

Background

You have been assigned to a surprise credit audit, separate from your previous work, and you need to create a new project containing the Customer table from the ACL105 Foundations.acl project.

Your task

- Show how you would organize and separate your work.

Data Access

Activity 2.1 Accessing Flat Files

Project: Class.acl

Data file: Trans.fil

Purpose

This exercise provides practice defining flat files. Through this process, you create a table for working with a data file.

Background

You received a request from the audit manager to profile the Trans data file and identify irregularities in the transaction dates. As a new assignment, you must create a new project named Class and a new table named Trans.

Your task

- Using the following table layout, create the Trans table.

  Record length = 32

Start	Length	Field Name	Data Type	Decimals/Format
1	5	invoice	ASCII	
6	6	date	DATE	YYMMDD
12	2	prodcls	ASCII	
12	9	prodno	ASCII	
21	4	qty	NUMERIC	0
25	8	amount	NUMERIC	2

Activity 2.2 Copying and Linking Tables

Project: ACL105 Foundations.acl

Table: Inventory

Purpose

This exercise provides practice using copying and linking options with tables.

Background

The audit department downloads inventory data from the mainframe at the end of each month. Rather than use the Data Definition Wizard to manually create a table layout for the most recent inventory file, you want to reuse an existing table layout.

Your tasks

1. Copy the Inventory table and rename it Feb_invntry.
2. Link Feb_invntry to the Month2_invntry.fil data file.

Data Integrity Verification

Activity 3.1 Count, Total, and Statistics

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice using the Count, Total, and Statistics commands. It shows you how to test for data completeness and how to profile the data to determine average, maximum, and minimum values.

Background

You have downloaded and defined the Trans file for analysis. Also, you received a report from the IT group that displays control total details. According to the IT report, the total number of records is 339 and the total sum of invoice amounts is $300,682.04. You need to verify the integrity of the file and answer a few additional questions about the details listed in the file.

Your tasks

1. Determine if the Trans table is well defined and complete.
2. What is the value of the largest credit?
3. Credits issued must not exceed $1,000. Verify if there are any transactions over this limit.
4. Determine the total value of transactions over the credit limit.

Challenge question

- There may be transactions with invoice amounts of $0.00. Identify these types of transactions. How many are there?

Activity 3.2 Gaps and Duplicates

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice using the Gaps and Duplicates commands. It shows how to scan a table for irregularities such as missing items and duplicates. This is an essential step when using tables for the first time.

Background

Now that the Trans table has been checked for completeness with respect to record numbers and totals, you must determine if any invoices are missing or duplicated. As invoices are serially numbered, you can perform gap and duplicate testing to test the level of table integrity.

Your tasks

1. Determine if the table is missing any invoice numbers.
2. Locate and identify transactions categorized as possible duplicate invoices.
3. If there are possible duplicate invoices on the file, are these transactions identical?

Data Analysis

Activity 4.1 Filters

Project: ACL105 Foundations.acl

Table: Demo

Purpose

This exercise provides practice creating filters. It shows you how to view transactions that meet specific search criteria or tests.

Background

As a member of the audit team, you have been asked to profile the journal entries table, Demo, and comment on the use of a suspense batch account. It is believed that edit checks on data entry have not eliminated the use of "catch-all" batch accounts assigned code 99999. It is your task to quantify the impact of this practice on your organization.

Your tasks

1. Are any batch accounts still being assigned code 99999? If so, how many?
2. Are any GL110 accounts affected by this practice?

Challenge question

- Comment on the number of problem transactions that exceed the $100,000 threshold amount. Include batch 99999 and GL110 transactions in your analysis.

Activity 4.2 Computed Fields and Exception Reporting

Project: ACL105 Foundations.acl

Table: Badfile.fil

Background

You have been asked to determine the total value of billings. Before analyzing the table, you must use data integrity verification techniques to confirm the following:

- The invoiceno and prodno fields are populated
- All invoice numbers begin with 214
- Order quantities are less than 1000 and are positive values
- All prices are positive values

Your tasks

1. Create a computed field to identify records with at least one integrity problem.
2. The materiality threshold for computed data errors is $0.50 per transaction. Create a computed field to test related data fields for errors greater than this threshold.

Activity 4.3 Summarize

Project: ACL105 Foundations.acl

Table: Demo

Purpose

This exercise provides practice using the Summarize command. It shows how to create a summary report based on a key field. This command produces a summarization on a character field regardless of the number of unique key field items in the table.

Background

Your audit team has assigned you to profile the Demo journal entries table. Through other printed reports, you determine that activity in a number of branches has resulted in net credits during the fourth quarter period.

Your tasks

1. Identify branches with net credits.
2. Identify the branches with the five largest credits and save results in a filter.
3. Are there any branches over the $500,000 credit threshold?

Challenge question

- Which branch had the highest number of daily transactions in a single day during the period? What was the net amount for that branch?

Activity 4.4 Classify, Stratify, and Age

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice using the Classify, Stratify, and Age commands. It shows how to produce summary reports on character, numeric, and date fields. These reports also let you determine levels of materiality.

These commands are useful for grouping fields within certain ranges or aging periods.

Background

The manager of the marketing department at the ABC Department Store would like to know which amount range is the most popular. To do that, you need to profile the Trans table in greater detail by concentrating on the products sold.

Your tasks are to identify the best selling product based on the number of units sold and to identify the most popular invoice amount range. You are also required to identify the quarter with the highest revenue.

