**IGOR Information:**

**Screen Number in Dynamics:**

PA.IGO.00

**Application Path:**

C:\Program Files\IronWare Technologies\IGOR\Integer Group Optimized Reports.exe

**Location of the IGOR installation files:**

C:\Program Files\IronWare Technologies\IGOR

**File with Hard Coded User password and login:**

Integer Group Optimized Reports Viewer.exe.config

**-Lines of code to change:**

<setting name="ReportUser" serializeAs="String">

<value>ClientConApp</value>

</setting>

<setting name="ReportPassword" serializeAs="String">

<value>Qu!dPr0Qu0</value>

</setting>

**Files with Hard Coded Database Information:**

Integer Group Optimized Reports Viewer.exe.config

**-Lines of code to change:**

<setting name="ReportDB" serializeAs="String">

<value>IntegerDevAPP</value>

</setting>

Integer Group Optimized Reports.exe.config

**-Lines of code to change:**

<connectionStrings>

<add name="SLDBEntities" connectionString="metadata=res://\*/SLDB.csdl|res://\*/SLDB.ssdl|res://\*/SLDB.msl;provider=System.Data.SqlClient;provider connection string=&quot;Data Source=SQLDEV2;Initial Catalog=IntegerDevAPP;Integrated Security=True;MultipleActiveResultSets=True&quot;"

providerName="System.Data.EntityClient" />

</connectionStrings>

**-And**

<setting name="DBConnectionString" serializeAs="String"> <value>metadata=res://\*/SLDB.csdl|res://\*/SLDB.ssdl|res://\*/SLDB.msl;provider=System.Data.SqlClient;provider connection string="Data Source=SQLDEV2;Initial Catalog=IntegerDevAPP;Integrated Security=True;MultipleActiveResultSets=True"</value>

</setting>

**Stored Procedures called by IGOR:**

**xJAProcessReport\_IWT**

USE [DENVERAPP]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[xJAProcessReport\_IWT] Script Date: 03/06/2012 09:31:27 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[xJAProcessReport\_IWT]

@GUID varchar(255)

,@AgingDate smalldatetime

,@ActiveChildrenOnly smallint

,@SupressDfltDtl smallint

-- , @TemplateID smallint --added 03/17/2011 MSB

AS

BEGIN

SET NOCOUNT ON

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* SETUP TABLES \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

EXEC xJADeleteWrkTables\_IWT @GUID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* HEADER RECORD PROCESSING \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Update the #ListOfProjects with all the project "Bill With" projects

CREATE TABLE #ListOfProjectsWrk (Project char(16))

-- Determine all the Bill Withs for the selected projects

INSERT INTO #ListOfProjectsWrk (

Project)

SELECT

B.project

FROM

#ListOfProjects L INNER JOIN PJBILL A ON L.Project = A.Project

INNER JOIN PJBILL B ON A.project\_billwith = B.project\_billwith

WHERE

A.project\_billwith <> '' -- Note: This addresses an issue with bad billwith records in PJBill 12/3/2010 LGB

-- Clear out the list of project tables

DELETE FROM #ListOfProjects

-- Insert the complete list

INSERT INTO #ListOfProjects (

Project)

SELECT DISTINCT

Project

FROM

#ListOfProjectsWrk

-- Insert the Header records

INSERT INTO xJAReportHeaderWrk\_IWT (

SessionGUID

,Project)

SELECT

@GUID

,Project

FROM

#ListOfProjects

-- Update the Header Information

UPDATE

xJAReportHeaderWrk\_IWT

SET

Project\_desc = P.Project\_desc

,AgingDate = @AgingDate

,ASName = REPLACE(E2.Emp\_name, '~',', ')

,ClientName = ISNULL(XC.CName, 'NO CLIENT CONTACT')

,ClientEMail = ISNULL(XC.EmailAddress, 'NO EMAIL')

