
Article

[Alberto Fuentes](#) · Feb 18, 2021 2m read

Simple tool to download interoperability messages from your browser

Hi Community!

I'm sharing a little tool (REST service) to download interoperability messages from your browser.

You only need to:

1. Create a web application in Management Portal (e.g. /downloadmsg) and set DispatchClass=Util.DownloadMsg.
2. Call the tool using your browser passing the namespace and the message header id to download.

<http://localhost:52773/downloadmsg/ns/mydev/msgid/17441>

```
///  
/// Util to download messages given a message header id  
///  
/// Setup:  
/// 1. Create a webapplication (e.g. /downloadmsg) and set DispatchClass=Util.Downloa  
dMsg  
/// 2. Go to http://localhost:52773/downloadmsg/ns/user/msgid/19 to download the mess  
age reference by header 19 in namespace user  
Class Util.DownloadMsg Extends %CSP.REST  
{  
  
XData UrlMap [ XMLNamespace = "http://www.intersystems.com/urlmap" ]  
{  
<Routes>  
    <Route Url="/ns/:ns/msgid/:msgid" Method="GET" Call="DownloadMessage"/>  
</Routes>  
}  
  
/// Download a message given a Ens.MessageHeader id  
ClassMethod DownloadMessage(ns As %String, msgId As %String = "") As %Status  
{  
    set ret = $$$OK  
    set currentNs = $namespace  
  
    try {  
        set $namespace = ns  
        $$$ThrowOnError(..ExportMsgToStream(msgId, .stream, .filename))  
  
        // set headers to download stream as filename  
        do %response.SetHeader("Content-Type", "application/octet-stream")  
        do %response.SetHeader("Content-  
Disposition", "attachment; filename=""_filename_""")  
        do %response.SetHeader("Content-Length", stream.Size)  
        do stream.Rewind()  
        do stream.OutputToDevice()  
  
    } catch ex {  
        set ret = ex.AsStatus()  
    }  
}
```

```
// restore namespace
set $namespace = currentNs

quit ret
}

/// Export a message to stream given a Ens.MessageHeader id
ClassMethod ExportMsgToStream(msgId As %String, Output stream As %Stream.Object, Output filename As %String) As %Status
{
    set ret = $$$OK
    try {
        // stream
        set stream = ##class(%Stream.GlobalCharacter).%New()
        set filename = ""

        // message header
        set headerObj = ##class(Ens.MessageHeader).%OpenId(msgId, .sc)
        $$$ThrowOnError(sc)

        // message body
        set obj = $classmethod(headerObj.MessageBodyClassName, "%OpenId", headerObj.MessageBodyId)
        set classname = $classname(obj)

        // output to stream
        if classname="EnsLib.HL7.Message" {
            set sc = $method(obj, "OutputToLibraryStream", .stream)
            $$$ThrowOnError(sc)
            set filename = msgId_".hl7"
        }
        else {
            set writer = ##class(%XML.Writer).%New()
            set writer.Indent=1
            set writer.NoXMLDeclaration=1
            $$$ThrowOnError(writer.OutputToStream(.stream))
            $$$ThrowOnError(writer.RootObject(obj))
            set filename = msgId_".xml"
        }
    } catch ex {
        set ret = ex.AsStatus()
    }
    quit ret
}
}
```

[#HL7](#) [#Interoperability](#) [#Tools](#) [#XML](#) [#Ensemble](#) [#InterSystems IRIS for Health](#)

Source

URL: <https://community.intersystems.com/post/simple-tool-download-interoperability-messages-your-browser>