Artículo
Bernardo Linarez · 22 nov, 2021 Lectura de 3 min

Personalización de un Business Service core de IRIS/Ensemble

Mas de una vez nos ha sucedido que requerimos añadir algún comportamiento a algún *Bussiness Service* core (tal como viene "de caja").

En este caso puntual, tomaré de ejemplo el BS *EnsLib.RecordMap.Service.FileService,* el cual nos permite leer un archivo (normalmente un csv) desde una carpeta configurable. A veces sucede que el archivo que estamos leyendo es muy grande, y por tanto contiene muchas filas, y nuestra lógica requiera saber exactamente cuando se termino de procesar y además generar un evento que incluso sea procesado por algún Host personalizado.

Entonces, necesitamos personalizar el comportamiento. Para esto se puede utilizar el mecanismo de herencia de IRIS, soportado por ObjectScript. En este caso de uso, podemos intervenir el método *OnProcessInput*, de la siguiente manera:

```
Class Test.FS.BS.Service Extends EnsLib.RecordMap.Service.FileService
{
Method OnProcessInput(pInput As %Stream.Object, Output pOutput As %RegisteredObject,
ByRef pHint As %String) As %Status
{
    Set tSC = $$$OK
    Try
    {
        Set tSC = ##super(pInput,pOutput,.pHint)
    }
    Catch(e)
    {
        Set tSC = e.AsStatus()
    }
    Quit tSC
}
}
```

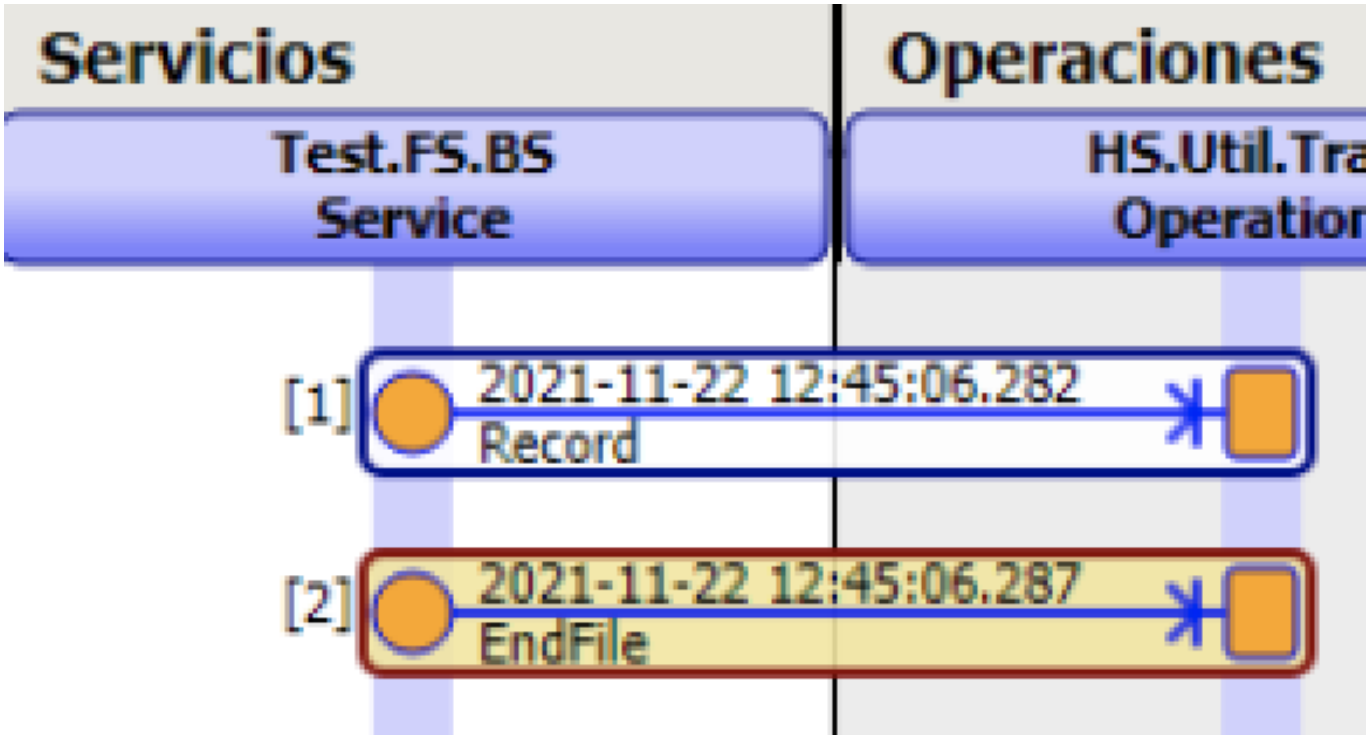
Con esta herencia, gracias al método *##super*, tenemos exactamente el mismo comportamiento que el BS core. Notar que respetamos la definición del método core, que se manifiesta en la misma descripción de los parámetros.

Entonces, dado que queremos enviar un evento personalizado cuando el archivo termine de procesarse, podemos realizar lo siguiente:

```
Class Test.FS.BS.Service Extends EnsLib.RecordMap.Service.FileService
{

Method OnProcessInput(pInput As %Stream.Object, Output pOutput As %RegisteredObject,
ByRef pHint As %String) As %Status
{
    Set tSC = $$$OK
    Try
    {
        Set tSC = ##super(pInput,pOutput,.pHint)
        Quit:$System.Status.IsError(tSC)
```

```
        //Mensaje personalizado.
        Set objEndFile = ##Class(Test.FS.Msg.EndFile).%New()

        //Envío de mensaje personalizado al Host respectivo
        Set tSC = ..SendRequestAsync("Host",objEndFile)
        Quit:$System.Status.IsError(tSC)
    }
    Catch(e)
    {
        Set tSC = e.AsStatus()
    }
    Quit tSC
}

}
```

Entonces en la traza generada por la ejecución de nuestro BS, obtenemos algo como esto:



Notar que escribimos el código personalizado después de la llamada a la sentencia *##super,* dado que bajo nuestra lógica, la ejecución del BS core tal cuál es garantía de que se leyó correctamente el archivo. Sin embargo, en otros casos de uso, se podría intervenir antes de la llamada a la lógica core.

Este mismo mecanismo se puede usar en otras *Host* proporcionados por IRIS.

Espero que les sea de utilidad.

#Ensemble #InterSystems IRIS #InterSystems IRIS for Health

URL de fuente:https://es.community.intersystems.com/post/personalizaci%C3%B3n-de-un-business-service-core-de-irisensemble