Your tasks

1. Examine the table and determine which product is the best seller based on quantities sold.
2. Based on default ranges, into which range do most invoice amounts fall? Eliminate any invoices with net credit balances from your analysis.
3. Which quarter of the calendar year 2000 produced the most revenue? Use the aging periods 0, 91, 182, 273, and 366.

Activity 4.5 Extract

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice using the Extract command. It shows how to create a table by extracting data from a master table. Use this command when you want to profile a subset of records.

Background

You want to examine transactions from product classes 04 and 09 to determine if any invoice amounts are unusually high.

To isolate these transactions, you can choose from the following solutions:

- Create a filter to work with these transactions. This works well for smaller tables, with no degradation in processing speed.
- If the table has a large number of records, extract those transactions that meet your test to another table for further processing. This smaller table processes more quickly than the larger parent table and eliminates the need to activate a filter. The downside to extracting is that it creates additional tables that occupy valuable hard disk space.

The ABC Department Store recently purchased a new 500 GB hard disk for the Audit LAN server, so disk space is not an issue.

Your task

- ▸ Create a table that contains only transactions from product classes 04 and 09. Name it Trans_extract_class0409.

Activity 4.6 Export

Project: ACL105 Foundations.acl

Table: Trans_extract_class0409

Purpose

This exercise provides practice using the Export command. It shows how to produce an exported file that can be used directly by other software applications.

With an exported file, you can merge information into a word-processed document, produce high-quality presentation graphics, or provide enhanced reporting.

Background

You want to export the results from the exercise in Activity 4.5 Extract to an Excel spreadsheet for analysis by a colleague who does not use *ACL*.

Your task

- ▸ Generate an Excel file and verify its completeness.

Activity 4.7 Index and Sort

Project: ACL105 Foundations.acl

Table: Customer

Purpose

This exercise provides practice using the Index and Sort commands. It also shows how to reorder a table on key fields.

Use indexed and sorted tables when you need a table to be in ascending or descending order based on key fields.

Background

The ABC Department Store is reviewing their customer tables. Customers with credit limits over $25,000 are to be re-evaluated for a possible increase. Upper management has asked the audit department to generate a hard copy report that profiles these customers based on their credit limits.

Your tasks

1. Reorder the Customer table by credit limit in descending value.
2. Create a table that meets the test "credit limit over $25,000". Ensure that the table is in descending order by credit limit.

Activity 4.8 Extract Using the Append option

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice using the Extract command with the Append option. It shows how to produce a master table from two tables that have the same structure.

This exercise also shows how to use this technique to combine two tables. For example, you can create a year-to-date table that is appended to the end of each month. You can also select transactions from two different tables with the same structure and append them to a new table.

Background

In the audit department, certain tests are executed on credit transactions for each branch location. The credit transactions for product classes 01 and 07 need to be jointly summarized and forwarded to upper management for review.

Your task

- Extract transactions for product class 01 to one table and transactions for product class 07 to another table. Combine these two new tables into one table named Trans_extract_class_0107.

Activity 4.9 Join (Basic)

Project: ACL105 Foundations.acl

Tables: Empmast, Payroll

Purpose

This exercise provides practice using the Join command. It shows how to compare information from two separate tables and how to create a new table containing information from both tables.

Background

The Human Resources Administrator has had complaints from a couple of employees who say they have been underpaid. The Human Resources Administrator wants you to analyze the payroll records to identify all underpaid employees.

All employee information is recorded in an employee master table, Empmast. This table contains all of the usual employee information, including employee names, addresses, annual salaries, and period salaries.

Payroll transactions are generated from the payroll system monthly and are written to a table named Payroll. This table includes employee numbers and gross pay for the period.

Both tables are defined in the ACL105 Foundations.acl project.

Your task

- Identify underpaid employees. There are three of them. Of these, one was not paid at all.

Empmast.fil

Record length = 236

Start	Length	Field	Type	Decimals/Format
1	6	empno	EBCDIC	
7	25	first	EBCDIC	
32	25	last	EBCDIC	
57	35	address	EBCDIC	
92	35	city	EBCDIC	
127	25	state_ province	EBCDIC	
152	25	country	EBCDIC	
177	2	code	EBCDIC	
179	3	workdept	EBCDIC	
182	4	phoneno	EBCDIC	
186	10	hiredate	DATE	MM/DD/YYYY
196	8	job	EBCDIC	
204	2	edlevel	BINARY	0
206	1	sex	EBCDIC	
207	10	birthdate	DATE	MM/DD/YYYY
217	5	salary	PACKED	2
222	5	bonus	PACKED	2
227	5	comm	PACKED	2
232	5	pay_per_period	PACKED	2

Payroll.fil

Record length = 47

Start	Length	Field	Type	Decimals/Format
1	6	empno	EBCDIC	
7	9	gross_pay	PACKED	2
16	9	tax_amount	PACKED	2
25	9	net_pay	PACKED	2
34	3	work_dept	EBCDIC	
37	6	pay_date	DATE	DDMMYY
43	5	cheque_no	EBCDIC	

Activity 4.10 Join (Intermediate)

Project: ACL105 Foundations.acl

Tables: Empmast, Payroll

Purpose

This exercise provides practice using the Join command. It shows how to join two separate tables so that you can use the data in a single table.

Background

The Human Resources Administrator has had complaints from a couple of employees who say they were underpaid. Also, last month's payroll is several thousand dollars higher than expected. You suspect that other employees may have been overpaid by mistake. The Human Resources Administrator wants you to use your *ACL* skills to analyze the payroll records and to identify all potential payroll problems.

