Ein kleines Howto für das DVBViewer-Plugin

Installation

Kopiere die DVBViewer.dll sowie die die (Beispiel-)dvbviewer.vbs und (Beipiel-)dvbviewer.gml ins PLUGIN-Verzeichnis von Girder.
Starte Girder und aktiviere unter Einstellungen/Plugins das "DVBViewer COM Control Plugin"
Importiere die dvbviewer.gml aus dem Girder-Plugin-Verzeichnis. Wenn Du Girder nur für den DVBViewer benutzt, kannst Du natürlich auch die dvbviewer.gml öffnen und mit dieser arbeiten.

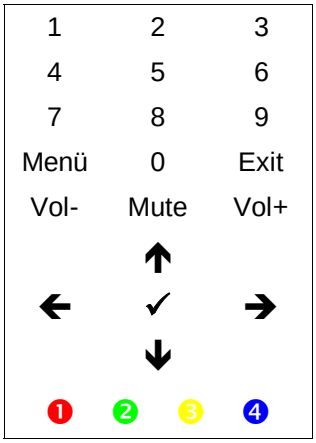
Im folgenden wird davon ausgegangen, dass folgende Verzeichnis-Struktur existiert
C:\Programme\DVBViewer
C:\Programme\Girder

und dass Du es mit entsprechenden Girder-Plugins bereits geschafft hast, dass Girder die Fernbedienung "versteht", Du also schon sog. Ereignisse der Fernbedienung entlocken konntest.

Die Dateien dvbviewer.dll, dvbviewer.vbs und dvbviewer.gml liegen im Verzeichnis
C:\Programme\Girder\Plugins

Beispiel: Mini-Fernbedienung

Um dieses Beispiel halbwegs überschaubar zu halten und zu zeigen, was man mit dem neuen Plugin daraus machen kann, gehen wir davon aus, dass die benutzte Fernbedienung außer den Ziffern-Tasten, den Navigationstasten (Links, Rechts, Raus, Runter, OK), nur die vier Farb-Tasten, Tasten für die Lautstärke und Tasten für den Aufruf und das Verlassen des OSDs hat.



Die (Beispiel-)DVBViewer.gml nimmt nun eine solche Fernbedienung an. Wer darauf aufbauend seine Fernbedienung darin abbilden will, kann sich neue Tasten später hinzufügen.

Zuweisen der Fernbedienungstasten

Als ersten Schritt sollte man das einfachste tun, nämlich den Befehlen unter "Remote Control" die entsprechenden Ereignisse zuzuweisen.
Im Fall "Vol+" würde das also heißen, dass Du die Gruppe "Remote Control" öffnest und den Befehl "Vol+" anklickst. Rechts im Girder-Fenster klickst Du auf "Ereignis lernen", nachdem Du sichergestellt hast, dass in der Dropdownbox links daneben "Alles" steht. Dann drückst Du die Taste, mit der Du auf der Fernbedienung die Lautstärke erhöhen willst. Anschließend siehst Du im Baum links unterhalb von "Vol+" einen "Ereignisstring".

Wiederhole das für alle anderen Tasten.

Du kannst das Ganze nun bei entsprechend großer Fernbedienung (ungefähr in der Größenordnung einer Tastatur) anstatt in der Gruppe "Remote-Control" direkt in der Gruppe "DVBViewer" machen und die Gruppe Remote Control löschen. Dann jedoch benötigst Du die nächsten Schritte nicht mehr. :-)

Umbelegen der Funktionen zu einer Taste

Im Fall der Ziffern-Tasten ist die Belegung aktuell im DVBViewer so, dass ihnen die entsprechenden Fernsehkanäle 0-9 zugeordnet sind. Wer jetzt aber nicht die Känale darüber ansteuern sondern seine Favoriten wählen möchte, klickt die "1" an und anschließend auf der rechten Seite unter dem Reiter "Plugins" auf "Einstellungen" (DVBViewer COM Control Plugin sollte bereits voreingestellt sein). In der folgenden Dialogbox wählt man dann anstatt "Channel1" "Fav1" und klickt dann auf "OK".

> ***Hinweis***: Wenn später einmal neue Funktionen im DVBViewer eingebaut werden, muss die dvbiewer.ini erweitert oder durch eine aktuellere ersetzt werden.

Wiederhole das entsprechend für die Tasten 2-0.

Die Farb-Tasten sowie die Lautstärke-Tasten belassen wir in diesem Beispiel hier unverändert, aber auch hier kannst Du natürlich auf die gleiche Weise andere Funktionen auf diese Tasten definieren. Macht halt nur keinen sonderlichen Sinn ;-)

Soweit so gut, bis zu dieser Stelle kann das DVBViewer-Plugin genau soviel wie früher ...

