

Article

[Dmitry Zasyepkin](#) · May 22, 2020 2m read

Message Viewer Filters To Support Stream Properties

If you work with interoperability productions of InterSystems IRIS or Ensemble, no doubt you are familiar with the Message Viewer page. The page supports filtering messages according to filter criteria you enter in the Basic and/or Extended Criteria sections. Extended Criteria conditions are specified as property-operator-value triples. Once you click Search button, such triples become WHERE clause conditions of a generated SQL query executed against message header/body tables.

As a side note, one can view recently generated message queries by following the instructions listed in the [doc](#).

Unfortunately stream properties cannot be used in the filtering conditions since most of the SQL WHERE predicates (including LIKE) are [applied to OID of a stream - not to its contents](#). But there is a way to overcome that limitation *to some extent* using a simple two-step approach:

1) Subclass [EnsPortal.MsgFilter.Assistant](#) as follows

```
Class is cru.interop.MsgFilterAssistant Extends EnsPortal.MsgFilter.Assistant
```

```
{
ClassMethod GetSQLCondition(pOperator As %String, pProp As %String, pValue As %String, pDisplay As
%Boolean = 0, pNoIndex As %Boolean = 0) As %String
{
if (pValue = "") || ((pOperator = "Like") &&
(pOperator = "NotLike")) quit ##super(pOperator, pProp, pValue, pDisplay, pNoIndex)

if ("%%" = $extract(pValue, *-2, *-1))
{
set pValue = "" _ $extract(pValue, 1, *-3) _ " ESCAPE " _ $extract(pValue, *) _ ""
}
else
{
set pValue = "" _ $replace(pValue, "'", "''") _ ""
}
}
```

```
quit "substring(" _ pProp _ ", 1, 3000000) " _ $case(pOperator, "Like": "LIKE", "NotLike": "NOT LIKE") _ "  
" _ pValue  
}  
}
```

2) Execute the following command in the current namespace:

```
set ^EnsPortal.Settings("MessageViewer", "AssistantClass") = "<name of the class created in step 1>"
```

Now you can apply Like/NotLike conditions to stream properties of message bodies with one limitation: only the first 3 mln bytes of a stream are taken into account when applying Like/NotLike pattern. Anyway this is more of a proof-of-concept solution that does not "scale", but still works well in some cases.

If the 3M limitation is not acceptable then it might make sense to look into implementing Like operator with stream argument, wrapping it in a stored procedure method, and using that procedure for WHERE subclause in the code above. Perhaps that can be a good topic for another article...

[#Interoperability](#) [#Message Search](#) [#Ensemble](#) [#InterSystems IRIS](#)

Source URL: <https://community.intersystems.com/post/message-viewer-filters-support-stream-properties>