Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
1   |********************************************************************************
2   |* txesbboncm101   0  VRC B61O a  ext
3   |* Extension Script for BOD DefectiveMaterialNoticeInBOD
4   |* gritter
5   |* 08/31/26 [11:20]
6   |********************************************************************************
7   |* Script Type: Library
8   |* WARNING: Do NOT edit this file using the Baan Tools (SCM) !!!
9   |********************************************************************************
10
11  | Hook declarations
12  #context_on "declarations"
13  |global declarations for extension script.
14  #include<bic_dam>
15  #context_off
16
17  | Additional generated declarations
18  table   tqmncm100
19  extern  boolean is.outbound
20
21  #pragma used dll "otlbctinterface"
22
23  #define initialize()
24  ^       retl = tlbctinterface.cust.initialize(o.xml)
25  ^       if retl <> 0 then
26  ^           return(retl)
27  ^       endif
28  ^       import("is.outbound", is.outbound)
29  ^       if not is.outbound then
30  ^           retl = xmlSetAttribute(o.xml, "library", "otxesbboncm101")
31  ^           retl = xmlSetAttribute(o.xml, "path", i.component.path)
32  ^       endif
33
34  #define addTableField(elementName, tableCode, columnName, dataType)
35  ^       retl = tlbctinterface.cust.add.element(o.xml, elementName,
36  ^                                   tableCode, columnName, dataType)
37  ^       if retl <> 0 then
38  ^           return(retl)
39  ^       endif
40
41  |Note: the macro below can only be used in ES 10.4.1 and later.
42  #define addEnumTableField(elementName, tableCode, columnName, domainCode)
43  ^       retl = tlbctinterface.cust.add.element(o.xml, elementName,
44  ^                           tableCode, columnName, "enum", domainCode)
45  ^       if retl <> 0 then
46  ^           return(retl)
47  ^       endif
48
49  #define addValue(elementName, description, dataType)
50  ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, description,
51  ^                           dataType)
52  ^       if retl <> 0 then
53  ^           return(retl)
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
54      ^       endif
55
56      #define addAmountValue(elementName, value, currencyid)
57      ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, value, "amount"
        ,
58      ^               "currencyID", currencyid)
59      ^       if retl <> 0 then
60      ^               return(retl)
61      ^       endif
62
63      #define addCodeValue(elementName, value, listid, accountingentity)
64      ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, value, "code",
65      ^               "listID", listid, "accountingEntity", accountingentity)
66      ^       if retl <> 0 then
67      ^               return(retl)
68      ^       endif
69
70      #define addHierarchyReferenceValue(elementName, value, nodeid)
71      ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, value, "hierarc
        hyReference",
72      ^               "nodeID", nodeid)
73      ^       if retl <> 0 then
74      ^               return(retl)
75      ^       endif
76
77      #define addMasterDataReferenceValue(elementName, value, nounname, accountingenti
        ty)
78      ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, value, "masterD
        ataReference",
79      ^               "nounName", nounname, "accountingEntity", accountingentity)
80      ^       if retl <> 0 then
81      ^               return(retl)
82      ^       endif
83
84      #define addQuantityValue(elementName, value, unitcode)
85      ^       retl = tlbctinterface.cust.add.value(o.xml, elementName, value, "quantit
        y",
86      ^               "unitCode", unitcode)
87      ^       if retl <> 0 then
88      ^               return(retl)
89      ^       endif
90
91      #define addDescription(value)
92      ^       retl = xmlSetAttribute(xmlGetLastChild(o.xml), "Description", value)
93
94      #define addEffectiveTimePeriod(startDateTime, endDateTime)
95      ^       retl = xmlSetAttribute(xmlGetLastChild(o.xml), "StartDateTime", startDat
        eTime)
96      ^       retl = xmlSetAttribute(xmlGetLastChild(o.xml), "EndDateTime", endDateTim
        e)
97
98      #define addXml(xml)
99      ^       retl = tlbctinterface.cust.add.xml(o.xml, xml)
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
100     ^       if retl <> 0 then
101     ^               return(retl)
102     ^       endif
103
104     #define getIdentifierValueFromIdentifierStructure(xml, table, column)
105     ^       tlbctinterface.cust.get.value.from.identifier.structure(xml, table, colu
        mn)
106
107     #define getIdentifierDataTypeFromIdentifierStructure(xml, table, column)
108     ^       tlbctinterface.cust.get.datatype.from.identifier.structure(xml, table, c
        olumn)
109
110     #define removeIdentifierStructure(xml)
111     ^       retl = xmlDelete(xml)
112     ^       xml = 0
113
114     #define getTableIdentifiers.DefectiveMaterialNoticeInBOD(xml)
115     ^       retl = tlbctinterface.cust.get.component.table.identifiers(
116     ^                       "boncm101", "DefectiveMaterialNoticeInBOD", xml)
117     ^       if retl <> 0 then
118     ^               xml = 0
119     ^               return(retl)
120     ^       endif
121
122     function extern long get.additional.elements(
123             const   string  i.component.path,
124             ref     long    o.xml)