,CpnyID = P.CpnyID

,CustID = C.CustID

,CustomerName = C.Name

,End\_Date = CASE WHEN P.End\_Date = '1/1/1900' THEN NULL ELSE P.End\_Date END

,HasChildren = 0

,Inv\_format\_cd = B.Inv\_format\_cd

,Pm\_id01 = P.Pm\_id01

,Pm\_id02 = P.Pm\_id02

,Pm\_id08 = CASE WHEN P.Pm\_id08 = '1/1/1900' THEN NULL ELSE P.Pm\_id08 END

,Pm\_id28 = CASE WHEN PE.Pm\_id28 = '1/1/1900' THEN NULL ELSE PE.Pm\_id28 END

,Pm\_id32 = P.Pm\_id32

,PMName = REPLACE(E1.Emp\_name, '~',', ')

,Project\_BillWith = B.Project\_BillWith

,ProdCode = X.descr

,Purchase\_order\_num = P.Purchase\_order\_num

,Start\_Date = CASE WHEN P.Start\_Date = '1/1/1900' THEN NULL ELSE P.Start\_Date END

,Status\_pa = P.Status\_pa

-- , TemplateID = @TemplateID --added 03/17/2011 MSB

FROM

xJAReportHeaderWrk\_IWT H INNER JOIN PJProj P ON H.Project = P.Project

INNER JOIN PJProjEX PE ON H.Project = PE.Project

INNER JOIN Customer C ON P.pm\_id01 = C.CustID

INNER JOIN PJBILL B ON H.Project = B.Project

LEFT OUTER JOIN xIGProdCode X ON P.pm\_id02 = X.Code\_ID

LEFT OUTER JOIN xClientContact XC ON P.User2 = XC.EA\_ID

LEFT OUTER JOIN PJEmploy E1 ON P.manager1 = E1.employee

LEFT OUTER JOIN PJEmploy E2 ON P.manager2 = E2.employee

WHERE

SessionGUID = @GUID

-- Get a temp table with a list of the parent projects

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

SELECT

B.Project\_BillWith as Project

INTO

#ListOfParentsWithChildren

FROM

PJBILL B INNER JOIN xJAReportHeaderWrk\_IWT P ON B.Project\_BillWith = P.Project

AND P.SessionGUID = @GUID

AND P.Project = P.Project\_BillWith

GROUP BY

B.Project\_BillWith

HAVING

COUNT(B.Project\_BillWith) > 1

UPDATE

xJAReportHeaderWrk\_IWT

SET

HasChildren = 1

FROM

xJAReportHeaderWrk\_IWT INNER JOIN #ListOfParentsWithChildren L ON xJAReportHeaderWrk\_IWT.Project = L.Project

WHERE

xJAReportHeaderWrk\_IWT.SessionGUID = @GUID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* DETAIL RECORD PROCESSING \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO xJAReportDetailWrk\_IWT (

SessionGUID

,Acct\_group\_cd

,Amount

,BilledAmount

,Employee

,Emp\_name

,Ih\_id12

,Inv\_format\_cd

,Invoice\_num

,Invoice\_type

,MarkupAmount

,Project

,Pjt\_entity

,Pjt\_entity\_desc

,Trans\_date

,TranType

,Tr\_comment

,Tr\_id02

,Tr\_id03

,Tr\_id04

,Tr\_id08

,Units

,Vendor\_num

,Vendor\_name

, POLineItem) --added 1/28/2011 MSB

SELECT

@GUID

,A.Acct\_group\_cd -- Account Group

,CASE WHEN ISNULL(I.Invoice\_type, '') <> '' AND ISNULL(I.Ih\_id12, '') <> '' THEN -T.amount ELSE T.amount END

,ISNULL(I.amount, 0) -- Billed Amount

,T.Employee -- Employee ID

,REPLACE(E.Emp\_name, '~',', ') -- Employee Name

,I.Ih\_id12 -- Original Invoice Number

,I.Inv\_format\_cd -- Invoice Format Code

,I.Invoice\_num -- Client Invoice Number

,I.Invoice\_type -- Invoice Type

,0 -- Markup Amount

,L.Project

,P.Pjt\_entity

,P.Pjt\_entity\_desc

,T.Trans\_date

,'' -- TranType

,T.Tr\_comment -- Description

,T.Tr\_id02 -- InvoiceNbr

,T.Tr\_id04 -- Batch Number

,T.Tr\_id03 -- PONbr

,T.Tr\_id08 -- Invoice Date

,CASE A.Acct\_group\_cd WHEN 'LB' THEN T.Units ELSE 0 END -- Units

,T.Vendor\_num -- Vendor ID

,REPLACE(V.Name, '~',', ') -- Vendor Name

, ISNULL(P.user1, '') AS 'POLineItem' -- POLineItem --added 01/28/2011 MSB updated 02/18/2011 MSB

