Question
Thembelani Mlalazi · May 18, 2018

Customising the SimpleBatch GetBatchheader Method

I have a csv record mapper that reads in a batch and my batch has a header and trailer that changes all the time but it is of fixed length so I override the Get Batch Header method from the simplebatch class to something like below but my problem is when the batch is read in It still tries to bring in another empty batch like the picture below any reason why that is here is my code.The red code is the changes made. After several trace logs it is being called twice first time with the batch [1] and[5]is the second call how can I stop that from happening

```
/// Get the Batch Header from an incoming stream. This is only invoked once by the Re
cordMap
/// batch services when a new stream is passed in. Any extra data must be returned in
 pLookAhead
/// for use in parsing the first record.
ClassMethod GetBatchHeader(pIOStream As %IO.DeviceStream, pTimeout As %Numeric = -1,
Output pBatch As EnsLib.RecordMap.SimpleBatch, ByRef pLookAhead As %String) As %Statu
s
{
 Try {
   Set tStatus = $$$OK
   Set pBatch = ""
   Set tTerm = ..GetHeaderTerm()
   //Set tFullHeader = 63 _ tTerm
   Set tHeaderLen =63+$length(tTerm)
  If tHeaderLen
    {
   Set tFound = 0
   Set tLeadingJunk = ""
   Set pLookAhead = $get(pLookAhead)
   Set tTimeout = pTimeout
   Set tEndTime = $zhorolog + pTimeout
   While ('tFound) && ('pIOStream.AtEnd)
     {
     Set tReadLen = tHeaderLen - $length(pLookAhead)
     If tReadLen > 0
        {
        Set tData = pLookAhead _ pIOStream.Read(tReadLen, .tTimeout, .tStatus)
        $$$TRACE("1")
        If $$$ISERR(tStatus) Quit
        If tTimeout
          {
         Set tStatus = $$$ERROR($$$EnsErrTCPReadTimeoutExpired, pTimeout, tReadLen)
         Quit
          }
         Set pLookAhead = ""
       }
    Else
      {
       $$$TRACE("2")
```

```
      Set tData = $extract(pLookAhead, 1, tHeaderLen)
      Set pLookAhead = $extract(pLookAhead, tHeaderLen + 1, *)
      }
      If ($extract(tData,1,3) = "001" )
         {
        Set pBatch = ..%New()
        set pBatch.BatchHeader = tData
        Set tFound = 1
        $$$TRACE("Foundlee"_tFound)
        Quit
        }
        Else
          {
           $$$TRACE("3")
         Set pLookAhead = pLookAhead _ tData
          #; Check if we should start discarding leading data
         If ($length(pLookAhead) >= tHeaderLen)
            {
          If ($length(tLeadingJunk) < 400)
             {
            Set tLeadingJunk = tLeadingJunk _ $extract(pLookAhead,1)
             }
             Set pLookAhead = $extract(pLookAhead,2,*)
            }

          }
          If (pTimeout = -1)
             {
            Set tTimeout = -1
             }
    Else {
     Set tCurrTime = $zhorolog
     If (tCurrTime > tEndTime) {
      Set tStatus = $$$ERROR($$$EnsErrTCPReadTimeoutExpired, pTimeout, tReadLen)
      Quit
     }
     Set tTimeout = tEndTime - tCurrTime
    }
   } //while
   If $$$ISERR(tStatus) Quit
   #; Clear the lookahead buffer if we didn't find the batch header
   $$$TRACE("###Status###"_tStatus)
   If (('tFound) && ($length(tLeadingJunk) < 400)&&($$$ISERR(tStatus))){
    Set tLeadingJunk = tLeadingJunk _ $get(pLookAhead)
    Set pLookAhead = ""
    $$$TRACE("###Not Found###"_tFound)
   }
   If (tLeadingJunk '= "") && ('..#IgnoreLeadingData) {
    #; Use JS escaping to handle control characters
    Set tLoggedJunk = $zconvert($extract(tLeadingJunk,1,400),"O","JS") _ $select($len
gth(tLeadingJunk) > 400: "...", 1: "")
    $$$LOGWARNING($$$FormatText($$$Text("Discarding unexpected leading data: '%1'","E
nsemble"),tLoggedJunk))
   }
   If (('tFound)&&($$$ISERR(tStatus))) {
    Set pBatch = ""
    $$$TRACE("Not Found")
    Set tStatus = $$$ERROR($$$EnsRecordMapErrBatchHeaderNotFound,$classname($this))
    Quit
```

```
   }
  }
  Else {
   Set pBatch = ..%New()
   Set pLookAhead = $get(pLookAhead)
   Quit
  }
 }
 Catch ex {
  Set tStatus = $$$EnsSystemError
 }
 Quit tStatus
}
```

#Business Service #Object Data Model #Ensemble

Source URL:https://community.intersystems.com/post/customising-simplebatch-getbatchheader-method