125     {
126             FunctionUsage
127             Desc:   Define the additional elements (columns) for a component.
128             Input:  i.component.path - the component path.
129             Output: o.xml - if return value is = 0: an xml structure containing
130                             the details on custom fields (if any)
131             Return: 0 (OK) or DALHOOKERROR (one or more DAL error
132                             messages are set)
133             EndFunctionUsage
134
135             long    retl            | return value to be checked
136
137             initialize() | do not remove this line
138
139             on case i.component.path
140             case "DefectiveMaterialNoticeInBOD":
141                     addTableField("ff.as1d", "qmncm100", "cdf_as1d", "Date")
142
143     #pragma used field "qmncm100.cdf_as1d"
144                     addTableField("ff.as2d", "qmncm100", "cdf_as2d", "Date")
145
146     #pragma used field "qmncm100.cdf_as2d"
147                     addTableField("ff.as3d", "qmncm100", "cdf_as3d", "Date")
148
149     #pragma used field "qmncm100.cdf_as3d"
150                     addTableField("ff.buyid", "qmncm100", "cdf_byee", "String")
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
151
152     #pragma used field "qmncm100.cdf_byee"
153             addTableField("ff.buyname", "qmncm100", "cdf_bynm", "String")
154
155     #pragma used field "qmncm100.cdf_bynm"
156             addTableField("ff.cm1d", "qmncm100", "cdf_cm1d", "Date")
157
158     #pragma used field "qmncm100.cdf_cm1d"
159             addTableField("ff.cm2d", "qmncm100", "cdf_cm2d", "Date")
160
161     #pragma used field "qmncm100.cdf_cm2d"
162             addTableField("ff.cm3d", "qmncm100", "cdf_cm3d", "Date")
163
164     #pragma used field "qmncm100.cdf_cm3d"
165             addTableField("ff.engid", "qmncm100", "cdf_enee", "String")
166
167     #pragma used field "qmncm100.cdf_enee"
168             addTableField("ff.engname", "qmncm100", "cdf_ennm", "String")
169
170     #pragma used field "qmncm100.cdf_ennm"
171             addTableField("ff.qenid", "qmncm100", "cdf_qeee", "String")
172
173     #pragma used field "qmncm100.cdf_qeee"
174             addTableField("ff.qenname", "qmncm100", "cdf_qenm", "String")
175
176     #pragma used field "qmncm100.cdf_qenm"
177             break
178         default:
179             break
180         endcase
181         return(0)
182     }
183
184     function long getPropertyByNumber(long user.area, long nr)
185     {
186         long    property
187         long    i
188
189         if false then
190             | prevent compilation warning
191             i = getPropertyByNumber(0,0)
192         endif
193
194         if nr < 1 or
195           xmlGetNumChilds(user.area) < 1 or
196            xmlGetNumChilds(user.area) < nr then
197             return(0)
198         endif
199
200         property = xmlGetFirstChild(user.area)
201         for     i = 2
202         to      nr
203             property = xmlGetRightSibling(property)
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
204         endfor
205
206         return(property)
207     }
208
209     function extern long process.inbound.user.area(
210         const  string  i.component.path,
211         ref    long    i.xml)
212     {
213         FunctionUsage
214         Desc:  Processes the data from the UserArea node of an inbound BOD.
215         Input: i.component.path - the component path.
216                i.xml - the UserArea node from the BOD component.
