Quickfix - Issues with upgrading or installing Microsoft Dynamics GP 2010

Issues with upgrading or installing Microsoft Dynamics GP 2010

This page deals with resolving the common problem encountered by Great Plains users while upgrading to the latest version of Great Plains, Microsoft Dynamics Great Plains 2010.

The following SQL statement consistently produced an error:

```
CREATE VIEW GL10000CurrencyTranslationView AS select
\[GL10000Final\].\[OPENYEAR\], \[GL10000Final\].\[ACTINDX\],
\[GL10000Final\].\[CRDTAMNT\], \[GL10000Final\].\[DEBITAMT\],
\[GL10000Final\].\[ORCRDAMT\], \[GL10000Final\].\[ORDBTAMT\],
\[GL10000Final\].\[TRXDATE\], \[GL10000Final\].\[DSCRIPTN\], ...
```

If you too encounter the error, follow the scripts below to resolve the above issue.

Script 1

```
GO

SET ANSI_NULLS OFF
GO

SET QUOTED_IDENTIFIER ON
GO

SET ANSI_PADDING OFF
GO

CREATE TABLE \[dbo\].\[MC40600\](
\[CURNCYID\] \[char\](15) NOT NULL,
\[CurrentExchangeTableID\] \[char\](15) NOT NULL,
\[HistoricalExchgTableID\] \[char\](15) NOT NULL,
\[AverageExchangeTableID\] \[char\](15) NOT NULL,
\[BudgetExchangeTableID\] \[char\](15) NOT NULL,
\[DEX_ROW_ID\] \[int\] IDENTITY(1,1) NOT NULL,
CONSTRAINT \[PKMC40600\] PRIMARY KEY NONCLUSTERED
(
\[CURNCYID\] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON \[PRIMARY\]
) ON \[PRIMARY\]

GO

SET ANSI_PADDING OFF
GO
```