All employee information is recorded in an employee master table, Empmast. This table records all of the usual employee information, including employee names, addresses, annual salaries, and period salaries.

Payroll transactions are generated from the payroll system monthly and are written to a table named Payroll. This table includes employee numbers and gross pay for the period.

Both tables are defined in the ACL105 Foundations.acl project.

Your tasks

1. Confirm that the difference between these two tables for this month is $3,284.82.
2. Identify the following exceptions by using only one join:
   - Valid employees who were paid an incorrect amount
   - Valid employees who were not paid
   - Payroll payments that are not accounted for by matching master table records
3. Identify employees who were paid more than one time.

   Ensure that all identified discrepancies account for the full amount of the difference found in task 1.

Activity 4.11 Relations (Basic)

Project: ACL105 Foundations.acl

Tables: Ap_trans, Vendor, Inventory

Purpose

This exercise provides practice using the Relations command. It shows how to combine information from three tables into one logical table for easier processing.

Setting up relations between tables is a common database practice. This exercise shows how to use the *ACL* version of this technique, the Relations command.

Background

You are preparing a report on the outstanding accounts payable transactions contained in the Ap_trans table. To increase the readability of the report, you want to include the vendor name with each vendor number. You also want to include the product description with the product number for each item purchased.

To include the vendor name and the product description in your report, you need to access the vendor table and the inventory table.

The three tables that you need to use are defined within the ACL105 Foundations.acl project.

Your tasks

1. Relate the Ap_trans table to the Vendor and Inventory tables.
2. In the Ap_trans default view, add the vendor name and product description columns next to the vendor number and product number columns respectively.

Activity 4.12 Relations (Intermediate)

Project: ACL105 Foundations.acl

Tables: Ap_trans, Inventory, Dept

Purpose

This exercise provides practice building direct and indirect relations between tables.

Background

You are preparing a report on accounts payable transactions that you received in a table named Ap_trans. You want to include the product description and the department name in your report.

The product description for each item purchased is contained in the Inventory table. The department name is contained in the Dept table. To access this information, you need to build relations between the Ap_trans table and the Inventory and Dept tables.

Unfortunately, there is no direct relation between the Ap_trans table and the Dept table. As a result, you need to gain access to this information in an indirect manner through the Inventory table.

Your tasks

1. Open the existing ACL105 Foundations.acl project and relate the Ap_trans table to the Inventory and Dept tables.
2. In the Ap_trans default view, add the department description column next to the vendor number column.

ⓘ *The product class field in the Inventory table matches the department code field in the Dept table.*

Reporting Results

Activity 5.1 Graphing

Project: ACL105 Foundations.acl

Table: Trans

Purpose

This exercise provides practice creating graphs with different commands. It shows how to create visually appealing reports by change graph properties and types.

Background

The ABC Department Store is projecting inventory orders for the upcoming season. To support their projections, they request a product sales report that shows total sales by product class. To enhance the product class report, create a graph to show the results.

Your tasks

1. Create a bar graph showing the invoice amounts for each product class.
2. Change the graph type to a 3-D pie chart.

Solutions

In this Section

Fundamentals

Data Access

Data Integrity Verification

Data Analysis

Reporting Results

These solutions present the most effective and efficient ways to solve the problems presented in the Activities section.

Fundamentals

Activity 1.1 Navigating in ACL

Project: ACL105 Foundations.acl

Table: Customer

Your tasks

1. What steps are required to open the Customer table in the ACL105 Foundations.acl project?
2. Where do you access commands in *ACL*?
3. Under which menu do you find the Count command?
4. How do you quickly access frequently used commands?
5. What two pieces of information does the status bar always display?
6. What information can be accessed from the **Overview** tab?

Solution 1

1. Select **File** > **Open Project**, select ACL105 Foundations.acl, and click **Open.**
2. In the Project Navigator, click the **Overview** tab, click the plus sign to the left of the Accounts_receivable_audit folder, and double-click Customer.

Solution 2

- Most commands are stored on the menu bar under Data, Analyze, and Sampling.

Solution 3

- The Count command is found under Analyze.

Solution 4

- Commands are quickly accessed from the toolbar.

Solution 5

- The status bar always displays the table name and number of records.

Solution 6

- The **Overview** tab displays a list of project contents, including scripts, tables, and workspaces.

Activity 1.2 Fundamentals Review

Project: ACL105 Foundations.acl

Table: Customer

Your tasks

1. What information does the Customer table contain?
2. Where are the field definitions for the Customer table stored?
3. Is the Customer.fil data file stored in the project?
4. How are the data file and table layout related?
5. What information is stored in a table layout?
6. The log file ACL105_Foundations.log is activated when the project ACL105 Foundations.acl is opened. What information is stored in a log file?

Solution 1

▸ The Customer table contains the organization's customer information, including customer identification number, customer name, address, and phone number.

Solution 2

▸ Field definitions for a table are stored in the table layout.

Solution 3

▸ The Customer.fil data file is not stored in the project.

Solution 4

▸ The data file and the table are related through a linking process. This link is made automatically when the file is defined for the first time.

Solution 5

▸ The following information is stored in a table layout:

- File types
- Record lengths
- Field names
- Data types
- Field start positions and lengths
- The number of decimal places
- Field formatting
- Column headings

Solution 6

▸ The information that is stored in a log file includes all the commands that are issued during an *ACL* session and the results of those commands.