217         Return: 0 (OK) or DALHOOKERROR (one or more DAL error messages are set)
218         EndFunctionUsage
219
220     #define getFirstProperty()
221     ^       xmlGetFirstChild(i.xml)
222     | This returns the first property from the inbound user area
223
224     #define getNextProperty(property)
225     ^       xmlGetRightSibling(property)
226     | This returns the next property from the inbound user area
227
228     #define getNrProperties()
229     ^       xmlGetNumChilds(i.xml)
230     | This returns the number of properties from the inbound user area
231
232     #define getPropertyNr(offset)
233     ^       getPropertyByNumber(i.xml, offset)
234     | This returns the property with the offset
235
236     #define getNamedProperty(attr.name)
237     ^       xmlGetParent(xmlFindFirstMatch(
238     ^               "<>.<Property>.<NameValue name=""" & attr.name & """>", i.xml))
239     | This returns the first property with specified name
240
241     #define getPropertyName(property)
242     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "name")
243     | This returns the name of the property
244
245     #define getPropertyType(property)
246     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "type")
247     | This returns the type of the property
248
249     #define getPropertyValue(property)
250     ^       xmlData$(xmlFindFirst("NameValue", property))
251     | This returns the value of the property
252
253     #define getAccountingEntity(property)
254     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "accountingEntity")
255     | This returns the accountingEntity of the property with type CodeType or
256     | MasterDataReferenceType
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
257
258     #define getCurrencyID(property)
259     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "currencyID")
260     | This returns the currencyID of the property with type AmountType
261
262     #define getListID(property)
263     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "listID")
264     | This returns the listID of the property with type CodeType
265
266     #define getNodeID(property)
267     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "nodeID")
268     | This returns the nodeID of the property with type HierachyReferenceType
269
270     #define getNounName(property)
271     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "nounName")
272     | This returns the nounName of the property with type MasterDataReferenceType
273
274     #define getUnitCode(property)
275     ^       xmlAttribute$(xmlFindFirst("NameValue", property), "unitCode")
276     | This returns the unitCode of the property with type QuantityType
277
278     #define getDescription(property)
279     ^       xmlData$(xmlFindFirst("Description", property))
280     | This returns the description of the property
281
282     #define getStartDateTime(property)
283     ^       xmlData$(xmlFindFirst("StartDateTime", property))
284     | This returns the start date-time of the property
285
286     #define getEndDateTime(property)
287     ^       xmlData$(xmlFindFirst("EndDateTime", property))
288     | This returns the end date-time of the property
289
290     #define getUserAreaParent()
291     ^       xmlGetParent(i.xml)
292     | This returns the parent of the UserArea
293
294             long    retl            | return value to be checked
295
296             retl = 0
297
298             on case i.component.path
299             case    "DefectiveMaterialNoticeInBOD":
300     | Hook DefectiveMaterialNoticeInBOD.process.inbound.user.area
301     #context_on "DefectiveMaterialNoticeInBOD.process.inbound.user.area"
302     |* Use this hook to process the data from the UserArea node of an inbound BOD.