Script 2

```
GO

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF
GO

CREATE VIEW \[dbo\].\[GL10000CurrencyTranslationView\] AS select
\[GL10000Final\].\[OPENYEAR\], \[GL10000Final\].\[ACTINDX\],
\[GL10000Final\].\[CRDTAMNT\], \[GL10000Final\].\[DEBITAMT\],
\[GL10000Final\].\[ORCRDAMT\], \[GL10000Final\].\[ORDBTAMT\],
\[GL10000Final\].\[TRXDATE\], \[GL10000Final\].\[DSCRIPTN\],
\[GL10000Final\].\[REFRENCE\], \[GL10000Final\].\[CURNCYID\],
\[GL10000Final\].\[Original_Exchange_Rate\], \[GL10000Final\].\[JRNENTRY\],
\[GL10000Final\].\[TRXSORCE\],
\[GL10000Final\].\[SOURCDOC\], \[GL10000Final\].\[ORDOCNUM\],
```

```
\[GL10000Final\].\[ORTRXSRC\],
\[GL10000Final\].\[ORMSTRID\],\[GL10000Final\].\[ORMSTRNM\],
\[GL10000Final\].\[ORTRXTYP\], \[GL10000Final\].\[SERIES\],
\[GL10000Final\].\[VOIDED\], \[GL10000Final
\].\[Ledger_ID\], \[GL10000Final\].\[TranslationCurrencyID\],
\[GL10000Final\].\[CurrencyTranslationType\], \[GL10000Final
\].\[PERDENDT\], \[GL10000Final\].\[TranslationExchangeRate\], case
GL10000Final.CRDTAMNT when 0.0 then 0.0
else dbo.mcFuncCalculateAmountExtended(\[GL10000Final\].\[RTCLCMTD\], 3,
\[GL10000Final\].\[TranslationExchangeRate\],
\[GL10000Final\].\[DENXRATE\], \[GL10000Final\].\[MCTRXSTT\],
\[GL10000Final\].\[DECPLCUR\], \[GL10000Final\].\[CRDTAMNT\]) end as
TranslationCreditAmount, case GL10000Final.DEBITAMT when 0.0 then 0.0 else
dbo.mcFuncCalculateAmountExtended(\[GL10000Final
\].\[RTCLCMTD\], 3, \[GL10000Final\].\[TranslationExchangeRate\],
\[GL10000Final\].\[DENXRATE\], \[GL10000Final\].\[MCTRXSTT\],
\[GL10000Final\].\[DECPLCUR\], \[GL10000Final\].\[DEBITAMT\]) end as
TranslationDebitAmount, \[GL10000Final\].\[SequenceNumber\],
\[GL10000Final\].\[PERIODID\], \[GL10000Final\].\[CURRNIDX\],
\[GL10000Final\].\[DECPLCUR\], \[GL10000Final\].\[RATETPID\],
\[GL10000Final\].\[EXGTBLID\], \[GL10000Final\].\[EXCHDATE\],
\[GL10000Final\].\[TIME1\], \[GL10000Final\].\[RTCLCMTD\],
\[GL10000Final\].\[DENXRATE\], \[GL10000Final\].\[MCTRXSTT\],
\[GL10000Final\].\[Adjustment_Transaction\] from (select distinct
\[GL_TRX_WORK\].\[OPENYEAR\], \[GL_TRX_WORK\].\[ACTINDX\],
\[GL_TRX_WORK\].\[CRDTAMNT\], \[GL_TRX_WORK\].\[DEBITAMT\],\[GL_TRX_WORK
\].\[ORCRDAMT\], \[GL_TRX_WORK\].\[ORDBTAMT\], \[GL_TRX_WORK\].\[TRXDATE\],
\[GL_TRX_WORK\].\[DSCRIPTN\], \[GL_TRX_WORK
\].\[REFRENCE\], \[GL_TRX_WORK\].\[CURNCYID\], \[GL_TRX_WORK\].\[XCHGRATE\]
as Original_Exchange_Rate, \[GL_TRX_WORK
\].\[JRNENTRY\], \[GL_TRX_WORK\].\[TRXSORCE\],
\[GL_TRX_WORK\].\[SOURCDOC\], \[GL_TRX_WORK\].\[ORDOCNUM\], \[GL_TRX_WORK
\].\[ORTRXSRC\], \[GL_TRX_WORK\].\[ORMSTRID\],
\[GL_TRX_WORK\].\[ORMSTRNM\], \[GL_TRX_WORK\].\[ORTRXTYP\], \[GL_TRX_WORK
\].\[SERIES\], \[GL_TRX_WORK\].\[VOIDED\], \[GL_TRX_WORK\].\[Ledger_ID\],
\[GL_TRX_WORK\].\[TranslationCurrencyID\], \[GL_TRX_WORK
\].\[CurrencyTranslationType\], \[GL_TRX_WORK\].\[PERDENDT\], F.XCHGRATE as
TranslationExchangeRate, \[GL_TRX_WORK\].\[SQNCLINE\]
as SequenceNumber, \[GL_TRX_WORK\].\[PERIODID\], E.\[CURRNIDX\],
(E.\[DECPLCUR\]-1) as DECPLCUR, \[GL_TRX_WORK\].\[RATETPID\],
\[GL_TRX_WORK\].\[EXGTBLID\], F.\[EXCHDATE\], F.\[TIME1\], D.\[RTCLCMTD\],
dbo.mcFuncGetDenExchRate(GL_TRX_WORK.TranslationCurrencyID,D.RTCLCMTD) as
DENXRATE, \[GL_TRX_WORK\].\[MCTRXSTT\], \[GL_TRX_WORK
\].\[Adjustment_Transaction\] from DYNAMICS..MC40200 E, DYNAMICS..MC40300 D
cross apply (select e.YEAR1 as
OPENYEAR,a.JRNENTRY,a.SOURCDOC,a.REFRENCE,f.DSCRIPTN,a.TRXDATE,
a.TRXSORCE,f.ACTINDX,a.SERIES,f.ORTRXTYP,f.ORMSTRID,f.ORMSTRNM,
f.ORDOCNUM,a.ORTRXSRC,a.SQNCLINE,a.CURNCYID,b.CURNCYID as
TranslationCurrencyID, a.CURRNIDX,a.RATETPID,b.ExchangeTableID as
EXGTBLID,a.XCHGRATE,
a.EXCHDATE,a.TIME1,a.RTCLCMTD,dbo.glFuncGetPeriodID(a.TRXDATE,a.OPENYEAR,2)
as
PERIODID,f.CRDTAMNT,f.DEBITAMT,f.ORCRDAMT,f.ORDBTAMT, e.PERDENDT,
```

```
dbo.mcFuncGetMCTrxState(b.CURNCYID) as
MCTRXSTT,b.CurrencyTranslationType, a.VOIDED,a.Ledger_ID,
a.Adjustment_Transaction, case b.CurrencyTranslationType when 1 then
e.PERDENDT when 3 then a.TRXDATE end as ExchangeRateDate from GL10000 a,
GL10001 f, (select c.ACTINDX,b.CURNCYID, ExchangeTableID=
case CurrencyTranslationType when 1 then b.AverageExchangeTableID when 3
then b.HistoricalExchgTableID end,
c.CurrencyTranslationType from MC00200 c,MC40600 b where c.CURNCYID='' and
c.CurrencyTranslationType<>2) b, (select distinct
b.PERIODID, a.YEAR1,a.FSTFSCDY,a.LSTFSCDY, b.PERIODDT,b.PERDENDT from
SY40101 a, SY40100 b where a.YEAR1=b.YEAR1 and b.SERIES=2) e
where a.JRNENTRY=f.JRNENTRY and f.ACTINDX=b.ACTINDX and
a.PERIODID=e.PERIODID and e.YEAR1=e.YEAR1 and a.TRXDATE >=e.FSTFSCDY and
a.TRXDATE <=e.LSTFSCDY) GL_TRX_WORK cross apply
dbo.mcFuncGetExchangeRateTable(GL_TRX_WORK.ExchangeRateDate,
GL_TRX_WORK.EXGTBLID,
D.TRXDTDEF,
D.DATELMTS,D.PRVDSLMT,D.Base_Exchange_Rate_On,GL_TRX_WORK.MCTRXSTT) F where
GL_TRX_WORK.EXGTBLID = D.EXGTBLID and
```

```
D.CURNCYID=E.CURNCYID) GL10000Final
GO
```