FROM

#ListOfProjects L INNER JOIN PJTran T ON L.Project = T.Project

INNER JOIN PJPent P ON T.Pjt\_entity = P.Pjt\_entity AND

T.Project = P.Project

INNER JOIN PJAcct A ON T.Acct = A.Acct

LEFT OUTER JOIN PJEmploy E ON T.Employee = E.Employee

LEFT OUTER JOIN Vendor V ON T.Vendor\_num = V.VendID

LEFT OUTER JOIN (

-- Associated Invoice Information as Derived Table

SELECT

P.Project

,T.user2 AS 'tr\_id12'

,MAX(H.invoice\_num) as 'invoice\_num'

,MIN(H.ih\_id12) as 'ih\_id12'

,MIN(H.invoice\_date) as 'invoice\_date'

,SUM(D.amount) as 'amount'

,MAX(H.inv\_format\_cd) as 'inv\_format\_cd'

,MAX(H.invoice\_type) as 'invoice\_type'

FROM

#ListOfProjects P INNER JOIN PJTran T ON P.Project = T.project

INNER JOIN PJInvDet D ON T.tr\_id23 = D.in\_id12

INNER JOIN PJInvHdr H ON D.draft\_num = H.draft\_num

WHERE

T.user2 <> ''

AND D.bill\_status = 'B'

GROUP BY

P.Project

,T.user2 ) AS I ON I.tr\_id12 = T.user2

WHERE

T.Trans\_date <= @AgingDate

AND A.acct\_group\_cd IN ('FE', 'WA', 'LB', 'PB')

-- 'WP' and related 'CM' amounts

INSERT INTO xJAReportDetailWrk\_IWT (

SessionGUID

,Acct\_group\_cd

,Amount

,BilledAmount

,Employee

,Emp\_name

,Ih\_id12

,Inv\_format\_cd

,Invoice\_num

,Invoice\_type

,MarkupAmount

,Project

,Pjt\_entity

,Pjt\_entity\_desc

,Trans\_date

,TranType

,Tr\_comment

,Tr\_id02

,Tr\_id03

,Tr\_id04

,Tr\_id08

,Units

,Vendor\_num

,Vendor\_name

, POLineItem) --added 01/28/2011 MSB

SELECT

@GUID

,'WP' -- Account Group

,SUM(CASE WHEN A.Acct\_group\_cd = 'CM' THEN 0 ELSE CASE WHEN ISNULL(I.Invoice\_type, '') <> '' AND ISNULL(I.Ih\_id12, '') <> '' THEN 0 ELSE T.amount END END)

,SUM(CASE WHEN A.Acct\_group\_cd = 'CM' THEN 0 ELSE CASE WHEN ISNULL(I.invoice\_type, '') = 'REVD' THEN 0 ELSE ISNULL(I.amount, 0) END END) -- Billed Amount

,T.Employee -- Employee ID

,REPLACE(E.Emp\_name, '~',', ') -- Employee Name

,I.Ih\_id12 -- Original Invoice Number

,I.Inv\_format\_cd -- Invoice Format Code

,I.Invoice\_num -- Client Invoice Number

,I.Invoice\_type -- Invoice Type

,SUM(CASE WHEN A.acct\_group\_cd = 'CM' THEN T.amount ELSE 0 END) -- Markup Amount