303     |* GSR 07/13/26 Added CDFs for Buyer tracking per CMT 5897
304
305                     long            l.header.xml
306                     long            l.ret1
307                     long            userarea.property.node.hdr
308             domain  tcdate          l.as1d          |* Assigned 1 Date
309             domain  tcdate          l.as2d          |* Assigned 2 Date
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
310         domain  tcdate      l.as3d      |* Assigned 3 Date
311         domain  tcdate      l.cm1d      |* Completed 1 Date
312         domain  tcdate      l.cm2d      |* Completed 2 Date
313         domain  tcdate      l.cm3d      |* Completed 3 Date
314         domain  tcemno      l.buy.id    |* Buyer ID
315         domain  tcnama      l.buy.name  |* Buyer Name
316         domain  tcemno      l.eng.id    |* Engineer ID
317         domain  tcnama      l.eng.name  |* Engineer Name
318         domain  tcemno      l.qen.id    |* Quality Engineer ID
319         domain  tcnama      l.qen.name  |* Quality Engineer Name
320         domain  tcorno      l.ncmr.no   |* NCMR Number
321         domain  qmncm.rsst  l.rsst      |* Response Status
322
323         l.ret1 = 0
324         l.header.xml = 0
325         l.as1d = 0
326         l.as2d = 0
327         l.as3d = 0
328         l.cm1d = 0
329         l.cm2d = 0
330         l.cm3d = 0
331         l.buy.id = ""
332         l.buy.name = ""
333         l.eng.id = ""
334         l.eng.name = ""
335         l.qen.id = ""
336         l.qen.name = ""
337         l.rsst = qmncm.rsst.rec
338
339         l.ret1 = getTableIdentifiers.DefectiveMaterialNoticeInBOD(l.header.xml)
340         l.ncmr.no = getIdentifierValueFromIdentifierStructure(l.header.xml,
341                                         "qmncm100","ncmr")
342
343         userarea.property.node.hdr = 0
344         userarea.property.node.hdr = getNamedProperty("ff.as1d")
345         if userarea.property.node.hdr <> 0 then
346             l.as1d = lval(getPropertyValue(userarea.property.node.hdr))
347         endif
348
349         userarea.property.node.hdr = 0
350         userarea.property.node.hdr = getNamedProperty("ff.as2d")
351         if userarea.property.node.hdr <> 0 then
352             l.as2d = lval(getPropertyValue(userarea.property.node.hdr))
353         endif
354
355         userarea.property.node.hdr = 0
356         userarea.property.node.hdr = getNamedProperty("ff.as3d")
357         if userarea.property.node.hdr <> 0 then
358             l.as3d = lval(getPropertyValue(userarea.property.node.hdr))
359         endif
360
361         userarea.property.node.hdr = 0
362         userarea.property.node.hdr = getNamedProperty("ff.cm1d")
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
363         if userarea.property.node.hdr <> 0 then
364             l.cm1d = lval(getPropertyValue(userarea.property.node.hdr))
365         endif
366
367         userarea.property.node.hdr = 0
368         userarea.property.node.hdr = getNamedProperty("ff.cm2d")
369         if userarea.property.node.hdr <> 0 then
370             l.cm2d = lval(getPropertyValue(userarea.property.node.hdr))
371         endif
372
373         userarea.property.node.hdr = 0
374         userarea.property.node.hdr = getNamedProperty("ff.cm3d")
375         if userarea.property.node.hdr <> 0 then
376             l.cm3d = lval(getPropertyValue(userarea.property.node.hdr))
377         endif
378
379         userarea.property.node.hdr = 0
380         userarea.property.node.hdr = getNamedProperty("ff.buyid")
381         if userarea.property.node.hdr <> 0 then
382             l.buy.id = getPropertyValue(userarea.property.node.hdr)
383         endif
384
385         userarea.property.node.hdr = 0
386         userarea.property.node.hdr = getNamedProperty("ff.buyname")
387         if userarea.property.node.hdr <> 0 then
388             l.buy.name = getPropertyValue(userarea.property.node.hdr)
389         endif
390
391         userarea.property.node.hdr = 0
392         userarea.property.node.hdr = getNamedProperty("ff.engid")
393         if userarea.property.node.hdr <> 0 then
394             l.eng.id = getPropertyValue(userarea.property.node.hdr)
395         endif
