BDO EYES: Word-Check phonetic cross control

To improve the hit search algorithm, in the BDO EYES solution, an additional control method based on phonetic search, has been developed based on SoundEx & Metaphone.

Metaphone is a phonetic algorithm, for indexing words by their English pronunciation. A new version of this algorithm, named Double metaphone take into account spelling peculiarities of a number of other languages.

The combination of these 2 algorithms (SoundEx & Double metaphone) is used to improve the word-check cross control in BDO EYES.

The double metaphone algorithm has been adapted to be able to configure the length of the returned string and thus improve the precision of the results.

The control principle is as follows:

For a natural person, compare the name of the person transformed according to the SoundEx, DoubleMetaphon and DoubleMetaphonCustom algorithms with the phonetics of the World-Check database records. Same with the first name. For a company, control is only done on Name of the company.

An example of structure and control applied by BDO EYES software is described below. This model cannot be used as is and will have to be adapted by the provider.

The process can be divided into 3 parts:

1. Creation of a structure for storing World-Check data in phonetic form
2. Loading and (phonetic) conversion of this data during the daily update
3. Search for the name (first name) of the entity in this phonetic database

# Storage of the phonetic data

Create ExtSrceFullSound table, with SoundEx & Metaphone fields:

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ExtSrceFullSound](

[UID] [int] NOT NULL, --World-Check UID

[LastNameSoundEx] [varchar](4) NULL,

[FirstNameSoundEx] [varchar](4) NULL,

[LastNameMetaphone1] [varchar](4) NULL,

[FirstNameMetaphone1] [varchar](4) NULL,

[LastNameMetaphone2] [varchar](4) NULL,

[FirstNameMetaphone2] [varchar](4) NULL,

[LastNameMetaphoneCustom] [varchar](10) NULL,

[FirstNameMetaphoneCustom] [varchar](10) NULL,

CONSTRAINT [PK\_ExtSrceFullSound] PRIMARY KEY CLUSTERED

(

[UID] 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

# Phonetic data loading

During Word-Check file loading (daily update), populate fields SoundEx, Metaphone 1 & 2 and Metaphone custom on last name & first name for individuals and Last name for companies.

Parameters:

@Length : Length of the double metaphone custom. Value 10 used by BDO EYES

@CleanHistoric : 0 = No, 1 = True, based on your loading policy

@DBname : Name of the External source database. BDO EYES is built for managing several types of external sources

The fnDoubleMetaphoneScalar function returns the phonetic value, following the Double metaphone algorithm on 4 digits (See chapter Double metaphone algorithm).

The fnDoubleMetaphoneScalarCustom function returns the phonetic value, following the Double metaphone algorithm on the length given as parameter (See chapter Double metaphone algorithm).

The algorithm given in the appendix has been adapted to extract a string of 4 characters, for the standard function and a string of variable length for the 'Custom' function.

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[sp\_ExtSrce\_WorldCheck\_Sounds]

@Length INT, @CleanHistoric INT, @DBName SYSNAME

AS

BEGIN

DECLARE @sql AS VARCHAR (MAX);

IF @CleanHistoric = 1

BEGIN

SET @sql = 'DELETE FROM ' + @DBName + '..ExtSrceFullSound';

EXECUTE (@sql);

END

SET @sql = 'INSERT INTO ' + @DBName + '..ExtSrceFullSound (UID, LastNameSoundEx, FirstNameSoundEx, LastNameMetaphone1, LastNameMetaphone2, FirstNameMetaphone1, FirstNameMetaphone2, LastNameMetaphoneCustom, FirstNameMetaphoneCustom)';

SET @sql = @sql + ' SELECT a.UID, SOUNDEX(a.LastName), SOUNDEX(a.FirstName) , dbo.fnDoubleMetaphoneScalar(1,a.LastName), dbo.fnDoubleMetaphoneScalar(2,a.LastName)';

SET @sql = @sql + ' , dbo.fnDoubleMetaphoneScalarCustom(1,a.FirstName, 4), dbo.fnDoubleMetaphoneScalarCustom(2,a.FirstName, 4)';

SET @sql = @sql + ' , dbo.fnDoubleMetaphoneScalarCustom(1,a.LastName, ' + CONVERT (VARCHAR (2), @Length) + '), dbo.fnDoubleMetaphoneScalarCustom(1,a.FirstName, ' + CONVERT (VARCHAR (2), @Length) + ')';

IF @CleanHistoric = 1

BEGIN

SET @sql = @sql + ' FROM ' + @DBName + '..ExtSrceFull a';