ⓘ *To ensure the log does not become too large over time, regularly create a backup and start a new log.*

Activity 1.3 ACL Projects

Project: ACL105 Foundations.acl

Your task

- You have been assigned to a surprise credit audit, separate from your previous work, and you need to create a new project containing the Customer table from the ACL105 Foundations.acl project. Show how you would organize and separate your work.

Solution

Create a new project:

1. Select **File** > **New** > **Project**, name the new project `Credit_audit`, and click **Save**.
2. To close the Data Definition Wizard, click **Cancel**.
3. Click the **Overview** tab.
4. Right-click the Credit_audit.acl folder, select **Copy from Another Project**, and select **Table**.
5. In the Locate Project File dialog box, double-click ACL105 Foundations.acl.
6. In the Import dialog box, double-click Customer and click **OK**.

Data Access

Activity 2.1 Accessing Flat Files

Project: Class.acl

Data file: Trans.fil

Your task

- Using the following table layout, create the Trans table.

  Record length = 32

Start	Length	Field Name	Data Type	Decimals/Format
1	5	invoice	ASCII	
6	6	date	DATE	YYMMDD
12	2	prodcls	ASCII	
12	9	prodno	ASCII	
21	4	qty	NUMERIC	0
25	8	amount	NUMERIC	2

Solution

Create a new project and select a flat file:

1. Select **File** > **New** > **Project**, name the new project `Class`, and click **Save**.

   The Data Definition Wizard starts automatically.

2. Click **Next**.

   By default, the Data Definition Wizard selects Disk as the data source.

3. Click **Next**.
4. In the Select File to Define dialog box, select Trans.fil and click **Open**.

   The Data Definition Wizard correctly identifies the character set as ASCII.

5. Click **Next**.

   The Data Definition Wizard correctly identifies the file format as Other.

6. Click **Next**.

   In the File Properties window, the Data Definition Wizard identifies the records as fixed length, with a length of 32 bytes each.

7. Click **Next**.

   The File Type window shows that the Data Definition Wizard has correctly identified the file as a single-record-type data file.

8. Click **Next**.

   Confirm the Data Definition Wizard's placement of field separators.

Name the fields and select data types:

1. Click **Next**.

   The Edit Field Properties window shows the field properties.

2. Click a column heading to select a field. Using the table layout:

   a. Enter a name to identify the field in the table layout.

   b. Enter a column heading to identify the field in the view.

   c. Select the data type from the **Type** drop-down list and adjust the format if necessary.

3. Repeat Step 2 for each field you want to define.
4. Click **Next.**

   The Final window shows a summary of table properties.

Name and save the table:

1. Click **Finish.**
2. Name the table `Trans` and click **OK.**

   The Data Definition Wizard creates the table and displays the default view.

Activity 2.2 Copying and Linking Tables

Project: ACL105 Foundations.acl

Table: Inventory

Your tasks

1. Copy the Inventory table and rename it Feb_invntry.
2. Link Feb_invntry to the Month2_invntry.fil data file.

Solution 1

Copy and rename the table:

1. In the **Overview** tab, right-click Inventory and select **Copy.**
2. Right-click the Inventory_review folder and select **Paste.**

   ACL places the copy in the folder and names the table Inventory2.
3. Right-click Inventory2, select **Rename**, enter `Feb_invntry`, and click **OK.**

Solution 2

Link the new table to the data file:

1. Right-click Feb_invntry and select **Link to New Source Data.**
2. In the Select File dialog box, select Month2_invntry.fil and click **Open.**

   The Feb_invntry table is now linked to the data file.

Data Integrity Verification

Activity 3.1 Count, Total, and Statistics

Project: ACL105 Foundations.acl

Table: Trans

Your tasks

1. Determine if the Trans table is well defined and complete.
2. What is the value of the largest credit?
3. Credits issued must not exceed $1,000. Verify if there are any transactions over this limit.
4. Determine the total value of transactions over the credit limit.

Challenge question

- There may be transactions with invoice amounts of $0.00. Identify these types of transactions. How many are there?

Solution 1

Use the Count command to verify the number of records received:

- Select **Analyze** > **Count Records** and click **OK**.

  The total number of records appears in the status bar. The number of records matches the information received.

ⓘ *To view the results in the display area, click the Log tab and double-click the Count command.*

Find the net total and compare it to your control total:

1. Select **Analyze** > **Total Fields.**
2. From the **Total Fields** list, select amount and click **OK.**

   The control totals match the information received.

Solution 2

Run the Statistics command on amount to find the largest value:

1. In the display area, click the **Trans** tab.
2. Select **Analyze** > **Statistical** > **Statistics.**
3. From the **Statistics On** list, select amount and click **OK.**

   The largest credit is $3,366.30.

Solution 3

Use the Count command with a filter to determine if any amounts exceed the credit limit of $1,000.00:

1. In the display area, click the **Trans** tab.
2. Select **Analyze** > **Count Records.**
3. In the **If** box, enter `amount < -1000` and click **OK.**

   Four records exceed the credit limit.

Solution 4

Total the amount for any records over the credit limit:

1. In the display area, click the **Trans** tab.
2. Select **Analyze** > **Total Fields.**
3. From the **Total Fields** list, select amount.
4. In the **If** box, enter `amount < -1000` and click **OK.**

   The total value of transactions over the credit limit is $7,308.42.

Challenge solution

Create a filter to determine which transactions have an invoice amount of $0.00:

1. In the display area, click the **Trans** tab.
2. Click the (Edit View Filter).
3. Enter `amount = 0` in the **Expression** box, enter `zero` in the **Save As** box, and click **OK**.

