

A Real Life Experience In Object Oriented Templates

Encapsulating the windows API

Thomas R. Hummel
Information Packaging
Unlimited
Phone: (610) 617-8820
www.cwupage.com

Why OOP?

- The windows API is already object oriented
- API calls are reusable code which you can call from anywhere in any application
- OOP is not only a method of programming, with syntax and structure, but also a way of thinking

Why Templates?

- Templates allow the functionality to be easily customized at design time
- Functionality can easily be added to multiple applications
- Templates are familiar

Why the Windows API?

- Some tasks just cannot be done effectively with Clarion
 - You open up a whole new world to your applications
-

Create Your Classes

- Identify your objects
 - Develop the class interfaces
 - Properties
 - Methods
 - Create your prototypes
-

Create Your Include Files

- Class prototypes
 - Class methods
 - Windows API prototypes
 - Using CW winapi utility
 - What are all of those strange data types?
 - Windows equates
-

Create Your Template

- Include your include files
 - Include your API prototypes
 - Include your equates
 - Create the programmer interface
 - From here you can create subclasses
 - Code templates to call methods can add substantial productivity
-

Some Questions to Answer

- How do I choose what should be *encapsulated* in my objects?
 - How do I use *inheritance* and *polymorphism* to create subclasses?
 - How do I create instances of my objects?
 - How do I make my objects reusable?
-
