



Type Library Browser

Introduction

The *ProvideX Type Library Browser* (TLB) was developed to provide extended type information for Windows COM objects and to help simplify the process of creating event class objects. This utility (`pvxtlb.exe`) is freely downloadable from www.pvx.com.

The TLB can be used to display any COM object's type library, showing the object's CoClass and GUID information along with the properties, methods, and events supported by the control (see below for an explanation of each).

It can also be used to create ProvideX OOP objects that will help simplify the handling of events generated by COM controls.



Note: A GUID (Globally Unique Identifier) is a 128-bit number maintained in the Windows Registry for uniquely identifying COM objects, DLLs, etc. This number may be used to obtain details on any COM object in the Registry (type library, physical location, etc.).

Using the TLB

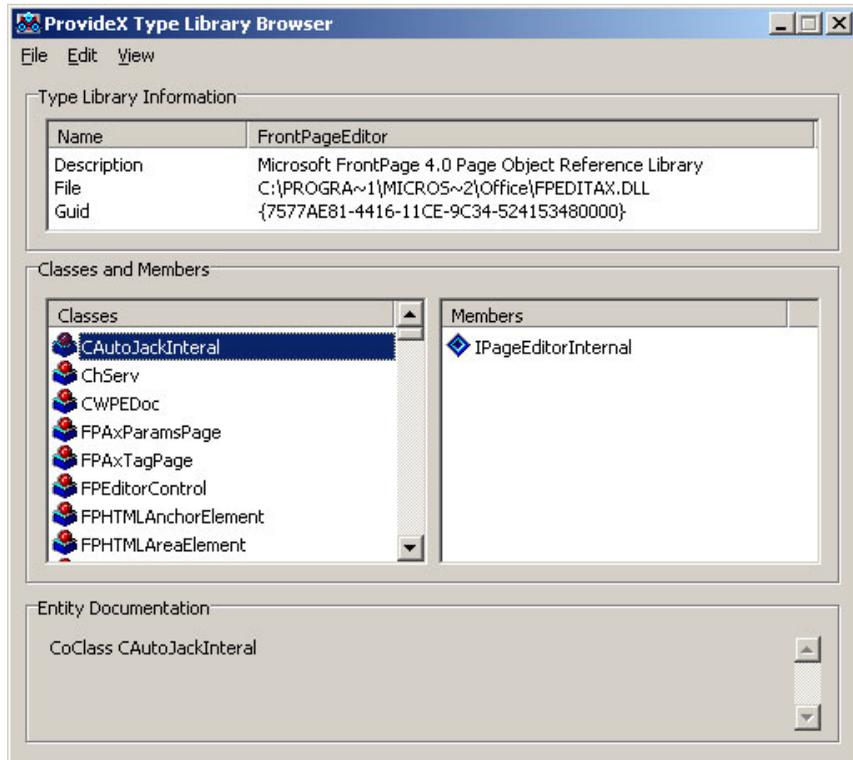
The following steps illustrate how to start the Type Library Browser and display extended type information for a selected COM object (in this case, the Microsoft Calendar Control):

1. Start the TLB by launching `pvxtlb.exe` from Windows Explorer, or via **Start > Run..**
2. Select **Open** from the **File** menu to display all registered OLE/COM objects. The Registered Type Libraries dialogue window appears.
3. Scroll down and select Microsoft Calendar Control.

The type library information for the Microsoft Calendar Control is loaded into the TLB.

Refer to the illustration on the following page.

The Type Library Information section includes a description of the object, the location and name of the OCX file, and the GUID. (This can be stored internally in this OCX file or in a separate file with a .TLB extension).

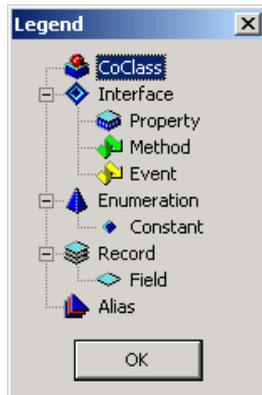


The Classes and Members lists contain all the components of this control.

The Entity Documentation section provides more detail for the selected item including type information and parameter lists.

4. Click on the Calendar object in the Classes list. This shows that the Calendar object has two members: *DCalendarEvents* and *ICalendar*.
5. The icon to the left of each item denotes the class type. To see a complete list of the icons used to identify class types in the TLB, select **Legend** from the **View** menu.