,L.Project

,P.Pjt\_entity

,P.Pjt\_entity\_desc

,T.Trans\_date

,'' -- TranType

,T.Tr\_comment -- Description

,T.Tr\_id02 -- InvoiceNbr

,T.Tr\_id03 -- PONbr

,T.Tr\_id04 -- Batch Number

,T.Tr\_id08 -- Invoice Date

,SUM(T.Units) -- Units

,T.Vendor\_num -- Vendor ID

,REPLACE(V.Name, '~',', ') -- Vendor Name

, ISNULL(P.user1, '') AS 'POLineItem' -- POLineItem --added 01/28/2011 MSB updated 02/18/2011 MSB

FROM

#ListOfProjects L INNER JOIN PJTran T ON L.Project = T.Project

INNER JOIN PJPent P ON T.Pjt\_entity = P.Pjt\_entity AND

T.Project = P.Project

INNER JOIN PJAcct A ON T.Acct = A.Acct

LEFT OUTER JOIN PJEmploy E ON T.Employee = E.Employee

LEFT OUTER JOIN Vendor V ON T.Vendor\_num = V.VendID

LEFT OUTER JOIN (

-- Associated Invoice Information as Derived Table

SELECT

P.Project

,T.user2 AS 'tr\_id12'

,MAX(H.invoice\_num) as 'invoice\_num'

,MIN(H.ih\_id12) as 'ih\_id12'

,MIN(H.invoice\_date) as 'invoice\_date'

,SUM(D.amount) as 'amount'

,MAX(H.inv\_format\_cd) as 'inv\_format\_cd'

,MAX(H.invoice\_type) as 'invoice\_type'

FROM

#ListOfProjects P INNER JOIN PJTran T ON P.Project = T.project

INNER JOIN PJInvDet D ON T.tr\_id23 = D.in\_id12

INNER JOIN PJInvHdr H ON D.draft\_num = H.draft\_num

WHERE

T.user2 <> ''

AND D.bill\_status = 'B'

GROUP BY

P.Project

,T.user2 ) AS I ON I.tr\_id12 = T.user2

WHERE

T.Trans\_date <= @AgingDate

AND A.acct\_group\_cd IN ('WP', 'CM')

GROUP BY

T.Employee -- Employee ID

,REPLACE(E.Emp\_name, '~',', ') -- Employee Name

,I.Ih\_id12 -- Original Invoice Number

,I.Inv\_format\_cd -- Invoice Format Code

,I.Invoice\_num -- Client Invoice Number

,I.Invoice\_type -- Invoice Type

,L.Project

,P.Pjt\_entity

,P.Pjt\_entity\_desc

,T.Trans\_date

,T.Tr\_comment -- Description

,T.Tr\_id02 -- InvoiceNbr

,T.Tr\_id03 -- PONbr

,T.Tr\_id04 -- Batch Number

,T.Tr\_id08 -- Invoice Date

,T.User2 -- User2

,T.Vendor\_num -- Vendor ID

,T.Voucher\_Num -- Voucher Number

,T.Voucher\_Line -- Voucher Line

,REPLACE(V.Name, '~',', ') -- Vendor Name

, P.user1 -- POLineItem --added 01/28/2011 MSB

-- 'PO' amounts

INSERT INTO xJAReportDetailWrk\_IWT (

SessionGUID

,Acct\_group\_cd

,Amount

,BilledAmount

,Employee

,Emp\_name

,Ih\_id12

,Inv\_format\_cd

,Invoice\_num

,Invoice\_type

,MarkupAmount

,Project

,Pjt\_entity

,Pjt\_entity\_desc

,Trans\_date

,TranType

,Tr\_comment

,Tr\_id02

,Tr\_id03

,Tr\_id04

,Tr\_id08

,Units

,Vendor\_num

,Vendor\_name

, POLineItem) --added 01/28/2011 MSB

SELECT

@GUID

,A.acct\_group\_cd

,C.Amount

,0

,''

,''

,''

,''

,''

,''

,0

,C.Project

,C.Pjt\_entity

,P.Pjt\_entity\_desc

,C.PO\_Date

,'PO'

,C.Tr\_comment

,''

,C.Purchase\_order\_num

,''

,''

,0

,C.Vendor\_num

,REPLACE(V.Name, '~',', ')