SET @sql = @sql + ' WHERE a.LastName IS NOT NULL AND a.FirstName IS NOT NULL';

END

ELSE

BEGIN

SET @sql = @sql + ' FROM ' + @DBName + '..ExtSrceFullUpdate a';

SET @sql = @sql + ' LEFT OUTER JOIN ' + @DBName + '..ExtSrceFullSound b ON a.UID = b.UID';

SET @sql = @sql + ' WHERE a.LastName IS NOT NULL AND a.FirstName IS NOT NULL AND b.UID IS NULL';

END

EXECUTE (@sql);

--Init PM

SET @sql = 'INSERT INTO ' + @DBName + '..ExtSrceFullSound (UID, LastNameSoundEx, FirstNameSoundEx, LastNameMetaphone1, LastNameMetaphone2, FirstNameMetaphone1, FirstNameMetaphone2, LastNameMetaphoneCustom, FirstNameMetaphoneCustom)';

SET @sql = @sql + ' SELECT a.UID, SOUNDEX(a.LastName), NULL , dbo.fnDoubleMetaphoneScalar(1,a.LastName), dbo.fnDoubleMetaphoneScalar(2,a.LastName)';

SET @sql = @sql + ' , NULL, NULL';

SET @sql = @sql + ' , dbo.fnDoubleMetaphoneScalarCustom(1,a.LastName, ' + CONVERT (VARCHAR (2), @Length) + '), NULL';

IF @CleanHistoric = 1

BEGIN

SET @sql = @sql + ' FROM ' + @DBName + '..ExtSrceFull a';

SET @sql = @sql + ' WHERE a.LastName IS NOT NULL AND a.FirstName IS NULL';

END

ELSE

BEGIN

SET @sql = @sql + ' FROM ' + @DBName + '..ExtSrceFullUpdate a';

SET @sql = @sql + ' LEFT OUTER JOIN ' + @DBName + '..ExtSrceFullSound b ON a.UID = b.UID';

SET @sql = @sql + ' WHERE a.LastName IS NOT NULL AND a.FirstName IS NULL AND b.UID IS NULL';

END

EXECUTE (@sql);

A screenshot of a computer

Description automatically generated

# Phonetic cross control

Here is an extract of the part of the script related to the phonetic control during the cross-control process. This control is applied only when length of the name controlled is longer than 3 characters. Otherwise, too many alerts will be pointed out.

In the same way, to avoid too many alerts, an additional control is done on the birthdate or age of the individual controlled.

For a company, the part related to the first name is removed.

This type of control, following these rules & criteria, must be adapted to the control method developed in the supplier's software and cannot be repeated as is.

IF LEN(@NameIn) > 3

BEGIN

SET @SQL = @SQL + ' UNION ALL'

SET @SQL = @SQL + ' (SELECT Ext.UID'

SET @SQL = @SQL + ' , Scenario = 600 '

SET @SQL = @SQL + ' FROM ' + @DBName + '..ExtSrceFullSound' + @TableContains + ' ExtS '

SET @SQL = @SQL + ' INNER JOIN ' + @DBName + '..ExtSrceFull' + @TableContains + ' Ext ON ExtS.[UID] = Ext.[UID]'

SET @SQL = @SQL + ' WHERE (ExtS.LastNameMetaphoneCustom = dbo.fnDoubleMetaphoneScalarCustom(1, ''' + @NameIn + ''', 10) '

SET @SQL = @SQL + ' AND ExtS.FirstNameMetaphoneCustom = dbo.fnDoubleMetaphoneScalarCustom(1, ''' + @FirstNameIn + ''', 10) '

SET @SQL = @SQL + ' AND ExtS.LastNameSoundEx = SOUNDEX(''' + @NameIn + ''') '

SET @SQL = @SQL + ' AND ExtS.FirstNameSoundEx = SOUNDEX(''' + @FirstNameIn + ''') '

SET @SQL = @SQL + ' AND LEN(''' + @NameIn + ''') > 3 '

SET @SQL = @SQL + ' AND ((LEFT(Ext.Dob, 4) = ''' + @YearBirthDate + ''' AND ((CONVERT(int, SUBSTRING(Ext.Dob, 6,2)) = 0) OR CONVERT(int, SUBSTRING(Ext.Dob, 6,2)) = ' + CONVERT(nvarchar(2), @MonthBirthDate) + ' ))'

SET @SQL = @SQL + ' OR ''' + @YearBirthDate + ''' IN (YEAR(Ext.[AgeDate]) - Ext.[Age], YEAR(Ext.[AgeDate]) - Ext.[Age] - 1))'

SET @SQL = @SQL + ' )'

SET @SQL = @SQL + ' )'

END

# Double metaphone algorithm

An example of the Double metaphone algorithm matching, to be adapted, is given here below.

