

Artículo

[Nancy Martínez](#) · 28 jun, 2019 Lectura de 2 min

Cómo evitar el truncamiento en mensajes HL7 con campos muy grandes (más de 32K)

¡Hola Comunidad!

Las transformaciones DTL y las llamadas a los métodos `GetValueAt/SetValueAt` en los mensajes HL7 pueden truncar los campos que tengan más de 32K. Para evitar los truncamientos en el caso en que los campos excedan los 32K, es posible utilizar los métodos `GetFieldStreamRaw` y `StoreFieldStreamRaw`. Un ejemplo frecuente en el que sucede el truncamiento, es el segmento `OBX:5`. Se recomienda que estos métodos se utilicen cuidadosamente, teniendo en cuenta los detalles específicos que sean requeridos.

Estos métodos no es posibles utilizarlos, de manera habitual arrastrándolos de izquierda a derecha en un DTL. Debe hacerse con una implementación a nivel del código. Además, cuando se llame al método `StoreFieldStreamRaw` se debe realizar una revisión al segmento y si es necesario, hacer una última edición al segmento ya que después de eso, este ya no podrá modificarse.

El método `StoreFieldStreamRaw` coge 3 argumentos: La secuencia que se coloca en el campo, la ruta VDoc del campo en el que se almacenará la secuencia y un argumento `pRemainder`. Si no se especifica el argumento `pRemainder`, entonces todos los campos que se encuentren después del campo que se está almacenando desaparecerán. Si se especifica `pRemainder`, entonces todos los campos que se encuentren después del que se está almacenando serán sustituidos por lo que esté en `pRemainder`.

Ejemplo:

```
OBX|1|2|3|4|5|6|7
after StoreFieldStreamRaw(stream, "OBX:5")
OBX|1|2|3|4|<stream>
(notice fields 6 and 7 are gone).
after StoreFieldStreamRaw(stream, "OBX:5", "|six|seven"):
OBX|1|2|3|4|<stream>|six|seven
```

De la misma forma, el método `GetFieldStreamRaw` coge 3 argumentos: el argumento de salida de la secuencia, la ruta VDoc del campo y el argumento de salida `pRemainder`. `pRemainder` se llenará con todos los campos que se encuentren después del que se extrae.

Ejemplo:

```
OBX|1|2|3|4|5|6|7
after GetFieldStreamRaw(stream, "OBX:5", .rem)
rem contains "|6|7"
```

Entonces, en resumen, queremos extraer el campo desde la fuente y almacenarlo en el objetivo. Pero no queremos tomar todo lo demás desde la fuente y almacenarlo en el objetivo, porque esto revertirá cualquier edición que ya hayamos hecho en los campos después de colocar el campo más largo en el objetivo. Por lo tanto, lo que queremos es tomar el campo de la secuencia desde la fuente, pero todo lo demás tomarlo desde el objetivo y almacenar ambas cosas en el objetivo de la siguiente forma:

```
do source.GetFieldStreamRaw(.stream, "OBX:5")
do target.GetFieldStreamRaw(.dummy, "OBX:5", .rem)
do target.StoreFieldStreamRaw(stream, "OBX:5", rem)
```

Si lo hacemos de esta manera, evitaremos que las ediciones que hayamos realizado anteriormente en los campos, después de utilizar OBX:5 en el objetivo, se reviertan o se eliminen.

[#Code Snippet](#) [#DTL](#) [#HL7](#) [#Ensemble](#)

URL de

fuelle: <https://es.community.intersystems.com/post/c%C3%B3mo-evitar-el-truncamiento-en-mensajes-hl7-con-campos-muy-grandes-m%C3%A1s-de-32k>