, ISNULL(P.user1, '') AS 'POLineItem' -- POLineItem --added 01/28/2011 MSB updated 02/18/2011 MSB

FROM

#ListOfProjects L INNER JOIN PJComDet C ON L.Project = C.Project

INNER JOIN PJAcct A ON C.Acct = A.Acct

INNER JOIN PJPent P ON C.Pjt\_entity = P.Pjt\_entity AND

C.Project = P.Project

LEFT OUTER JOIN Vendor V ON C.Vendor\_num = V.VendID

WHERE

C.PO\_Date <= @AgingDate

-- Enter the tasks for these projects that didn't have activity

INSERT INTO xJAReportDetailWrk\_IWT (

SessionGUID

,Acct\_group\_cd

,Project

,Pjt\_entity

,Pjt\_entity\_desc

, POLineItem) --added 02/18/2011 MSB

SELECT

@GUID

,'WP'

,P.Project

,P.Pjt\_entity

,P.Pjt\_entity\_desc

, ISNULL(p.user1, '') as 'POLineItem' --added 02/18/2011 MSB

FROM

#ListOfProjects L INNER JOIN PJPent P ON L.Project = P.Project

LEFT OUTER JOIN xJAReportDetailWrk\_IWT X ON X.Project = P.Project

AND X.Pjt\_entity = P.Pjt\_entity

AND X.SessionGUID = @GUID

WHERE

X.Pjt\_entity IS NULL

-- Delete the accrual tasks - we do this here BEFORE totals are calculated... Jill does not want these

-- anywhere on the report

DELETE FROM

xJAReportDetailWrk\_IWT

WHERE

Pjt\_entity = '99999'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* BILLING HISTORY RECORD PROCESSING \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Insert Bill History - AR

INSERT INTO xJAReportBillHistWrk\_IWT (

SessionGUID

,Project

,Amount

,DocDate

,DocType

,InvoiceNbr

,Payment

,Source)

SELECT

@GUID

,A.ProjectID

,CASE WHEN A.DocType IN ('IN','DM','FI','SC','CS') THEN A.OrigDocAmt ELSE -A.OrigDocAmt END

,A.DocDate

,A.DocType

,A.RefNbr

,CASE WHEN A.DocType = 'CS' THEN A.OrigDocAmt ELSE 0 END

,'ARDoc'

FROM

#ListOfProjects L INNER JOIN ARDoc A ON L.Project = A.ProjectID

WHERE

A.DocDate <= @AgingDate

AND A.DocType IN ('IN', 'DM', 'CM', 'FI', 'SB', 'SC', 'CS')

-- Insert Bill History - PJ

INSERT INTO xJAReportBillHistWrk\_IWT (

SessionGUID

,Project

,Amount

,CheckNumber

,DocDate

,DocType

,InvoiceNbr

,Payment

,Source)

SELECT

@GUID

,L.Project

,0

,D.AdjgRefNbr

,D.AdjgDocDate

,D.AdjgDocType

,P.Invoice\_refnbr

,D.CuryAdjgAmt

,'PJARPay'

FROM

#ListOfProjects L INNER JOIN ARDoc A ON L.Project = A.ProjectID

INNER JOIN PJARPay P ON P.custid = A.custid

AND P.invoice\_type = A.doctype

AND P.invoice\_refnbr = A.refnbr

INNER JOIN ARADJUST D ON P.custid = D.custid

AND P.invoice\_type = D.AdjdDocType

AND P.invoice\_refnbr = D.AdjdRefNbr

WHERE

D.AdjgDocDate <= @AgingDate

AND P.DocType = 'PA'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* BILLING SUMMARY RECORD PROCESSING \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Payment info into a Temp Table

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

SELECT

L.Project

,SUM(D.CuryAdjgAmt) AS Payments

INTO

#PJPaymentInfo

FROM

#ListOfProjects L INNER JOIN ARDoc A ON L.Project = A.ProjectID

INNER JOIN PJARPay P ON P.custid = A.custid

AND P.invoice\_type = A.doctype

AND P.invoice\_refnbr = A.refnbr

INNER JOIN ARADJUST D ON P.custid = D.custid

AND P.invoice\_type = D.AdjdDocType

AND P.invoice\_refnbr = D.AdjdRefNbr

WHERE

D.AdjgDocDate <= @AgingDate

AND P.DocType = 'PA'