/\*#########################################################################

Double Metaphone Phonetic Matching Function

This reduces word to approximate phonetic string. This is deliberately

not a direct phonetic

Based off original C++ code and algorithm by

Lawrence Philips (lphilips\_AT\_verity.com)

Published in the C/C++ Users Journal:

http://www.cuj.com/articles/2000/0006/0006d/0006d.htm?topic=articles

Original Metaphone presented in article in "Computer Language" in 1990.

Reduces alphabet to

The 14 constonant sounds:

"sh" "p"or"b" "th"

| | |

X S K J T F H L M N P R 0 W

Drop vowels except at the beginning

Produces a char(10) string. The left(@result,5) gives the most common

pronouciation, right(@result,5) gives the commonest alternate.

Translated into t-SQL by Keith Henry (keithh\_AT\_lbm-solutions.com)

#########################################################################\*/

Declare @original varchar(70),

@primary varchar(70),

@secondary varchar(70),

@length int,

@last int,

@current int,

@strcur1 char(1) ,

@strnext1 char(1) ,

@strprev1 char(1),

@SlavoGermanic bit

set @SlavoGermanic = 0

set @primary = ''

set @secondary = ''

set @current = 1

set @length = len(@str)

set @last = @length

set @original = isnull(@str,'') + ' '

set @original = upper(@original)

if patindex('%[WK]%',@str) + charindex('CZ',@str) + charindex('WITZ',@str) <> 0

set @SlavoGermanic = 1

-- skip this at beginning of word

if substring(@original, 1, 2) in ('GN', 'KN', 'PN', 'WR', 'PS')

set @current = @current + 1

-- Initial 'X' is pronounced 'Z' e.g. 'Xavier'

if substring(@original, 1, 1) = 'X'

begin

set @primary = @primary + 'S' -- 'Z' maps to 'S'

set @secondary = @secondary + 'S'

set @current = @current + 1

end

if substring(@original, 1, 1) in ('A', 'E', 'I', 'O', 'U', 'Y')

begin

set @primary = @primary + 'A' -- all init vowels now map to 'A'

set @secondary = @secondary + 'A'

set @current = @current + 1

end

while @current <= @length

begin

if len(@primary) >= 5 break

set @strcur1 = substring(@original, @current, 1)

set @strnext1 = substring(@original, (@current + 1), 1)

set @strprev1 = substring(@original, (@current - 1), 1)

if @strcur1 in ('A', 'E', 'I', 'O', 'U', 'Y')

set @current = @current + 1

else

if @strcur1 = 'B' -- '-mb', e.g. 'dumb', already skipped over ...

begin

set @primary = @primary + 'P'

set @secondary = @secondary + 'P'

if @strnext1 = 'B'

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'Ç'

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

set @current = @current + 1

end

else

if @strcur1 = 'C'

begin

if @strnext1 = 'H'

begin

if substring(@original, @current, 4) = 'CHIA' -- italian 'chianti'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

if @current > 1 -- find 'michael'

and substring(@original, @current, 4) = 'CHAE'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'X'

end

else

begin

if @current = 1 -- greek roots e.g. 'chemistry', 'chorus'

and (substring(@original, @current + 1, 5) in ('HARAC', 'HARIS')

or substring(@original, @current + 1, 3) in ('HOR', 'HYM', 'HIA', 'HEM')

)

and substring(@original, 1, 5) <> 'CHORE'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

if ( substring(@original, 0, 4) in ('VAN ', 'VON ') -- germanic, greek, or otherwise 'ch' for 'kh' sound

or substring(@original, 0, 3) = 'SCH'

)

or substring(@original, @current - 2, 6) in ('ORCHES', 'ARCHIT', 'ORCHID') -- 'architect' but not 'arch', orchestra', 'orchid'

or substring(@original, @current + 2, 1) in ('T', 'S')

or ( ( @strprev1 in ('A','O','U','E')

or @current = 0

)

and substring(@original, @current + 2, 1) in ('L','R','N','M','B','H','F','V','W',' ') -- e.g. 'wachtler', 'weschsler', but not 'tichner'

)

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

if (@current > 1)

begin

if substring(@original, 1, 2) = 'MC' -- e.g. 'McHugh'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'K'

end

end

else

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

end

end

end

end

end

set @current = @current + 2

end --ch logic

else

begin

if @strnext1 = 'C' -- double 'C', but not McClellan'

and not(@current = 1

and substring(@original, 1, 1) = 'M'