   Transaction number 12903 has an invoice amount of $0.00.
4. Run the Count command to record the number of records that match the filter in the log.

Activity 3.2 Gaps and Duplicates

Project: ACL105 Foundations.acl

Table: Trans

Your tasks

1. Determine if the table is missing any invoice numbers.
2. Locate and identify transactions categorized as possible duplicate invoices.
3. If there are possible duplicate invoices on the file, are these transactions identical?

Solution 1

Test for missing and duplicate invoice numbers in the sequence:

1. In the **Overview** tab, double-click Trans.
2. Select **Analyze** > **Look for Gaps.**
3. From the **Sequence On** list, select invoice and click **OK.**

   Twelve gaps are detected.
4. Click the **Trans** tab.
5. Select **Analyze** > **Look for Duplicates.**
6. From the **Sequence On** list, select invoice.
7. On the **Output** tab, select the **Screen** option and click **OK.**

   Twenty-two duplicates are detected.

Solution 2

Test for duplicate invoice numbers and include other fields:

1. Click the **Trans** tab.
2. Select **Analyze** > **Look for Duplicates.**

3. From the **Sequence On** list, select invoice.
4. Click **List Fields**, add amount, date, and prodno to the **Selected Fields** list.
5. Click **OK**.
6. Click the **Output** tab, select the **File** option, name the file `Trans_dups_invoice`, and click **OK**.

   The new table contains 42 records with duplicate invoice, date, and prodno values. To view the table, double-click Trans_dups_invoice in the **Overview** tab.

Solution 3

Test the Trans_dups_invoice table for duplicate records:

1. In the **Overview** tab, double-click Trans_dups_invoice.
2. Select **Analyze** > **Look for Duplicates**.
3. Click **Sequence On**, add invoice and amount to the **Selected Fields** list, and click **OK**.

   Results show that there are no records with the same invoice number and amount.

Data Analysis

Activity 4.1 Filters

Project: ACL105 Foundations.acl

Table: Demo

Your tasks

1. Are any batch accounts still being assigned code 99999? If so, how many?
2. Are any GL110 accounts affected by this practice?

Challenge question

- Comment on the number of problem transactions that exceed the $100,000 threshold amount. Include batch 99999 and GL110 transactions in your analysis.

Solution 1

Create a filter:

1. Click (Edit View Filter).
2. In the **Expression** box, enter `batch = "99999"`.
3. In the **Save As** box, enter `batch_99999` and click **OK**.
4. Select **Analyze** > **Count** and click **OK**.

   There are 385 records that match the batch_ 99999 filter. These records equal approximately 77% of the table.

Solution 2

Expand the filter:

1. Click (Edit View Filter).
2. Change the expression to read `batch_99999 AND gl = "110"` and click **OK**.
3. Select **Analyze** > **Count** and click **OK**.

   The new filter identifies 314 records, or more than 62% of the transactions.

Challenge solution

Add the amount condition to the filter:

1. Click (Edit View Filter).
2. Change the expression to read `batch_99999 AND gl = "110" AND amount > 100000` and click **OK**.
3. Select **Analyze** > **Count** and click **OK**.

   Nineteen records match the filter.

Activity 4.2 Computed Fields and Exception Reporting

Project: ACL105 Foundations.acl

Table: Badfile.fil

Your tasks

1. Create a computed field to identify records with at least one integrity problem.
2. The materiality threshold for computed data errors is $0.50 per transaction. Create a computed field to test related data fields for errors greater than this threshold.

Solution 1

Create a computed field:

1. Select **Edit** > **Table Layout** and click (Add a New Expression).
2. In the **Name** box, enter `Problem`, and in the **Default Value** box, enter `F` (no quotation marks).
3. Click (Insert a Condition) and create the following condition and value pairs:

Condition	Value
ISBLANK(invoiceno)	T
ISBLANK(prodno)	T
invoiceno <> "214"	T
orderqty <= 0 OR orderqty > 1000	T
price <= 0	T

4. Click ✓ (Accept Entry) button and close the Table Layout window.
5. Click (Edit View Filter).
6. In the **Filters** list, double-click Problem to move it to the **Expression** box and click **OK** to display the filtered view.
7. Select **Analyze** > **Count** and click **OK**.

   Four records have integrity problems.

Solution 2

Create a second computed field:

1. Click (Remove Filter).
2. Following the procedure in Solution 1, create a computed field to test the reliability of the total value field.

   The total value field is the product of shipping quantity and unit price, so name the computed field `total_test` and enter the following expression in the default value box: `shipqty * price`.
3. Because you are willing to tolerate a difference of up to $0.50 between the data field and the expected value, enter the following global filter to identify those records where the total-value data field varies by more than $0.50 from its recalculated value:

   `total - total_test > 0.50 OR total_test - total > 0.50`
4. With this filter active, select **Analyze** > **Count** and click **OK**.

   Four transaction records exceed the materiality threshold.

Activity 4.3 Summarize

Project: ACL105 Foundations.acl

Table: Demo

Your tasks

1. Identify branches with net credits.
2. Identify the branches with the five largest credits and save results in a filter.
3. Are there any branches over the $500,000 credit threshold?

Challenge question

- Which branch had the highest number of daily transactions in a single day during the period? What was the net amount for that branch?