GROUP BY

L.Project

-- Estimate info into a Temp Table (by project)

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

SELECT

P.Project

,SUM(P.eac\_amount) AS Estimate

INTO

#PJEstimates

FROM

#ListOfProjects L INNER JOIN PJPTDROL P ON L.Project = P.Project

WHERE

P.Acct IN ('ESTIMATE', 'ESTIMATE TAX')

GROUP BY

P.Project

-- Task Level Estimate info into a Temp Table

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

SELECT

P.Project

,P.Pjt\_entity

,SUM(P.eac\_amount) AS Estimate

INTO

#PJTaskEstimates

FROM

#ListOfProjects L INNER JOIN PJPTDSUM P ON L.Project = P.Project

WHERE

P.Acct IN ('ESTIMATE', 'ESTIMATE TAX')

GROUP BY

P.Project

,P.Pjt\_entity

-- Get the billing information summed up from the detail table

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

SELECT

A.Project

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd IN ('WP', 'CM') THEN (A.Amount + A.MarkupAmount) ELSE 0 END ELSE 0 END) AS VendorCost

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd IN ('WP', 'CM') THEN A.BilledAmount ELSE 0 END ELSE 0 END) AS VendorBill

,SUM(CASE WHEN A.Trantype = 'PO' THEN A.Amount ELSE 0 END) AS POCost

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'FE' THEN A.Amount ELSE 0 END ELSE 0 END) AS FeesCost

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'FE' THEN A.BilledAmount ELSE 0 END ELSE 0 END) AS FeesBill

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'WA' THEN A.Amount ELSE 0 END ELSE 0 END) AS WIPAPSCost

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'WA' THEN A.BilledAmount ELSE 0 END ELSE 0 END) AS WIPAPSBill

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'LB' THEN A.Units ELSE 0 END ELSE 0 END) AS Hours

,SUM(CASE WHEN A.Trantype = '' THEN CASE WHEN A.Acct\_group\_cd = 'PB' THEN A.Amount ELSE 0 END ELSE 0 END) AS Prebill

INTO

#PJTotals

FROM

xJAReportDetailWrk\_IWT A

WHERE

A.SessionGUID = @GUID

GROUP BY

A.project

-- Current Unlocked Estimate Amount info into a Temp Table

-- Note: You must do this as a temp table because you cannot perform an aggregate function on an expression containing an aggregate or a subquery

-- added 1/27/2011 MSB

SELECT

E.Project

,SUM(ISNULL(E.ULEAmount, 0)) AS 'ULEAmount'

INTO

#PJULEAmount

FROM

#ListOfProjects L LEFT JOIN xvr\_Est\_ULE\_Project E ON L.Project = E.Project

GROUP BY

E.Project

-- Now Update the project headers with the aggregate data from above 3 temp tables.

UPDATE

xJAReportHeaderWrk\_IWT

SET

Tot\_Estimate = ISNULL(E.Estimate,0)

,Tot\_FeesCost = ISNULL(T.FeesCost,0)

,Tot\_FeesBill = ISNULL(T.FeesBill,0)

,Tot\_Hours = ISNULL(T.Hours,0)

,Tot\_Payments = ISNULL(P.Payments,0)

,Tot\_PreBill = ISNULL(T.PreBill,0)

,Tot\_POCost = ISNULL(T.POCost,0)

,Tot\_VendorBill = ISNULL(T.VendorBill,0)

,Tot\_VendorCost = ISNULL(T.VendorCost,0)

,Tot\_WIPAPSBill = ISNULL(T.WIPAPSBill,0)

,Tot\_WIPAPSCost = ISNULL(T.WIPAPSCost,0)

