Programming Microsoft Dynamics NAV

* **C/AL:** Client/Application Language is the programming language meant for customization of NAV. It was built by the NAV development team using C++, though we never see any C++ code directly in the NAV product. C/AL is the tool used to define the processes by which data is manipulated, to define the business rules that will control the various applications, and to control the flow of all the logical processing sequences. C/AL is also used to manipulate objects, to control the execution flow of objects, to create new functions complementing the functions that are built in, and to manipulate data in many different ways.
* **C/SIDE:** Client/Server Integrated Development Environment is the development tool specified for using C/AL. It includes the language editor, compiler, debugger, reports and form generators and code management tools. Almost all the C/AL development is done within C/SIDE without the use of external tools. For most application development, NAV is entirely self sufficient except for those services provided by the Windows operating systems. It is possible, though generally not recommended, to write code using a text editor and then import it into C/SIDE.
* **Filtering:** The application of range constraints is to control what data is processed or made visible. For example, a filter for payment data for Customer No. 20134 would show the payments for that customer only. Although not really unique to NAV, filters combined with other NAV features are uniquely powerful in NAV. The extreme flexibility of filtering in NAV allows you to easily create very focused views into the data. Filters can be defined as ranges, boolean expressions, specific selections, etc. that delimit the data to be selected into a subset to be utilized in a process (display, calculation, report, etc.). Thus, NAV filters are a very powerful tool for both the developer and the user.
* **SIFT:** Sum Index Field Technology is a very clever method of providing instantaneous response to user inquiries. Most application systems provide fast response to requests for summary information by maintaining pre-calculated totals ("bucketed data"). NAV retains all data in detail and, through the use of SIFT and applied data filters, it provides the activity totals or subsets of information subject to a wide range of selection constraints instantly. Your data structure design will determine whether SIFT results are available and if available, how fast the response will be. Even though the designers of NAV were very clever in giving you special tools to use, you are still responsible for how well those tools will work for your users.
* **C/FRONT:** This is an application programming interface that allows you to develop applications in other programming languages to access a Microsoft NAV database, either the C/SIDE Database Server or the Microsoft SQL Server. The primary component of C/FRONT is a library of callable C functions, which provide access to every aspect of data storage and maintenance. This allows creation of custom components written in C, C++, VB, Delphi, and the Visual Studio.NET languages as well as other languages that support compatible calling conventions. C/FRONT is only tested by Microsoft for use with code built using either the Watcom C or Microsoft C++ compilers. C/AL triggers cannot be invoked via C/FRONT code. C/FRONT comes as a set of files to be installed guided by the instructions given in the C/FRONT manual.
* **C/OCX:** This is an application interface to allow integration between C/AL and a properly defined OCX routine. This allows access to many ActiveX controls available from third-party vendors. Such controls must be non-visual as far as NAV is concerned (but they may open their own windows for user interaction).

## **Some Unique NAV Terms Defined**

The following are some unique definitions in NAV:

* **C/AL:** Client/Application Language is the programming language meant for customization of NAV. It was built by the NAV development team using C++, though we never see any C++ code directly in the NAV product. C/AL is the tool used to define the processes by which data is manipulated, to define the business rules that will control the various applications, and to control the flow of all the logical processing sequences. C/AL is also used to manipulate objects, to control the execution flow of objects, to create new functions complementing the functions that are built in, and to manipulate data in many different ways.
* **C/SIDE:** Client/Server Integrated Development Environment is the development tool specified for using C/AL. It includes the language editor, compiler, debugger, reports and form generators and code management tools. Almost all the C/AL development is done within C/SIDE without the use of external tools. For most application development, NAV is entirely self sufficient except for those services provided by the Windows operating systems. It...

## **The C/SIDE Integrated Development Environment**

The C/SIDE Integrated Development Environment is referred to as the**Object Designer** within NAV. It is accessed through the**Tools | Object Designer** menu option as shown in the following screenshot:﻿

### Object Designer Tool Icons

The following screenshot shows an**Object Designer** form, containing a list of several tool icons. These Object Designer tool**Icons** are shown isolated in the screenshot and then described briefly in the following table. Some of the terminologies in these descriptions will be explained later in this book. Additional information is available in the C/SIDE**Help** files and the Microsoft NAV documentation.