Solution 1

Summarize the Demo data:

1. Select **Analyze** > **Summarize.**
2. From the **Summarize On** list, select br, and from the **Subtotal Fields** list, select amount.
3. Click the **Output** tab, select the **File** option, name the table `Demo_summ_br`, and click **OK.**
4. Click (Edit View Filter) and enter `amount < 0` in the **Expression** box.
5. Name the filter `Credit` and click **OK.**

Solution 2

Create an index:

1. Select **Analyze** > **Statistics.**
2. Click **Statistics on Amount.**
3. Create a filter `Amount<=Low1` and click **OK.**

Solution 3

Apply a filter:

- In the **Expression** box, enter `amount <-500000` and click **OK.**

Branches 21, 22, 31, and 41 are over the $500,000 credit limit.

Challenge solution

Summarize the Demo data:

1. Select **Analyze** > **Summarize.**
2. Click **Summarize On**, add br and date to the **Selected Fields** list, and click **OK.**
3. From the **Subtotal Fields** list, select amount.
4. Click the **Output** tab, select the **File** option, name the table `Demo_summ_br_date`, and click **OK.**
5. With Demo_summ_br_date open, select **Analyze** > **Statistical** > **Statistics**, select count, and click **OK.**
6. Click the **Demo_summ_br_date** tab.
7. Click (Edit View Filter), enter `count = MAX1` in the **Expression** box, and click **OK.** You can select MAX1 from the **Variables** list.

   Branch 21 had the highest number of daily transactions.

Determine the net amount for that branch:

1. Click (Edit View Filter) and enter `br = "21"` in the **Expression** box, and click **OK.**
2. Select **Analyze** > **Total Fields**, select amount, and click **OK.**

   The net credit for branch 21 is $694,795.08.

Activity 4.4 Classify, Stratify, and Age

Project: ACL105 Foundations.acl

Table: Trans

Your tasks

1. Examine the table and determine which product is the best seller based on quantities sold.
2. Based on default ranges, into which range do most invoice amounts fall? Eliminate any invoices with net credit balances from your analysis.
3. Which quarter of the calendar year 2000 produced the most revenue? Use the aging periods 0, 91, 182, 273, and 366.

Solution 1

Use the Classify command with a subtotal:

1. Select **Analyze** > **Classify.**
2. From the **Classify On** drop-down list, select prodno, and from the **Subtotal Fields** list, select qty.
3. Click the **Output** tab, select the **File** option, name the table `Trans_classify_prodno`, and click **OK.**

   The table shows total quantities sold for each product number.
4. Select **Analyze** > **Statistical** > **Statistics.**
5. From the **Statistics On** list, select qty and click **OK.**

   The largest quantity sold was 4101.
6. Click the **Trans_classify_prodno** tab, click (Edit View Filter), enter `Qty = Max1` in the **Expression** box, and click **OK.**

   The best seller was prodno 052530155, with 4101 units sold.

Solution 2

Determine the range of values:

1. In the **Overview** tab, double-click Trans.
2. Select **Analyze** > **Classify.**
3. From the **Classify On** list, select invoice.
4. From the **Subtotal Fields** list, select amount.
5. Click the **Output** tab, select the **File** option, and name the table `Trans_classify_invoice`.
6. Select **Analyze** > **Statistical** > **Statistics.**
7. From the **Statistics On** list, select amount.
8. In the **If** box, enter `amount >= 0` and click **OK.**
9. Select **Analyze** > **Stratify.**
10. From the **Stratify On** drop-down list, select amount.
11. In the **If** box, enter `amount >= 0` and click **OK.**

    More than 34% of the total invoice amounts fall within the range of 0.00 – 3,789.49.

Solution 3

Age the data:

1. In the **Overview** tab, double-click Trans.
2. Select **Analyze** > **Age.**
3. From the **Age On** drop-down list, select date, set the cutoff date to December 31, 2000, enter the aging periods 0, 91, 182, 273, 366, select amount from the **Subtotal Fields** list, and click **OK.**

   The results show that the third quarter generated the most income with more than 40% of the year's earnings.

Activity 4.5 Extract

Project: ACL105 Foundations.acl

Table: Trans

Your task

- Create a table that contains only transactions from product classes 04 and 09. Name it Trans_extract_class0409.

Solution

Extract the data:

1. Select **Data** > **Extract Data**.
2. In the **If** box, enter the expression `prodcls = "04" OR prodcls = "09"`.
3. In the **To** box, enter the table name `Trans_extract_class0409` and click **OK**.

   The filter shows that 87 records match.

Activity 4.6 Export

Project: ACL105 Foundations.acl

Table: Trans_extract_class0409

Your task

- Generate an Excel file and verify its completeness.

Solution

Export the data to an Excel file:

1. Select **Data** > **Export To Other Application**.
2. Click **Export Fields**, add all of the fields to the **Selected Fields** list, and click **OK**.
3. From the **Export As** drop-down list, select **Excel**.
4. In the **To** box, enter `Trans0409` and click **OK**.

   The original table and the new Excel file each contain 87 records.

ⓘ *The Export tab shows that ACL adds the appropriate extension to the file name.*

Activity 4.7 Index and Sort

Project: ACL105 Foundations.acl

Table: Customer

Your Tasks

1. Reorder the Customer table by credit limit in descending value.
2. Create a table that meets the test "credit limit over $25,000". Ensure that the table is in descending order by credit limit.

Solution 1

Create a descending index:

1. Select **Data** > **Create Index.**
2. Click **Index On** and add limit to the **Selected Fields** list.

   An upward-pointing arrow, which represents ascending order, is displayed beside the field.
3. Click the arrow one time to change it to a downward-pointing arrow for descending order and click **OK.**
4. In the **To** box, enter `Limit_descending` and click **OK.**

   The view is now in descending order by credit limit. Remove the index by selecting None from the **Index** drop-down list in the top right corner of the view.

