Newspapers in Education (NIE)

NIE.mde is a Microsoft Access executable that Jennifer Ramsdell runs on a regular basis. It is updated every August with a list (?) of subscribed schools.

This document assumes that the user is familiar with Microsoft Access (in particular, the older, 2002-2007 versions).

The database itself can be found on [\\Vern\NIE](file:///\\Vern\NIE). It comprises

* Niedata.mdb – the database itself (i.e., the “back end”). Stored in folder \\Vern\Nie.
* Niedata.ldb – A lock file indicating all users currently accessing the database. Stored in folder \\Vern\Nie.
* Nie.mdb – The application front end.
* Nie.mde – The front end’s executable (made from Nie.mdb).

As of August 2019, master copies of the front end, along with archived copies of the back end, are stored on GitHub (NIE repository).

Yearly updates are a manual process. This process is outlined below. In general,

* Copy subscription data associated with subscriptions that have ended to the corresponding archive tables;
* Delete the stale subscription data from the tables.

The tables containing “active” subscription data include:

* tblCorrespondenceHistory
* tblLedger
* tblSponsors
* tblSponFin
* tblSubscriptions

*NOTE: The front end application is linked to the back end data via the front end’s Linked Table Manager. If the back end data is ever moved from* [*\\Vern\Nie*](file:///\\Vern\Nie)*, open the Linked Table Manager and re-link the front end tables to the new location for Niedata.mdb.*

**Niedata.mdb Modifications**

□ Make an archived copy of the database file \\Vern\NIE\Niedata.mdb.

□ Make a local copy of [\\Vern\NIE\Niedata.mdb](file:///\\Vern\NIE\Niedata.mdb).

□ Open the local copy of the MDB file (Niedata.mdb)

□ Edit the table zz-EndDate. Stick the desired end date into the first (and only!) record in the table! **Warning!** Be sure that there is only a single record in this table as it is used as a criteria inside of the WHERE clauses in the numbered queries, below.

□ Run the following queries (which update tables EOY\_Delivery\_Draw and EOY\_School\_Info):

* Step1\_EOY\_mtg\_DeliveryTotals
* Step2\_EOY\_mtg\_info

□ Get the list of school accounts that **should not** be archived. This is generated every August (typically by Jen Ramsdell) and comprises those accounts that have been extended through the next school year.

□ Edit the query Step3\_GetSubDListFromAccountNums, and change the criteria under the WHERE clause to conform to the “**should not** be archived” list. For example, this is the criteria clause for 2019:

SELECT EOY\_School\_info.AcctNum, tblSubscriptions.SubID, tblSubscriptions.StartDate, tblSubscriptions.EndDate,

tblSubscriptions.OrdDate, tblSubscriptions.ActFlag

FROM tblSubscriptions

INNER JOIN EOY\_School\_info ON tblSubscriptions.SubID = EOY\_School\_info.SubID

WHERE (((EOY\_School\_info.AcctNum)='mna079' Or (EOY\_School\_info.AcctNum)='mna103'

Or (EOY\_School\_info.AcctNum)='mna115' Or (EOY\_School\_info.AcctNum)='mna142'

Or (EOY\_School\_info.AcctNum)='mna157' Or (EOY\_School\_info.AcctNum)='mna625'

Or (EOY\_School\_info.AcctNum)='mna710' Or (EOY\_School\_info.AcctNum)='mna835'

Or (EOY\_School\_info.AcctNum)='mna844' Or (EOY\_School\_info.AcctNum)='mna916'

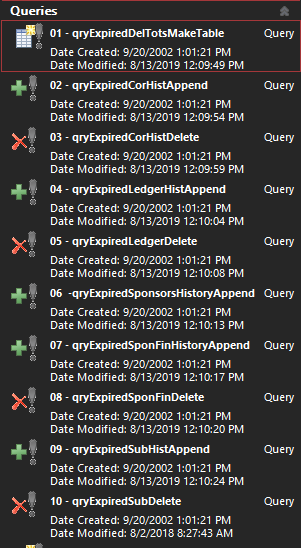
Or (EOY\_School\_info.AcctNum)='mnb033'))

ORDER BY EOY\_School\_info.AcctNum, tblSubscriptions.SubID, tblSubscriptions.EndDate;

□ Run the query Step3\_GetSubIDListFromAccountNums. For ease of use, cut and paste the results into a spreadsheet. These are the subscriptions whose end dates much change.