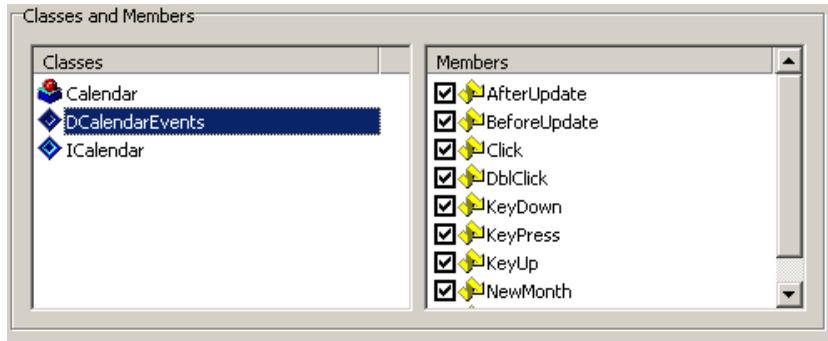
Refer to the illustration on the following page.



The class types are described as follows:

- CoClass** A CoClass is a COM class which contains a collection of interfaces defined by the COM object. The Calendar class is considered a CoClass object.
- Interface** An interface is a class which exposes methods, properties and events for use by other objects. The DCalendarEvents and ICalendar classes are interfaces.
- Property** Properties are the class's member variables that represent the class's current state. Typically properties should only be set or read by class methods. (Refer to the ProvideX OOP documentation for more information on class functions and properties).
- Method** Essentially, methods are similar to ProvideX class functions. They direct the control's behavior.
- Event** COM objects can generate events to let the host object know something has changed; e.g., when a user clicks a button changing the current month, the Calendar object may fire a NewMonth event. This event will be dispatched to any host object that has registered to listen for NewMonth events.
- Enumeration** An Enumeration is a collection of related integer constants. COM object's Constants are only accessible through Enumerations.
- Constant** Constants are attributes that cannot be changed, usually numbers; e.g., an OCX that controls a multiline object has three constants representing the possible alignment of text: *left*, *right*, and *center*. These constants would be exposed through an Enumeration.
- Record** Records represent collections of elements, called fields.
- Field** A field is an element variable contained in a record (much like a single field in an IOList for a file).
- Alias** An alias is a data type that is a reference to another data type.

6. Close the Legend window and select the ICalendar class. The Members list will now display the methods and properties of the ICalendar class. This class contains about a dozen methods with names such as NextWeek() and NextYear(), which can be accessed using the ProvideX OCX/COM Interface.
7. Select the DCalendarEvents class from the Classes list. The Members list will display all the methods, properties, and events supported by the DCalendarEvents class.



Check boxes identify which members are to be used by the TLB for generating an *event template* (described below). Initially, they are all selected by default. This defines all the events the Calendar control can fire.

Creating an Event Template

The following steps describe the event template generation feature of the TLB:

1. To create an event template for the DCalendarEvents interface, choose events (select/deselect check boxes) from the Members list, then click the link **Create an EventTemplate based on this Interface** located in the Entity Documentation section. The Save As file box appears.

2. Save the generated file as DCalendarEvents.pvc. The TLB will generate a text file that contains the source code for a ProvideX OOP object. This text file can be opened with an editor, such as *Notepad*.

The ProvideX event handler class DCalendarEvents.pvc has a function declared for each member event selected in the Members list. If the NewMonth event check box is selected then an Event_NewMonth function will be declared.

3. Each function contains a comment "Insert code here". Replace this comment with the necessary ProvideX code; e.g.,

```
Print "NewMonth Event has been fired"
```

Retrieving a Loaded COM Object's Type Library Information

The TLB accepts optional arguments that enable the retrieval of the type library information of a given COM object. To retrieve the type library information of the Calendar control, pass the name and path of the Calendar OCX. To do this from a ProvideX console, perform the following

```
invoke "pvxtlb.exe "+QUO+C:\Program Files\Microsoft
      Office\Office\MSCAL.OCX"+QUO
```

ProvideX objects contain two properties: *PvxTypeLib* and *PvxISA*. The *PvxTypeLib* property returns the type library file from an object. The *PvxISA* property contains the name of the instantiated class. Invoke the TLB from a ProvideX console and pass the value of these two properties to display the type library information of a loaded COM object; e.g.,

1. Instantiate the Calendar COM Object:

```
Def object x, "MSCAL.Calendar.7"
```

2. Invoke the TLB, passing the ProvideX object's *PvxTypeLib*\$ and *PvxISA*\$ properties:

```
Invoke "pvxtlb.exe "+QUO+x'PvxTypeLib$+QUO+" "+x'PvxISA$
```

The TLB window displays the Calendar controls type library with the *Icalendar* class highlighted.

An alternative method for invoking the Type Library Browser is to use the ProvideX command line shortcut program called **TLB**; e.g.,

```
Def object x, "MSCAL.Calendar"
TLB x
```

