

# USW - Schnittstelle

## User Supplied Widgets

Mit der USW-Schnittstelle können Sie den IDM um selbstdefinierte Dialogobjekte erweitern.

Dies ist ein weiterer Schritt bei der Öffnung des IDM zur Einbindung externer Komponenten. Bisher konnten Motif-Widgets bereits über das Canvas-Objekt in den IDM integriert werden.

### Selbstdefinierte Klassen

Jetzt können Sie eigene Klassen von Dialogobjekten definieren. Diese selbstdefinierten Klassen von Dialogobjekten erhalten automatisch die bekannten Eigenschaften von IDM Dialogobjekten. Ohne weiteres Zutun steht Ihnen für diese Objekte der Vererbungsmechanismus zur Verfügung. Auch von diesen Dialogobjekten können Defaults, Modelle und Instanzen definiert werden. Um die Standardattribute wie `.xleft`, `.xright`, `.width`, `.height`... brauchen Sie sich nicht zu kümmern, diese erhalten die neuen Objekte automatisch.

### Die Architektur der USW-Schnittstelle

Die folgende Abbildung zeigt die Architektur der USW-Schnittstelle. Sie bietet einen offengelegten Anschlußpunkt, an dem Sie Ihre eigenen Widgets an den IDM ankoppeln können. Das User Class-File muß vom Benutzer bereitgestellt werden. Es hat die Aufgabe, Anfragen wie `create`, `destroy` und `attribute change` in die entsprechenden Funktionsaufrufe der

### User Class File-Funktionen

- `create()`  
Die Funktion `create` wird vom IDM Kern aufgerufen, wenn eine neue Instanz der Dialogklasse erzeugt werden soll.
- `destroy()`  
Die Funktion `destroy` wird aufgerufen, bevor man ein Widget löscht.
- `update()`  
Die Funktion `update` ist das Herz der USW. Sie muß Attributänderungen anzeigen, indem die entsprechenden Widget Ressourcen verändert werden.
- `method()`  
Diese Funktion wird verwendet, um die Methoden: `insert`, `:delete`, `:clear` und `:exchange` schneller aufzurufen.

Außerdem enthält diese Datei eine Liste von Attributbeschreibungen, Ereignissen und die Abbildung auf die Motif Ressourcen, die vom Widget verwendet werden.

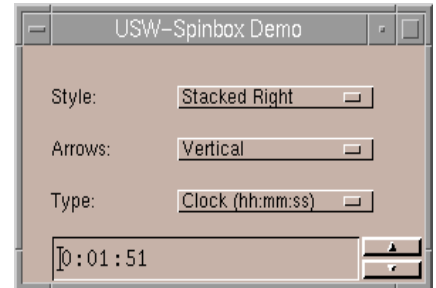
### USW-Register

Diese Funktion meldet alle neuen Dialogklassen beim IDM Kern an. Sie muß sich in Ihrer eigenen Objektdatei befinden und zu der Anwendung gelinkt werden.

### Das Linken der Anwendung

Die IDM Bibliotheken, zumindest die Bibliotheken `libUSW` und `libIDM`, müssen mit der Anwendung verbunden werden; dabei ist zu beachten, daß die Datei `usw_register.o` in der Kommandozeile vor der `libIDM`-Bibliothek stehen muß.

### Beispiel für eine IDM-Anwendung mit einem Spinbox-Widget



### Voraussetzungen

- ab IDM Version A.03.08
- Das Motif-Widget muß über eine Objektdatei oder eine Bibliothek verfügbar sein
- Gute Kenntnisse des X-Toolkits oder Motif sowie der Widgets



ISA Informationssysteme  
für computerintegrierte  
Automatisierung GmbH

Azenbergstr. 35  
D-70174 Stuttgart

Tel.: +49 711 22769 20  
Fax: +49 711 22769 29  
Email: info@isa.de  
www.isa.de