□ Run the query Step4\_ChangeTheseEndDates, and manually change each of the end dates to some date beyond the archive date (9/1 of the current year should work).

□ Scroll the query navigation pane on the left and note that there are 10 queries numbered 01-10.



□ Execute these queries in order. You will (unfortunately) be prompted to allow the tables to be updated; just click YES on all the prompts. All records prior to the archive date will be removed from the active data tables.

□ Close the database and move it to \\Vern\NIE

**Nie.mdb Modifications**

□ Edit Nie.mdb by holding down the shift key while double-clicking this file. If you get a warning that not all macros were enabled, continue holding down the shift key while clicking the “enable” button, then release the shift key. Holding the shift key **should** expose the underlying data tables, queries and reports that the Nie program executes.

□ Edit the rptAffidavit report (right-click, DesignView) and change the dates and whatever else may need to be modified for the new school year; then save it. Do the same with rptSponsorConfirmation.

□ Click Save As 🡪 Make MDE to create the MDE file. This executable gets copied to Jennifer Ramsdell’s local PC (in C:\NIE).

**More Notes:**

These are the queries that update the database by archiving stale subscription data.

**01 – qryExpiredDelTotsMakeTable**

SELECT tblDelivery.SubID, Count(tblDelivery.Date) AS CountOfDate, Sum(tblDelivery.Draw) AS TotDraw,

Sum(tblDelivery.Plus) AS TotAdds, Sum(tblDelivery.Minus) AS TotCuts,

IIf(IsNull([TotAdds]),0,[TotAdds])-IIf(IsNull([TotCuts]),0,[TotCuts]) AS AdjTot,

[TotDraw]+[AdjTot] AS AdjDrawTot, [AdjTot]\*0.22 AS AdjCost, [AdjDrawTot]\*0.22 AS SubCost

INTO [xx-DelTotsExpiredSubs]

FROM tblSubscriptions

INNER JOIN tblDelivery ON tblSubscriptions.SubID = tblDelivery.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True) AND ((tblDelivery.Deliver)=True))

GROUP BY tblDelivery.SubID

ORDER BY tblDelivery.SubID;

Creates the xx-DelTotsExpiredSubs table. Once this gets created I THINK it needs to be edited to remove the subscriptions that should be carried over into the subsequent school year **or else tblSubscriptions must be edited prior to use!**

**02 – qryExpriedCorHistAppend**

INSERT INTO tblCorHistHistory ( SubID, Form, Printime, [User], Days, Draw, Charges )

SELECT tblCorrespondenceHistory.SubID, tblCorrespondenceHistory.Form,

tblCorrespondenceHistory.Printime, tblCorrespondenceHistory.User,

tblCorrespondenceHistory.Days, tblCorrespondenceHistory.Draw, tblCorrespondenceHistory.Charges

FROM tblSubscriptions

INNER JOIN tblCorrespondenceHistory ON tblSubscriptions.SubID = tblCorrespondenceHistory.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Appends expired subscription data from tblSubscriptions to tblCorHistHistory

**03 – qryExpiredcorHistDelete**

DELETE tblCorrespondenceHistory.\*, IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False) AS OldData

FROM tblSubscriptions

INNER JOIN tblCorrespondenceHistory ON tblSubscriptions.SubID = tblCorrespondenceHistory.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Deletes old history data from tblCorrespondenceHistory any records older than the specified subscription end date.

**04 – qryExpiredLedgerHistAppend**

INSERT INTO tblLedgerHistory ( TransID, SubID, School, AcctNum, TransDate, DueDate,

Charge, Payment, Description, TransCode, Clerk )

SELECT tblLedger.TransID, tblLedger.SubID, tblLedger.School, tblLedger.AcctNum, tblLedger.TransDate, tblLedger.DueDate,

tblLedger.Charge, tblLedger.Payment, tblLedger.Description, tblLedger.TransCode, tblLedger.Clerk

FROM tblSubscriptions

INNER JOIN tblLedger ON tblSubscriptions.SubID = tblLedger.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Appends expired subscription data from tblSubscriptions to tblLedgerHistory

**05 – qryExpiredLedgerDelete**

DELETE tblLedger.\*, IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False) AS OldData