Solution 2

Create a sorted, descending table that contains records with limits that are greater than $25,000:

1. Select **Data** > **Sort Records.**
2. Click **Sort On,** add limit to the **Selected Fields** list, change the arrow to a downward-pointing arrow, and click **OK.**
3. In the **If** box, enter `limit > 25000` and click **OK.**
4. In the **To** box, enter `limit_desc` and click **OK.**

   The result is a table in descending order by limit for customers with a credit limit over $25,000.

   Forty-eight records match the filter.

   See "Comparing Sort and Index" on page 67 for a discussion of the differences between the Sort and Index commands.

Activity 4.8 Extract Using the Append option

Project: ACL105 Foundations.acl

Table: Trans

Your task

- Extract transactions for product class 01 to one table and transactions for product class 07 to another table. Combine these two new tables into one table named Trans_extract_class_0107.

Solution

Extract and append the data:

1. Select **Data** > **Extract Data**.
2. In the **If** box, enter `amount < 0 AND prodcls = "01"`.
3. In the **To** box, enter `Trans_extract_class_01` and click **OK**.

   Six records match the filter.
4. In the **Overview** tab, double-click Trans.
5. Select **Data** > **Extract Data**.
6. In the **If** box, enter `amount < 0 AND prodcls = "07"`.
7. In the **To** box, enter `Trans_extract_class_0107` and click **OK**.

   Four records match the filter.
8. Double-click Trans_extract_class_01.
9. Select **Data** > **Extract Data**.
10. In the **To** box, enter `Trans_extract_class_0107`, click **OK**, and select **Append**.

    The new table contains 10 records.

ⓘ *To examine the table history for the new master table, select Tools > Table History.*

Activity 4.9 Join (Basic)

Project: ACL105 Foundations.acl

Tables: Empmast, Payroll

Your task

- Identify underpaid employees. There are three of them. Of these, one was not paid at all.

Solution

Use the Join command:

1. In the **Overview** tab, double-click the Empmast table.
2. Select **Data** > **Join Tables.**
3. From the **Secondary Table** drop-down list, select Payroll.
4. From the **Primary Keys** list, select empno.
5. From the **Secondary Keys** list, select empno.
6. Click **Primary Fields**, add empno and pay_per_period to the **Selected Fields** list, and click **OK**.
7. Click **Secondary Fields**, add empno and gross_pay to the **Selected Fields** list, and click **OK**.
8. In the **To** box, enter `Join_underpaid`.
9. Click the **More** tab, select the **Matched Primary Records** option and the **Include All Primary Records** check box, and click **OK**.
10. Click (Edit View Filter), enter `gross_pay < > pay_per_period` in the **Expression** box, and click **OK**.

    Three records match the filter.

Activity 4.10 Join (Intermediate)

Project: ACL105 Foundations.acl

Tables: Empmast, Payroll

Your tasks

1. Confirm that the difference between these two tables for this month is $3,284.82.
2. Identify the following exceptions by using only one join:
   - Valid employees who were paid an incorrect amount
   - Valid employees who were not paid
   - Payroll payments that are not accounted for by matching master table records
3. Identify employees who were paid more than one time.

   Ensure that all identified discrepancies account for the full amount of the difference found in task 1.

Solution 1

1. In the **Overview** tab, double-click Empmast.
2. Select **Analyze** > **Total**, select pay_per_period from the **Total Fields** list, and click **OK**.
3. Select **Edit** > **Variables** and double-click TOTAL1.
4. In the **Save As** box, enter `ppp_total` and click **OK**.
5. In the **Overview** tab, double-click Payroll.
6. Select **Analyze** > **Total**, select gross_pay from the **Total Fields** list, and click **OK**.
7. Select **Window** > **Show Command Line**, enter `Calc ppp_total - TOTAL1`, and click ✓ (Accept Entry).

   The difference is $3,284.82.

Solution 2

Perform a join and apply a filter:

1. In the **Overview** tab, double-click the Empmast table.
2. Select **Data** > **Join Tables**.
3. From the **Secondary Table** drop-down list, select Payroll.
4. From the **Primary Keys** list, select empno.
5. From the **Secondary Keys** list, select empno.
6. Click **Primary Fields**, add empno and pay_per_period to the **Selected Fields** list, and click OK.
7. Click **Secondary Fields**, add empno and gross_pay to the **Selected Fields** list, and click **OK**.
8. In the **To** box, enter `Empmast_join_pay_allps`.
9. Click the **More** tab, select the **Matched Primary Records** option, select the **Include All Primary Records** and **Include All Secondary Records** check boxes, and click **OK**.
10. With the Empmast_join_pay_allps table displayed, click (Edit View Filter) button and apply each of the following filters in turn:

To Detect	Use	Exceptions
Valid employees who were paid an incorrect amount	pay_per_period <> gross_pay AND NOT ISBLANK(empno) AND gross_pay > 0	empno = 000020, 000120
Valid employees who were not paid	NOT ISBLANK(empno) AND gross_pay = 0	empno = 000090
Payroll payments that are not accounted for by matching master table records	gross_pay > 0 AND empno <> empno2	empno2 = 000108, 000109, 000320

Solution 3

1. Remove any filters on the Empmast_join_pay_allps table.
2. Select **Analyze** > **Look for Duplicates**.
3. From the **Sequence On** list, select empno2.
4. Click the **Output** tab, select the **Screen** option, and click **OK**.