Modus-abhängige Belegung

Den notwendigen WAF gibt´s natürlich nur, wenn die Navigations-Tasten genau das machen, was man sich intuitiv vorstellt (und ehrlich gesagt, wer will schon auf der Fernbedienung im Halbdunkel erst die Lesebrille anziehen, um die Tasten und deren Funktion zu finden?)

Wie im Forum schon des öfteren bemerkt, hat eine anständige Fernbedienung i.a. immer Lautstärke-Tasten, aber noch lange nicht Tasten, um in einem Film hin und her zu springen. Klar, die liegen auf den OSD-Farb-Tasten, aber das steht auf denen nicht drauf und man kann es sich auf Dauer merken, wenn man ab und zu mal mit "Blau" die entsprechende Hilfe anzeigen lässt.

Im folgenden soll als Beispiel die Belegung der Navigationstasten so geändert werden, dass sie die Lautstärke-Funktionen komplett übergehen und stattdessen im Playback-Modus im Film vor und zurückspringen.

Wenn Du Dir den Befehl bei "left" einmal ansiehst, wirst Du feststellen, dass hier nicht wie bei den zuvor besprochenen Tasten das Plugin direkt angesteuert wird, sondern abhängig vom Wert im Girder-Register 1 entweder zu "Playback-JumpLeft" oder zu "OSD-left" gesprungen wird. Das Register 1 beinhaltet die Nummer des aktuellen OSD-Fensters bzw. "-1", wenn gerade kein OSD-Fenster angezeigt wird. Somit wird gewährleistet, dass das OSD den entsprechenden Befehl "left" zum Navigieren in den OSD-Seiten erhält, ansonsten in einem Film oder einer Musik-Datei zurückspringt.

Woran erkennt Girder nun, welches das aktuelle DVBViewer-Fenster ist? An dieser Stelle kommt die dvbviewer.vbs ins Spiel. Das neue DVBViewer-Plugin erhält nämlich diverse Nachrichten vom DVBViewer (z.B. OnWindowChange), und ruft die entsprechende Routine aus der dvbviewer.vbs auf. (Eine enstprechende Liste gibt´s im Anhang).

Sobald also z.B. ein OSD-Seite aufgerufen oder geschlossen wird, wird dies der COM-Schnittstelle mitgeteilt, auf die auch das DVBViewer-Plugin Zugriff hat. Dieses wiederum ruft die in der dvbviewer.vbs definierte Routine windowchange auf. In unserem Beispiel tut die Routine nichts anderes, als das Girder-Register 1 auf denselben Wert zu setzen, der in der WindowID steht.

Analog zu "left" wurde "right" so gesetzt, dass zu "Playback-JumpRight" oder zu "OSD-right" gesprungen wird.

"up" und "down" unterliegen zwei Abhängigkeiten:

1. Ist das OSD aktiv (die WindowID ist also ungleich -1), wird OSD-Up benutzt
ansonsten wird
2. unterschieden, ob momentan eine Datei abgespielt wird oder nicht
   a) es wird eine Datei abgespielt => Die Wiedergabe wird gestoppt
   b) es wird keine Datei abgespielt => Es wird der nächste Kanal in der Kanalliste aufgerufen

1. und 2. wird wie oben beschrieben auf Basis des Registers 1 entschieden, a) und b) auf Basis des Registers 2, das in den Routinen PlaybackStart und PlaybackEnd gesetzt wird.

Ereignis-Steuerung

Analog zum Starten der event.exe aus dem Girder-Verzeichnis oder dem Aufruf von `Triggerevent("EventName", Device)` wird mit `SendEvent "Eventname"` in der dvbviewer.vbs ein Ereignis in die Girder-Warteschlage geschrieben. Nützlich ist dies vor allem dann, wenn man nicht nur Register order Variablen setzen möchte, sondern auch Girdereigene Funktionen (Aktivieren und Deaktivieren von Gruppen, Timer zurücksetzen etc.) aufrufen möchte. Dazu ordnest Du diesem Befehl (oder Multibefehl) ein Ereignis zu, indem Du rechts im Girder-Fenster auf "Ereignis lernen" klickst, nachdem Du sichergestellt hast, dass in der Dropdownbox links daneben "Girder-Ereignis" steht. In dem folgenden Dialogfenster trägst Du den Namen des Ereignisses ein, das Du aus dem dvbviewer.vbs heraus erzeugen möchtest (z.B. PlaybackStart). Anschließend siehst Du im Baum links unterhalb dieses (Multi-)Befehls einen "Ereignisstring".

