Initial setup and maintenance for Employee ID To LAN ID

This document outlines the process that we might have to use to record a hire date for all

Employees that are in our Employee table and use that hire date and end dates to setup Employee ID to LAN ID links. The first part of the document describes the process to capture the Hire Date and fill any gaps from information from other sources. The second part describes the process for populating and maintaining the Employee ID TO LAN ID links.

SECTION 1 - Capturing Hire Date

1. Last Hire Date is being requested in the Employee file from PeopleSoft.
2. Initial population - During a regular daily employee table update the Hire date will be populated for all active employees in the file.

* Steps 1 and 2 have been implemented in current production table.

1. Filling the gaps step1and 2 – this will be implemented at the time of the implementation of the redesigned database.
2. Source1: From Scorecard Hierarchy table using the earliest start date that can be found for an employee id. This step will cleanup a major chunk of missing start dates.

## BELOW STEPS ARE TUN AGAINST THE SCORECARDS DATABASE

## UPDATE eCOACHING DB NAME IN QUERY WITH APPROPRIATE ENVIRONMENT

;With StartDt as

(SELECT

[CsrEmployeeId]

,[CsrLanId]

,[Start\_Date]

,ROW\_NUMBER() OVER( PARTITION BY [CsrEmployeeId]

ORDER BY [Start\_Date]) AS Rn

FROM [bcc\_scrd].[HIERARCHY\_V02])

--SELECT \* FROM StartDt

--WHERE Rn=1

UPDATE [eCoachingDev].[EC].[Employee\_Hierarchy]

SET Start\_Date= sd.START\_DATE

FROM [eCoachingDev].[EC].[Employee\_Hierarchy]ec join StartDt sd

ON ec.Emp\_ID = sd.CsrEmployeeId

AND ec.Start\_Date is NULL

AND sd.Rn=1

1. Filling the gaps step2- For the records still missing s start date, we can get a few from the Scorecard Verint user stage table using the earliest hire date that can be found for an employee id.

;With StartDt as

(SELECT

[Employee\_ID]

,CONVERT(nvarchar(10),[User\_Hire\_Date],112)as [User\_Hire\_Date]

,ROW\_NUMBER() OVER( PARTITION BY [Employee\_ID]

ORDER BY [User\_Hire\_Date]) AS Rn

FROM [bcc\_scrd].[Verint\_User\_Stage])

--SELECT \* FROM StartDt

--WHERE Rn=1

UPDATE[eCoachingDev].[EC].[Employee\_Hierarchy]

SET Start\_Date= sd.[User\_Hire\_Date]

FROM [eCoachingDev].[EC].[Employee\_Hierarchy]ec join StartDt sd

ON ec.Emp\_ID = sd.[Employee\_ID]

AND ec.Start\_Date is NULL

AND sd.Rn=1

1. Filling gaps step3 – For any remaining employees, assign a default start date of 20130101 to all remaining records that have no start date.

UPDATE [eCoachingDev].[EC].[Employee\_Hierarchy]

SET Start\_Date= '20130101'

WHERE Start\_Date is NULL and end\_date >'20130101'

# THESE STATEMENTS RUN IN A WINDOW POINTING TO eCOACHING DB

SECTION 2 – Initial Populate and maintain Employee ID To LAN ID Table

Run sql below to do the Initial Insert. Subsequent maintenance will be handled as part of the daily Employee update process using Stored procedure [EC].[sp\_Update\_EmployeeID\_To\_LanID]

--1. Initial Populate

DECLARE @dtNow DATETIME

SET @dtNow = GETDATE()

INSERT INTO [EC].[EmployeeID\_To\_LanID]

([EmpID]

,[StartDate]

,[EndDate]

,[LanID]

,[DatetimeInserted]

,[DatetimeLastUpdated])

(SELECT

Emp\_ID,

--CONVERT(nvarchar(10),@dtNow,112),

Start\_Date,

End\_Date,

Emp\_LanID,

@dtNow ,

@dtNow

FROM [EC].[Employee\_Hierarchy]EH LEFT OUTER JOIN [EC].[EmployeeID\_To\_LanID]LAN

ON EH.[Emp\_LanID]= LAN.[LanID]

AND EH.[Emp\_ID]= LAN.[EmpID]

WHERE LAN.[EmpID]IS NULL AND LAN.[LanID]IS NULL

AND EH.[Emp\_LanID] IS NOT NULL)

1. Handle overlaps of Lan ID usage

;With Overlaps (OverlaplanId, BaselanId)

As

(

Select Overlap.lanId, Base.lanId

From [EC].[EmployeeID\_To\_LanID] As Base

Inner Join [EC].[EmployeeID\_To\_LanID] As Overlap On Overlap.LanID = Base.lanid

Where (Overlap.StartDate > Base.StartDate)

And (Overlap.StartDate < Base.EndDate)

)

/\*

select lan.[EmpID]

,lan.[StartDate]

,lan.[EndDate]

,lan.[LanID]

from [eCoachingDev].[EC].[EmployeeID\_To\_LanID]lan join Overlaps o

on lan.LanID = o.BaselanId

\*/

--This below sql will be used to overcome the lan id overlaps by updating the overlap rows to assign startdate of next row -1 to previous row

Update Base

set EndDate = [EC].[fn\_intDatetime\_to\_YYYYMMDD](DATEADD(DAY,-1,CONVERT(datetime,convert(char(8),Overlap.StartDate))))

From [EC].[EmployeeID\_To\_LanID] As Base

Inner Join [EC].[EmployeeID\_To\_LanID] As Overlap On Overlap.LanID = Base.lanid

Where (Overlap.StartDate > Base.StartDate)

And (Overlap.StartDate < Base.EndDate)

Note:\*This overlap cleanup will be part of the procedure [EC].[sp\_Update\_EmployeeID\_To\_LanID] to handle overlap cleanup on an ongoing basis.

# MAKE SURE TO UPDATE THE SCORECARDS DB NAME TO THE APPRIOPRIATE ENVIRONMENT

--3. Add missing records from Scorecards database

DECLARE @dtNow DATETIME

SET @dtNow = GETDATE()

;With Firstrecord AS

(SELECT

[EMPLOYEE\_ID]

,[START\_DATE]

,[END\_DATE]

,[LAN\_ID]

,ROW\_NUMBER() OVER( PARTITION BY [EMPLOYEE\_ID]

ORDER BY [START\_DATE]) AS Rn

FROM

[BCC\_Scorecards\_Dev]. [bcc\_scrd].[EMPLOYEE\_ID\_TO\_LAN\_ID]

--Where [EMPLOYEE\_ID]='232767'

)

--Select \* FROM Firstrecord

--WHERE Rn = 1

INSERT INTO [EC].[EmployeeID\_To\_LanID]

([EmpID]

,[StartDate]

,[EndDate]

,[LanID]

,[DatetimeInserted]

,[DatetimeLastUpdated])

Select F.[EMPLOYEE\_ID],F.[START\_DATE], F.[END\_DATE], F.[LAN\_ID], @dtNow,@dtNow

FROM Firstrecord AS F LEFT OUTER JOIN [EC].[EmployeeID\_To\_LanID]LAN

ON F.EMPLOYEE\_ID = LAN.[EmpID]

WHERE LAN.[EmpID]IS NULL

AND F.Rn =1