FROM tblSubscriptions

INNER JOIN tblLedger ON tblSubscriptions.SubID = tblLedger.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Deletes old data from tblLedger any records older than the specified subscription end date

**06 – qryExpiredSponsorsHistoryAppend**

INSERT INTO tblSponsorsHistory ( SponID, SponName, SponAdd, SponCity, SponState, SponZip, SponAttn )

SELECT tblSponsors.SponID, tblSponsors.SponName, tblSponsors.SponAdd, tblSponsors.SponCity,

tblSponsors.SponState, tblSponsors.SponZip, tblSponsors.SponAttn

FROM tblSponsors

LEFT JOIN tblSponsorsHistory ON tblSponsors.SponID = tblSponsorsHistory.SponID

WHERE (((tblSponsorsHistory.SponID) Is Null));

Appends data from tblSponsors to tblSponsorsHistory any records whose sponsor ID (sponID) is null.

**07 – qryExpiredsponFinHistoryAppend**

INSERT INTO tblSponFinHistory ( SponID, SubID, [Date], [Desc], Donation, Allocation )

SELECT tblSponFin.SponID, tblSponFin.SubID, tblSponFin.Date, tblSponFin.Desc, tblSponFin.Donation, tblSponFin.Allocation

FROM tblSubscriptions

LEFT JOIN tblSponFin ON tblSubscriptions.SubID = tblSponFin.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Appends expired subscription data from tblSubscriptions to tblSponFinHistory.

**08 – qryExpiredSponFinDelete**

DELETE tblSponFin.\*, IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False) AS OldData

FROM tblSubscriptions

LEFT JOIN tblSponFin ON tblSubscriptions.SubID = tblSponFin.SubID

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Deleted old data from tblSponFin any records older than the specified subscription end date.

**09 – qryExpiredSubHistAppend**

INSERT INTO tblSubscriptionHistory ( SubID, SchID, RouteID, MotherTruck, RelayID, DealerName, PO,

BillTo, BillAdd, BillCity, BillState, BillZip, Attn, Teacher, Grade, Subject, OrdDate, PromoID, Rate,

Mon, Tue, Wed, Thu, Fri, Sat, Sun, DeliverLoc, Comments, StartDate, EndDate, PayType, HistStatus,

SponName, SponAdd, SponCity, SponState, SponZip, SponAttn, Sponsorship, Vstop1, VStart1, Vstop2,

VStart2, Clerk, Email, DelDays, DelPapers, DelCharge, AdjPapers, AdjCharge, SchoolName, Address, CSZ,

AcctNum )

SELECT tblSubscriptions.SubID, tblSubscriptions.SchID, tblSubscriptions.RouteID, tblSubscriptions.MotherTruck,

tblSubscriptions.RelayID, tblSubscriptions.DealerName, tblSubscriptions.PO, tblSubscriptions.BillTo,

tblSubscriptions.BillAdd, tblSubscriptions.BillCity, tblSubscriptions.BillState, tblSubscriptions.BillZip,

tblSubscriptions.Attn, tblSubscriptions.Teacher, tblSubscriptions.Grade, tblSubscriptions.Subject,

tblSubscriptions.OrdDate, tblSubscriptions.PromoID, tblSubscriptions.Rate, tblSubscriptions.Mon,

tblSubscriptions.Tue, tblSubscriptions.Wed, tblSubscriptions.Thu, tblSubscriptions.Fri, tblSubscriptions.Sat,

tblSubscriptions.Sun, tblSubscriptions.DeliverLoc, tblSubscriptions.Comments, tblSubscriptions.StartDate,

tblSubscriptions.EndDate, tblSubscriptions.PayType, tblSubscriptions.HistStatus, tblSubscriptions.SponName,

tblSubscriptions.SponAdd, tblSubscriptions.SponCity, tblSubscriptions.SponState, tblSubscriptions.SponZip,

tblSubscriptions.SponAttn, tblSubscriptions.Sponsorship, tblSubscriptions.Vstop1, tblSubscriptions.VStart1,

tblSubscriptions.Vstop2, tblSubscriptions.VStart2, tblSubscriptions.Clerk, tblSubscriptions.Email,

[xx-DelTotsExpiredSubs].CountOfDate AS Expr1, [xx-DelTotsExpiredSubs].AdjDrawTot AS Expr2,