Skripten bis der Arzt kommt

Analog wie die Skripte, die im Verzeichnis SCRIPTS unter dem DVBViewer-Verzeichnis liegen, kann auch die dvbviewer.vbs programmiert werden. Es hat den gleichen Zugriff auf die COM-Schnittstelle wie die Einzelskripte. Der Vorteil ist hierbei, dass alle Routinen in einer Datei abgelegt sind und beim Start von Girder dem DVBViewer-Plugins bekannt sind. Es entfällt somit der Zeitaufwand für das einzelne Laden der Skripte. In der Beispiel-dbviewer.vbs sind alle aktuell zur Verfügung stehenden Routinen aufgeführt (auch wenn sie teilweise nichts tun). Es steht nun jedem frei, sich seine Routinen so zu basteln, wie er mag.

Debugging

Lars hat in das Plugin noch nette SchmankerIn eingebaut, die das Debugging der selbsterstellten Routinen enorm erleichtert.

Dafür gibts eine spezielle DLL (DVBScriptDbg.dll), die Du in ein Unterverzeichnis des Girder-Plugin-Verzeichnisses (z.B. debug) kopierst. Sie muss im System mit regDVBV.bat registriert werden (wenn Du sie nicht mehr brauchst, kannst Du sie wieder mit deregDVBV.bat deregistrieren).
Am Besten anschließend einen System-Neustart durchführen.
Weitere Erläuterungen dazu gibt´s im Anhang ...

Anhang

Girder-Skripting-Funktionen

Im sog. "Variablen Manipulationsskript" von Girder stehen mit dem DVBViewer COM Control Plugin einige Funktionen zur Verfügung, die im folgenden beschrieben sind:

DVBV_PropSetValue
Parameter : Name, Value
Funktion : überschreibt den Wert der DVBViewer-Variablen Name mit dem Wert Value
Beispiel :
```
DVBV_PropSetValue("#Displaytext",Displaytext)
```

DVBV_PropGetValue
Parameter : Name
Funktion : übergibt den Wert der DVBViewer-Variablen Name
Beispiel :
```
Displaytext = DVBV_PropSetValue("#Displaytext")
```

DVBV_OSDRefresh
Parameter : - keine -
Funktion : Veranlasst das OSD, die aktuell dargestellte Seite neu aufzubauen
Beispiel :
```
DVBV_OSDRefresh()
```

dvbviewer.vbs

Dieses Skript wird mit dem DVBViewer-Plugin gestartet.

Die Routinen der dvbviewer.vbs

Folgende Routinen ruft das Plugin auf, wenn das entsprechende Event vom DVBViewer ausgelöst wird:

ChannelChange
Parameter : Channel
Funktion : wird nach einem Kanalwechsel aufgerufen
Beispiel :
```
sub ChannelChange(Channel)
   DVBVChannelnumber = channel
   DVBVChannelname   = CurrentChannel.Name
   sendevent "Channelchange"
end sub
```

PlaybackStart
Parameter : - keine -
Funktion : wird nach dem Start der Wiedergabe einer Mediendatei aufgerufen
Beispiel :
```
sub PlaybackStart
   setvar "MediaTitle",propGetValue("#Media.title")
   setvar "MediaFile",propGetValue("#Media.file")
   sendevent "PlaybackStart"
end sub
```

Playbackend
Parameter : - keine -
Funktion : wird nach dem Ende (Stop) der Wiedergabe einer Mediendatei aufgerufen
Beispiel :
```
sub PlaybackEnd
   ClearVar "MediaTitle"
   ClearVar "MediaFile"
   sendevent "PlaybackEnd"
end sub
```

WindowChange

Parameter : WindowID

Funktion : wird mit jedem Öffnen und Schließen des OSDs sowie einem Wechsel zu einer anderen OSD-Seite aufgerufen

Beispiel :

```
sub WindowChange(Window)
   aid = Window
   if not OSD.isVisible then
      aid = " - 1 "
   else if WindowManager.IsOverlay then
         aid = " - " & WindowManager.OverlayWindowID
      end if
   end if
   Setvar reg1, aid
   sendevent "WindowChange"
end sub
```

onRDS

Parameter : Message

Funktion : wird aufgerufen, sobald sich der Radiotext des gewählten Kanals ändert

Beispiel :

```
sub onRDS(Message)
   Setvar "RDSText", Message
   sendevent "RDSChange"
end sub
```

OnAction

Parameter : ActionID

Funktion : Mit dieser Funktion können zusätzlich zum im DVBViewer definierten Aktionen eigene Aktionen definiert werden. Aktionen, die der DVBViewer nicht selbst abarbeitet leitet er an diese Funktion weiter.