   Employee 000320 was paid twice.

To ensure that the total amount of the discrepancy is accounted for, create a computed field:

1. Select **Edit** > **Table Layout**.
2. Click (Add a New Expression).
3. On the **Edit Fields/Expressions** tab, name the new field `difference` and in the **Default Value** box, enter `Gross_pay - pay_per_period`.
4. Click ✓ (Accept Entry).

   The new field is created. To confirm the amount of the discrepancy, run the Total command on the Difference field.

Activity 4.11 Relations (Basic)

Project: ACL105 Foundations.acl

Tables: Ap_trans, Vendor, Inventory

Your tasks

1. Relate the Ap_trans table to the Vendor and Inventory tables.
2. In the Ap_trans default view, add the vendor name and product description columns next to the vendor number and product number columns respectively.

Solution 1

Relate the tables:

1. In the **Overview** tab, double-click Ap_trans.
2. Select **Data** > **Relate Tables.**
3. Click **Add Table** and add the Vendor and Inventory tables to the Relations dialog box.
4. Click and drag prodno from the Ap_trans table to prodno in the Inventory table.
5. Click and drag vendor_no from the Ap_trans table to vendor_no in the Vendor table.
6. Click **Finish.**

Solution 2

Add new columns to the Ap_trans view:

1. Right-click in the view and select **Add Columns**.

   To control the placement of columns in views, highlight the column to the right of where you want the new columns inserted.
2. From the **From Table** drop-down list, select Inventory.
3. In the **Available Fields** list, double-click inventory.proddesc.
4. From the **From Table** drop-down list, select Vendor.
5. Add vendor.vendor_name to the **Selected Fields** list and click **OK**.

   The fields are added to the table and can be rearranged as required.

Activity 4.12 Relations (Intermediate)

Project: ACL105 Foundations.acl

Tables: Ap_trans, Inventory, Dept

This activity assumes that you have completed the solution for *Activity 4.11 Relations (Basic).*

Your tasks

1. Open the existing ACL105 Foundations.acl project and relate the Ap_trans table to the Inventory and Dept tables.
2. In the Ap_trans default view, add the department description column next to the vendor number column.

ⓘ *The product class field in the Inventory table matches the department code field in the Dept table.*

Solution 1

Relate the tables:

1. In the **Overview** tab, double-click Ap_trans.
2. Select **Data** > **Relate Tables.**

   The Relations box shows relations between the Ap_trans table and the Inventory and Vendor tables on prodno and vendor_no respectively.
3. Click **Add Table** and add the Dept table to the Relations dialog box.
4. Click and drag prodcls from the Inventory table to dept in the Dept table.
5. Click **Finish.**

Solution 2

Add a new column to the Ap_trans view:

1. Click the vendor_no column heading.
2. Right-click in the view and select **Add Columns.**
3. From the **From Table** drop-down list, select Dept.
4. Add dept.deptdesc to the **Selected Fields** list and click **OK.**

   The field is added to the table beside vendor_no.

Reporting Results

Activity 5.1 Graphing

Project: ACL105 Foundations.acl

Table: Trans

Your tasks

1. Create a bar graph showing the invoice amounts for each product class.
2. Change the graph type to a 3-D pie chart.

Solution 1

Classify the data and output to a graph:

1. Select **Analyze** > **Classify**.
2. From the **Classify On** drop-down list, select prodcls.
3. From the **Subtotal Fields** list, select amount.
4. Click the **Output** tab, select the **Graph** option, and click **OK**.

Solution 2

Change the graph type:

1. Click (Graph Type).
2. Click (3-D pie chart).
3. Click **OK**.

Index

ACL Training Course Evaluation

Participant's name: ____________________

Instructor's name: ____________________

Date: ____________________

Location: ____________________

General	Excellent				Poor	Comments:
Overall impression of the course	5	4	3	2	1	____________
Learning objectives clearly stated	5	4	3	2	1	____________
Learning objectives met	5	4	3	2	1	____________
Prerequisites appropriate	5	4	3	2	1	____________

Instructor	Excellent				Poor	Comments:
Quality of the instructor	5	4	3	2	1	____________
Instructor's presentation methods	5	4	3	2	1	____________
Content presented clearly and concisely	5	4	3	2	1	____________

Course Content	Excellent				Poor	Comments:
Clear and comprehensive	5	4	3	2	1	____________
Timely and relevant	5	4	3	2	1	____________
Topics of interest addressed	5	4	3	2	1	____________

Practice Sessions and Exercises	Excellent				Poor	Comments:
Overall impression	5	4	3	2	1	____________
Number of activities	5	4	3	2	1	____________
Time allotted	5	4	3	2	1	____________

Materials	Excellent				Poor	Comments:
Overall quality	5	4	3	2	1	
Helped to achieve learning objective	5	4	3	2	1	
Appropriate visual aids	5	4	3	2	1	

Course Facility	Excellent				Poor	Comments:
Overall evaluation of facility	5	4	3	2	1	
Work stations and equipment	5	4	3	2	1	
On-site staff	5	4	3	2	1	
General condition (cleanliness, rest rooms, coat closet, telephone services)	5	4	3	2	1	

Additional comments:

Your comments are important to us. Thank you for completing this survey. Please return this form to your instructor, or mail or fax it to:

ACL Services Ltd.
1550 Alberni Street
Vancouver, BC
Canada V6G 1A5

Telephone: +1-604-669-4225
Fax: +1-604-669-4087
E-mail: info@acl.com
Web: www.acl.com/training