

KopieerVenster (Window)

Constants

Integer MACHT = 26

Properties

m_stop as boolean

m_aGekopieerd as double

KopieerVenster.Action_Start:

Private Sub Action_Start()

 AbleControls(false)

 DoeAlles()

 AbleControls(true)

End Sub

KopieerVenster.Action_Cancel:

Private Sub Action_Cancel()

 m_stop=true

End Sub

KopieerVenster.AbleControls:

Private Sub AbleControls(actief as boolean)

 PB_Cancel.enabled=not actief

 PB_Start.enabled=actief

End Sub

KopieerVenster.DoeAlles:

Private Sub DoeAlles()

 dim invoerMapSpec as FolderItem

 dim uitvoerMapSpec as FolderItem

 dim aBestand as integer

 dim start as double

 dim isGekopieerd as boolean

 dim einde as double

 dim duur as double

 dim resultaat as string

 dim aBytePerSeconde as integer

 dim aMBPerSeconde as double

m_aGekopieerd=0

m_stop=false

// 1. Mappen.

```
if not ST_Bron.Geef_spec(invoerMapSpec) then
  MsgBox("Get 'Source' folder fails.")
  return
end
if not ST_Doel.Geef_spec(uitvoerMapSpec) then
  MsgBox("Search 'Destination' folder fails.")
  return
end
```

// 2. Tel.

```
if not Tel(invoerMapSpec,aBestand) then
  if m_stop then
    MsgBox("Canceled.")
  else
    MsgBox("Count fails.")
  end
  return
end
PB_Voortgang.value=0
PB_Voortgang.maximum=aBestand
```

// 3. Doorloop.

```
start=Microseconds()
isGekopieerd=KopieerMap(invoerMapSpec,uitvoerMapSpec)
einde=Microseconds()
duur=einde-start
```

```
if isGekopieerd then
  if duur>0 then
    aBytePerSeconde=(m_aGekopieerd/duur)*1000000+0.5
    resultaat=resultaat+"Speed is "+MEERVOUD(aBytePerSeconde,"byte")+"/s."+EndOfLine
    if aBytePerSeconde>=&hFFFFFF then
      aMBytePerSeconde=aBytePerSeconde/&hFFFFFF
      resultaat=resultaat+"Speed is "+STRF2(aMBytePerSeconde)+ " MB/s."+EndOfLine
    end
  end
  MsgBox(resultaat)
else
  if m_stop then
    MsgBox("Canceled.")
  else
    MsgBox("Copy fails.")
  end
end
```

```
PB_Voortgang.value=0
PB_Voortgang.maximum=1
```

End Sub

KopieerVenster.KopieerMap:

```
Private Function KopieerMap(invoerMapSpec as FolderItem, uitvoerMapSpec as FolderItem) As boo
    lean
    dim altem as integer
    dim i as integer
    dim invoerSpec as FolderItem
    dim uitvoerSpec as FolderItem
    dim opm as string

    altem=invoerMapSpec.count
    for i=1 to altem
        if m_stop then
            return false
        end

        invoerSpec=invoerMapSpec.TrueItem(i)

        if invoerSpec.directory then

            if not ExistHulp.ZoekOfMaakMap(uitvoerMapSpec, invoerSpec.name, uitvoerSpec) then
                return false
            end
            if not KopieerMap(invoerSpec, uitvoerSpec) then
                opm="Traverse "+DPAD(invoerSpec)+" fails."
                MsgBox(opm)
                return false
            end

        else

            uitvoerSpec=uitvoerMapSpec.TrueChild(invoerSpec.name)
            if uitvoerSpec=nil then
                MsgBox("Out of memory.")
                return false
            end
            if uitvoerSpec.exists then
                ExistHulp.MsgBestaatAl(uitvoerSpec)
                return false
            end
            if not Kopieer(invoerSpec, uitvoerSpec) then
                opm="Copy "+DPAD(invoerSpec)+" fails."
                MsgBox(opm)
                return false
            end

        end
    next
    return true
end
```

End Function

KopieerVenster.Kopieer:

```
Private Function Kopieer(invoerSpec as FolderItem, uitvoerSpec as FolderItem) As boolean
    if not KopieerSpecNaarSpec(invoerSpec, uitvoerSpec) then
        return false
    end
```

```
m_aGekopieerd=m_aGekopieerd+invoerSpec.length
```

```
PB_Voortgang.value=PB_Voortgang.value+1
```

```
App.DoEvents()
```

```
return true
```