396
397         userarea.property.node.hdr = 0
398         userarea.property.node.hdr = getNamedProperty("ff.engname")
399         if userarea.property.node.hdr <> 0 then
400             l.eng.name = getPropertyValue(userarea.property.node.hdr)
401         endif
402
403         userarea.property.node.hdr = 0
404         userarea.property.node.hdr = getNamedProperty("ff.qenid")
405         if userarea.property.node.hdr <> 0 then
406             l.qen.id = getPropertyValue(userarea.property.node.hdr)
407         endif
408
409         userarea.property.node.hdr = 0
410         userarea.property.node.hdr = getNamedProperty("ff.qenname")
411         if userarea.property.node.hdr <> 0 then
412             l.qen.name = getPropertyValue(userarea.property.node.hdr)
413         endif
414
415         select  qmncm100.*
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
416         from    qmncm100 for update
417         where   qmncm100._index1 = {:l.ncmr.no}
418         as set with 1 rows
419         selectdo
420             l.ret1 = dal.change.object("qmncm100")
421             if l.ret1 <> 0 then
422                 return(DALHOOKERROR)
423             endif
424             if l.as1d <> 0 then
425                 dal.set.field("qmncm100.cdf_as1d",l.as1d)
426             endif
427             if l.as2d <> 0 then
428                 dal.set.field("qmncm100.cdf_as2d",l.as2d)
429             endif
430             if l.as3d <> 0 then
431                 dal.set.field("qmncm100.cdf_as3d",l.as3d)
432             endif
433             if l.cm1d <> 0 then
434                 dal.set.field("qmncm100.cdf_cm1d",l.cm1d)
435             endif
436             if l.cm2d <> 0 then
437                 dal.set.field("qmncm100.cdf_cm2d",l.cm2d)
438             endif
439             if l.cm3d <> 0 then
440                 dal.set.field("qmncm100.cdf_cm3d",l.cm3d)
441             endif
442             if not isspace(l.buy.id) then
443                 dal.set.field("qmncm100.cdf_byee",l.buy.id)
444             endif
445             if not isspace(l.buy.name) then
446                 dal.set.field("qmncm100.cdf_bynm",l.buy.name)
447             endif
448             if not isspace(l.eng.id) then
449                 dal.set.field("qmncm100.cdf_enee",l.eng.id)
450             endif
451             if not isspace(l.eng.name) then
452                 dal.set.field("qmncm100.cdf_ennm",l.eng.name)
453             endif
454             if not isspace(l.qen.id) then
455                 dal.set.field("qmncm100.cdf_qeee",l.qen.id)
456             endif
457             if not isspace(l.qen.name) then
458                 dal.set.field("qmncm100.cdf_qenm",l.qen.name)
459             endif
460
461             dal.set.field("qmncm100.rsst",l.rsst)
462
463             l.ret1 = dal.save.object("qmncm100")
464             if l.ret1 <> 0 then
465                 return(DALHOOKERROR)
466             endif
467         endselect
468     #context_off
```

Prog.Scr/Libr. : tx esb boncm101 Extension Script for BOD DefectiveMaterialNoticeInBOD
VRC : B61O a ext User : gritter Date : 06/25/26 09:58:35 Exp. : No
TIV : 2590
Compile Flags :

```
469
470             break
471         default:
472             break
473         endcase
474         return(0)
475
476     #undef  getFirstProperty()
477     #undef  getNextProperty(property)
478     #undef  getNrProperties()
479     #undef getPropertyNr(offset)
480     #undef getNamedProperty(attr.name)
481     #undef getPropertyName(property)
482     #undef getPropertyType(property)
483     #undef getPropertyValue(property)
484     #undef getAccountingEntity(property)
485     #undef getCurrencyID(property)
486     #undef getListID(property)
487     #undef getNodeID(property)
488     #undef getNounName(property)
489     #undef getUnitCode(property)
490     #undef getDescription(property)
491     #undef getStartDateTime(property)
492     #undef getEndDateTime(property)
493     #undef getUserAreaParent()
494
495     }
496
497     function extern long get.extension.point.version()
498     {
499         FunctionUsage
500             Get the extension point version from the extension.
501         EndFunctionUsage
502
503         return(1)
504     }
505
```