Beispiel :

```
sub OnAction(ActionID)
   Setvar "Action", ActionID
   sendevent "Action"
end sub
```

StartRecord

Parameter : RecID

Funktion : signalisiert den Beginn einer Aufnahme und übergibt die entsprechende Aufnahme-ID

Beispiel :

```
sub StartRecord(ByVal ID)
   Setvar "Recording", TimerManager.Item(RecID).description
   sendevent "DVBV.Startrecord"
end sub
```

Endrecord

Parameter : - keine -

Funktion : signalisiert das Ende einer Aufnahme

Beispiel :

```
sub Endrecord
   sendevent "DVBV.Endrecord",18
end sub
```

onPlaystateChange

Parameter : RendererType, State

Funktion : signalisiert in RendererType die Art der dargestellten Wiedergabe (0=unbekannt, 1=DVD, 2=direkt vom Transponder oder 3=Mediendatei) und in State den aktuellen Status (0=Stop, 1=Pause, 2=Play).

Beispiel :

```
sub onPlaystateChange(RendererType, State)
   Setvar "CurrentState", State
   Setvar "RendererType", RendererType
   sendevent "PlayStateChange"
end sub
```

onControlChange

Parameter : WindowID, ControlID

Funktion : Sobald im OSD ein Button oder Liste fokusiert (ausgewählt) wurde, wird hierüber die Id des Fensters (WindowID) und die ID der entsprechenden Control übermittelt.

Beispiel :

```
sub onControlChange(WindowID, ControlID)
   Setvar "Selected_Control", propGetValue("#currentcontrol")
   sendevent "ControlChange"
end sub
```

onSelectedItemChange

Parameter : - keine -

Funktion : In Menüs oder Listen beinhaltet #selecteditem das ausgewählte Listenelement. Sobald sich dieses ändert wird dies über diese Routine signalisiert.

Beispiel :
```
sub onSelectedItemChange
   Setvar "Selected_Item", propGetValue("#selecteditem")
   sendevent "ItemChange"
end sub
```

onPlaylist

Parameter : aFilename

Funktion : Sobald eine neue Mediendatei wiedergegeben wird, wird in aFilename der komplette Pfadname der Datei übergeben.

Beispiel :
```
sub onPlaylist(aFilename)
   Setvar "PlayList_Item", aFilename
   sendevent "NewMediaFile"
end sub
```

onAddRecord

Parameter : RecID

Funktion : Mit dem erstellen eines neuen Timereintrags wird in RecID der neu erzeugte Timer übergeben.

Beispiel :
```
sub onAddRecord(ID)
   sendevent "AddTimer"
end sub
```

Close

Parameter : - keine -

Funktion : Signalisiert das Beenden des DVBViewers

Beispiel :
```
sub close
   sendevent "DVBViewerCloses"
end sub
```

main

Parameter : - keine -

Funktion : Wird einmal beim Starten des DVBViewers aufgerufen und dient der Initialisierung von Variablen und ggf. dem Aktivieren des Debug-Modus.

Beispiel :
```
sub main()
   DVBVMode = ""
end sub
```

Spezielle Funktionen für Girder

SetVar

Typ : interne Routine

Funktion : Setzt den Wert einer Variablen auf einen bestimmten Wert. Wurde die Variable zuvor noch nicht definiert, wird sie hiermit erzeugt.

Parameter : VarName, VarValue

Beispiel :
```
SetVar DVBModus, "TV"
```

ClearVar

Typ : interne Routine

Funktion : Entfernt eine zuvor gesetzte Variable aus Girder

Parameter : VarName

Beispiel :
```
ClearVar DVBModus
```

SendEvent

Typ : interne Routine

Funktion : Analog zum Starten der event.exe aus dem Girder-Verzeichnis oder dem Aufruf von Triggerevent("EventName", Device) wird hiermit ein Ereignis in die Girder-Warteschlage geschrieben.

Parameter : EventName[, Device]
Device ist optional, standardmäßig wird 18 übergeben.

Beispiel:
```
SendEvent "PlaybackStart"
```

Debugging-Funktionen

Debugging
Typ : Variable vom Typ Boolean
Funktion : Ist der Wert true, so werden die Debugging-Befehle (s.u.) abgearbeitet, ansonsten ignoriert.
Sie kann in der Routine main() gesetzt werden und hat dann für alle Routinen Gültigkeit.
Wird sie nur in einer der anderen Routinen auf "true" gesetzt, gilt sie nur für diese Routine.

ShowDebug
Typ : interne Routine
Parameter : - keine -
Funktion : zeigt den Skriptdebugger, sofern er installiert ist.

CloseDebug
Typ : interne Routine
Parameter : - keine -
Funktion : Schließt den Skriptdebugger, sofern er geöffnet ist

AddDebug
Typ : interne Routine
Parameter : Liste von Parametern
Funktion : Ist der Skriptdebugger installiert, werden die übergebenen Parameter im Debugger, ansonsten in einem separaten Fenster angezeigt.

ShowTags
Typ : interne Routine
Parameter : - keine -
Funktion : Ist der Skriptdebugger installiert, werden die Inhalte der DVBViewer-Tags im Debugger, ansonsten in einem separaten Fenster angezeigt.
(Entspricht dem Aufruf von propGetAll)