FROM

xJAReportHeaderWrk\_IWT H LEFT OUTER JOIN #PJTotals T ON H.Project = T.Project

LEFT OUTER JOIN #PJPaymentInfo P ON H.Project = P.Project

LEFT OUTER JOIN #PJEstimates E ON H.Project = E.Project

WHERE

H.SessionGUID = @GUID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* INSERT PARENT SUMMARY INFORMATION \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Enter Sum of the children tasks and roll up to parent project

INSERT INTO xJAReportDetailWrk\_IWT (

SessionGUID

,Acct\_group\_cd

,Amount

,BilledAmount

,MarkupAmount

,Project

,Pjt\_entity

,Pjt\_entity\_desc

,TranType

,Units

,Vendor\_num

,Vendor\_name)

SELECT

@GUID

,D.Acct\_group\_cd

,SUM(ISNULL(D.Amount,0))

,SUM(ISNULL(D.BilledAmount,0))

,SUM(ISNULL(D.MarkupAmount,0))

,H.Project\_BillWith

,D.Pjt\_entity

,D.Pjt\_entity\_desc

,'\*\*' -- We have to put something in the TranType - the RPT suppresses lines that have a NULL TranType

,SUM(ISNULL(D.Units,0))

,'\*\*\* ' -- We are using the Vendor Fields to Show the Summary line to display nicely on the report

,CASE D.TranType WHEN 'PO' THEN 'Sum of POs ' + H.Project ELSE 'Sum of ' + H.Project END

FROM

xJAReportHeaderWrk\_IWT H INNER JOIN xJAReportDetailWrk\_IWT D ON H.SessionGUID = D.SessionGUID

AND H.Project = D.Project

WHERE

H.SessionGUID = @GUID

AND H.Project <> H.Project\_BillWith

GROUP BY

D.Acct\_group\_cd

,H.Project\_BillWith

,D.Pjt\_entity

,D.Pjt\_entity\_desc

,CASE D.TranType WHEN 'PO' THEN 'Sum of POs ' + H.Project ELSE 'Sum of ' + H.Project END

HAVING

SUM(ISNULL(D.Amount,0)) <> 0 -- We only want summary entries if there are values - the zero $ entries should be ignored

OR SUM(ISNULL(D.BilledAmount,0)) <> 0

OR SUM(ISNULL(D.MarkupAmount,0)) <> 0

OR SUM(ISNULL(D.Units,0)) <> 0

-- Now Update the project details with the aggregate data from above 3 temp tables.

-- Moved location from other 2 temp tables to here to fix update and inserting of summary information.

-- this is due to no longer updating the parent with TOTAL estimates of all children. Must come after the children summary of costs onto the parent.

UPDATE

xJAReportDetailWrk\_IWT

SET

Estimate = IsNull(E.Estimate,0)

FROM

xJAReportDetailWrk\_IWT D INNER JOIN #PJTaskEstimates E ON D.Project = E.Project

AND D.Pjt\_entity = E.Pjt\_entity

WHERE

D.SessionGUID = @GUID

-- We are going to summarize the Project Total information for all the kids (and parent) and roll it up to the parent project

-- Commented out Estimate to remove rollup of estimates for parent jobs - 04/18/2011 - apatten

SELECT

H.Project\_BillWith AS Project

-- ,SUM(H.Tot\_Estimate) AS Tot\_Estimate

,SUM(H.Tot\_FeesCost) AS Tot\_FeesCost

,SUM(H.Tot\_FeesBill) AS Tot\_FeesBill

,SUM(H.Tot\_Hours) AS Tot\_Hours

,SUM(H.Tot\_Payments) AS Tot\_Payments

,SUM(H.Tot\_PreBill) AS Tot\_PreBill

,SUM(H.Tot\_POCost) AS Tot\_POCost

,SUM(H.Tot\_VendorBill) AS Tot\_VendorBill

,SUM(H.Tot\_VendorCost) AS Tot\_VendorCost

,SUM(H.Tot\_WIPAPSBill) AS Tot\_WIPAPSBill

,SUM(H.Tot\_WIPAPSCost) AS Tot\_WIPAPSCost

INTO

#ProjectRollUpTotals

FROM

xJAReportHeaderWrk\_IWT H

WHERE

H.SessionGUID = @GUID

GROUP BY

H.Project\_BillWith

-- Now update the parent project info with this aggregate data

-- Commented out Estimate to remove rollup of estimates for parent jobs - 04/18/2011 - apatten