End Function

KopieerVenster.Tel:

```
Private Function Tel(invoerMapSpec as FolderItem, byref aBestand as integer) As boolean
    dim altem as integer
    dim i as integer
    dim invoerSpec as FolderItem
    dim opm as string
```

```
altem=invoerMapSpec.count
```

```
for i=1 to altem
    if m_stop then
        return false
    end
```

```
invoerSpec=invoerMapSpec.TrueItem(i)
```

```
if not invoerSpec.visible then
```

```
    // opm: overslaan
```

```
elseif invoerSpec.directory then
```

```
    if not Tel(invoerSpec, aBestand) then
        opm="Traverse "+DPAD(invoerSpec)+" fails."
        MsgBox(opm)
        return false
    end
```

```
else
```

```
    aBestand=aBestand+1
```

```
end
```

```
next
return true
End Function
```

KopieerVenster.Bron_Open:

```
Private Sub Bron_Open()
    ST_Bron.AcceptFileDrop("special/disk")
    ST_Bron.AcceptFileDrop("special/folder")
End Sub
```

KopieerVenster.Doel_Open:

```
Private Sub Doel_Open()
    ST_Doel.AcceptFileDrop("special/disk")
    ST_Doel.AcceptFileDrop("special/folder")
End Sub
```

KopieerVenster.KopieerSpecNaarSpec:

```
Private Function KopieerSpecNaarSpec(invoerSpec as FolderItem, uitvoerSpec as FolderItem) As boolean
    dim uitvoer as BinaryStream
    dim ok as boolean

    uitvoer=uitvoerSpec.CreateBinaryFile("text")
    if uitvoer=nil then
        return false
    end

    ok=KopieerSpecNaarStream(invoerSpec, uitvoer)

    uitvoer.Close()

    if not ok then
        return false
    end

    uitvoerSpec.macType=invoerSpec.macType
    uitvoerSpec.macCreator=invoerSpec.macCreator

    return true
End Function
```

KopieerVenster.KopieerSpecNaarStream:

```
Private Function KopieerSpecNaarStream(invoerSpec as FolderItem, uitvoer as BinaryStream) As boolean
    dim lengte as integer
    dim invoer as BinaryStream
    dim ok as boolean

    lengte=invoerSpec.length
```

```
if lengte=0 then
  return true
end
```

```
invoer=invoerSpec.OpenAsBinaryFile()
if invoer=nil then
  return false
end
```

```
ok=KopieerStreamNaarStream(lengte,invoer,uitvoer)
```

```
invoer.Close()
```

```
return ok
```

```
End Function
```

KopieerVenster.KopieerStreamNaarStream:

```
Private Function KopieerStreamNaarStream(lengte as integer,invoer as BinaryStream,uitvoer as BinaryStream) As boolean
```

```
  dim aBlok as integer
  dim blokLengte as integer
  dim rest as integer
  dim i as integer
  dim blok as string
```

```
  aBlok=Bitwise.ShiftRight(lengte,MACHT)
  blokLengte=Bitwise.ShiftLeft(1,MACHT)
  rest=Bitwise.BitAnd(lengte,blokLengte-1)
```

```
  if aBlok>0 then
    for i=1 to aBlok
      blok=invoer.Read(blokLengte)
      if invoer.lastErrorCode<>0 then
        return false
      end
```

```
      uitvoer.Write(blok)
      if uitvoer.lastErrorCode<>0 then
        return false
      end
    next
  end
```

```
  if rest>0 then
    blok=invoer.Read(rest)
    if invoer.lastErrorCode<>0 then
      return false
    end
```

```
uitvoer.Write(blok)
if uitvoer.lastErrorCode<>0 then
  return false
end
end
```

```
return true
End Function
```

KopieerVenster.PB_Start.Action:

```
Sub Action()
  Action_Start()
End Sub
```

KopieerVenster.PB_Cancel.Action:

```
Sub Action()
  Action_Cancel()
End Sub
```

KopieerVenster.ST_Doel.Open:

```
Sub Open()
  Doel_Open()
End Sub
```

KopieerVenster.ST_Bron.Open:

```
Sub Open()
  Bron_Open()
End Sub
```