[xx-DelTotsExpiredSubs].SubCost AS Expr3, [xx-DelTotsExpiredSubs].AdjTot AS Expr4,

[xx-DelTotsExpiredSubs].AdjCost AS Expr5, tblSchools.School, tblSchools.MAdd,

tblSchools.MCity & " " & [MState] & " " & [MZip] AS CSZ, tblSchools.AcctNum

FROM [xx-DelTotsExpiredSubs]

INNER JOIN (tblSubscriptions

INNER JOIN tblSchools ON tblSubscriptions.SchID = tblSchools.SchID) ON [xx-DelTotsExpiredSubs].SubID = tblSubscriptions.SubID

WHERE (((IIf([tblSubscriptions].[EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Appends expired subscription data from tblSubscriptions to tblSubscriptionHistory.

**10 – qryExpiredSubDelete**

DELETE tblSubscriptions.SubID, tblSubscriptions.SchID, tblSubscriptions.DlrID AS Expr1, tblSubscriptions.TrkID AS Expr2,

tblSubscriptions.DropID AS Expr3, tblSubscriptions.PO, tblSubscriptions.BillTo, tblSubscriptions.BillAdd,

tblSubscriptions.BillCity, tblSubscriptions.BillState, tblSubscriptions.BillZip, tblSubscriptions.Attn,

tblSubscriptions.Teacher, tblSubscriptions.Grade, tblSubscriptions.Subject, tblSubscriptions.OrdDate,

tblSubscriptions.PromoID, tblSubscriptions.Rate, tblSubscriptions.Mon, tblSubscriptions.Tue,

tblSubscriptions.Wed, tblSubscriptions.Thu, tblSubscriptions.Fri, tblSubscriptions.Sat, tblSubscriptions.Sun,

tblSubscriptions.DeliverLoc, tblSubscriptions.Comments, tblSubscriptions.DlrName AS Expr4,

tblSubscriptions.StartDate, IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False) AS OldData,

tblSubscriptions.ActFlag, tblSubscriptions.PayType, tblSubscriptions.HistStatus, tblSubscriptions.SponID,

tblSubscriptions.SponAdd, tblSubscriptions.SponCity, tblSubscriptions.SponState, tblSubscriptions.SponZip,

tblSubscriptions.SponAttn, tblSubscriptions.Sponsorship, tblSubscriptions.SponID2, tblSubscriptions.SponAdd2,

tblSubscriptions.SponCity2, tblSubscriptions.SponState2, tblSubscriptions.SponZip2, tblSubscriptions.SponAttn2,

tblSubscriptions.Sponsorship2, tblSubscriptions.Vstop1, tblSubscriptions.VStart1, tblSubscriptions.Vstop2,

tblSubscriptions.VStart2, tblSubscriptions.Clerk, tblSubscriptions.Email

FROM tblSubscriptions

WHERE (((IIf([EndDate]<(SELECT EndDate FROM [zz-EndDate]),True,False))=True));

Deletes expired subscription data from tblSubscriptions.

**Step3\_GetSubIDListFromAccountNums – Use this to obtain the SubID field from the “Don’t Archive” list**

SELECT EOY\_School\_info.AcctNum, tblSubscriptions.SubID, tblSubscriptions.StartDate,

tblSubscriptions.EndDate, tblSubscriptions.OrdDate, tblSubscriptions.ActFlag

FROM tblSubscriptions

INNER JOIN EOY\_School\_info ON tblSubscriptions.SubID = EOY\_School\_info.SubID

WHERE (((EOY\_School\_info.AcctNum)='mna079' Or (EOY\_School\_info.AcctNum)='mna103'

Or (EOY\_School\_info.AcctNum)='mna115' Or (EOY\_School\_info.AcctNum)='mna142'

Or (EOY\_School\_info.AcctNum)='mna157' Or (EOY\_School\_info.AcctNum)='mna625'

Or (EOY\_School\_info.AcctNum)='mna710' Or (EOY\_School\_info.AcctNum)='mna835'

Or (EOY\_School\_info.AcctNum)='mna844' Or (EOY\_School\_info.AcctNum)='mna916'

Or (EOY\_School\_info.AcctNum)='mnb033'))

ORDER BY EOY\_School\_info.AcctNum, tblSubscriptions.SubID, tblSubscriptions.EndDate;