UPDATE

xJAReportHeaderWrk\_IWT

SET

-- Tot\_Estimate = ISNULL(T.Tot\_Estimate,0)

Tot\_FeesCost = ISNULL(T.Tot\_FeesCost,0)

,Tot\_FeesBill = ISNULL(T.Tot\_FeesBill,0)

,Tot\_Hours = ISNULL(T.Tot\_Hours,0)

,Tot\_Payments = ISNULL(T.Tot\_Payments,0)

,Tot\_PreBill = ISNULL(T.Tot\_PreBill,0)

,Tot\_POCost = ISNULL(T.Tot\_POCost,0)

,Tot\_VendorBill = ISNULL(T.Tot\_VendorBill,0)

,Tot\_VendorCost = ISNULL(T.Tot\_VendorCost,0)

,Tot\_WIPAPSBill = ISNULL(T.Tot\_WIPAPSBill,0)

,Tot\_WIPAPSCost = ISNULL(T.Tot\_WIPAPSCost,0)

FROM

xJAReportHeaderWrk\_IWT H INNER JOIN #ProjectRollUpTotals T ON H.Project = T.Project

WHERE

H.SessionGUID = @GUID

-- update the parent project info with Current Locked Estimate

-- added 1/27/2011 MSB

UPDATE

xJAReportHeaderWrk\_IWT

SET

CurrentLockedEstimate = ISNULL(E.ULEAmount ,0)

FROM

xJAReportHeaderWrk\_IWT H LEFT JOIN #PJULEAmount E ON H.Project = E.Project

WHERE

H.SessionGUID = @GUID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* DETAIL CLEANUP PROCESSING \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Now that we've calculated the billing totals, etc. we want to remove the PB records from the details that

-- display in the report

DELETE FROM

xJAReportDetailWrk\_IWT

WHERE

Acct\_group\_cd = 'PB'

-- We are going to remove the detail and headers for the Inactive children. We are doing this last because we still want the roll-up

-- to occur and we need the details which affect the totals.

IF @ActiveChildrenOnly <> 0 BEGIN

DELETE

xJAReportDetailWrk\_IWT

FROM

xJAReportDetailWrk\_IWT D INNER JOIN xJAReportHeaderWrk\_IWT H ON D.SessionGUID = H.SessionGUID

AND D.Project = H.Project

WHERE

H.SessionGUID = @GUID

AND H.Status\_pa = 'I'

AND H.Project <> H.Project\_BillWith

DELETE FROM

xJAReportHeaderWrk\_IWT

WHERE

SessionGUID = @GUID

AND Status\_pa = 'I'

AND Project <> Project\_BillWith

END

END

**xJADeleteTemplate\_IWT**

USE [DENVERAPP]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[xJADeleteTemplate\_IWT] Script Date: 03/06/2012 09:35:05 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[xJADeleteTemplate\_IWT]

@TemplateHdrKey int

AS

BEGIN

SET NOCOUNT ON

-- Delete the Detail Records

DELETE FROM

xJATemplateDtl\_IWT

WHERE

TemplateHdrKey = @TemplateHdrKey

-- Delete the Header Records

DELETE FROM

xJATemplateHdr\_IWT

WHERE

TemplateHdrKey = @TemplateHdrKey

END

**xJADeleteWrkTables\_IWT**

USE [DENVERAPP]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[xJADeleteWrkTables\_IWT] Script Date: 03/06/2012 09:35:29 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[xJADeleteWrkTables\_IWT]

@GUID varchar(255)

AS

BEGIN

SET NOCOUNT ON

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* SETUP TABLES \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

DELETE FROM xjareportheaderwrk\_iwt WHERE SessionGUID = @GUID

DELETE FROM xjareportdetailwrk\_iwt WHERE SessionGUID = @GUID

DELETE FROM xjareportbillhistwrk\_iwt WHERE SessionGUID = @GUID

END