)

begin

if substring(@original, @current + 2, 1) in ('I','E','H') -- 'bellocchio' but not 'bacchus'

and substring(@original, @current + 2, 2) <> 'HU'

begin

if ( @current = 2 -- 'accident', 'accede', 'succeed'

and @strprev1 = 'A'

)

or substring(@original, @current - 1, 5) in ('UCCEE', 'UCCES')

begin

set @primary = @primary + 'KS'

set @secondary = @secondary + 'KS'

end

else

begin -- 'bacci', 'bertucci', other italian

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

-- e.g. 'focaccia' if substring(@original, @current, 4) = 'CCIA'

end

set @current = @current + 3

end

else

begin

set @primary = @primary + 'K' -- Pierce's rule

set @secondary = @secondary + 'K'

set @current = @current + 2

end

end

else

begin

if @strnext1 in ('K','G','Q')

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

set @current = @current + 2

end

else

begin

if @strnext1 in ('I','E','Y')

begin

if substring(@original, @current, 3) in ('CIO','CIE','CIA') -- italian vs. english

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'X'

end

else

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

set @current = @current + 2

end

else

begin

if @strnext1 = 'Z' -- e.g. 'czerny'

and substring(@original, @current -2, 4) <> 'WICZ'

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'X'

set @current = @current + 2

end

else

begin

if @current > 2 -- various gremanic

and substring(@original, @current - 2,1) not in ('A', 'E', 'I', 'O', 'U', 'Y')

and substring(@original, @current - 1, 3) = 'ACH'

and ((substring(@original, @current + 2, 1) <> 'I')

and ((substring(@original, @current + 2, 1) <> 'E')

or substring(@original, @current - 2, 6) in ('BACHER', 'MACHER')

)

)

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

set @current = @current + 2

end

else

begin

if @current = 1 -- special case 'caesar'

and substring(@original, @current, 6) = 'CAESAR'

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

set @current = @current + 2

end

else

begin -- final else

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

if substring(@original, @current + 1, 2) in (' C',' Q',' G') -- name sent in 'mac caffrey', 'mac gregor'

set @current = @current + 3

else

set @current = @current + 1

end

end

end

end

end

end

end

end

else

if @strcur1 = 'D'

begin

if substring(@original, @current, 2) = 'DG'

begin

if substring(@original, @current + 2, 1) in ('I','E','Y')

begin

set @primary = @primary + 'J' -- e.g. 'edge'

set @secondary = @secondary + 'J'

set @current = @current + 3

end

else

begin

set @primary = @primary + 'TK' -- e.g. 'edgar'

set @secondary = @secondary + 'TK'

set @current = @current + 2

end

end

else

begin

if substring(@original, @current, 2) in ('DT','DD')

begin

set @primary = @primary + 'T'

set @secondary = @secondary + 'T'

set @current = @current + 2

end

else

begin

set @primary = @primary + 'T'

set @secondary = @secondary + 'T'

set @current = @current + 1

end

end

end

else

if @strcur1 = 'F'

begin

set @primary = @primary + 'F'

set @secondary = @secondary + 'F'

if (@strnext1 = 'F')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'G'

begin

if (@strnext1 = 'H')

begin

if @current > 1

and @strprev1 not in ('A', 'E', 'I', 'O', 'U', 'Y')

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

if not( (@current > 2 -- Parker's rule (with some further refinements) - e.g. 'hugh'

and substring(@original, @current - 2, 1) in ('B','H','D')

) -- e.g. 'bough'

or (@current > 3

and substring(@original, @current - 3, 1) in ('B','H','D')

) -- e.g. 'broughton'

or (@current > 4

and substring(@original, @current - 4, 1) in ('B','H')

) )

begin

if @current > 3 -- e.g. 'laugh', 'McLaughlin', 'cough', 'gough', 'rough', 'tough'

and @strprev1 = 'U'

and substring(@original, @current - 3, 1) in ('C','G','L','R','T')

begin

set @primary = @primary + 'F'

set @secondary = @secondary + 'F'

end

else

begin

if @current > 1

and @strprev1 <> 'I'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

if (@current < 4)

begin

if (@current = 1) -- 'ghislane', 'ghiradelli'

begin

if (substring(@original, @current + 2, 1) = 'I')

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'J'

end

else

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

end

end

end

end

end

end

set @current = @current + 2

end

else

begin

if (@strnext1 = 'N')

begin

if @current = 1

and substring(@original, 0,1) in ('A', 'E', 'I', 'O', 'U', 'Y')

and @SlavoGermanic = 0

begin

set @primary = @primary + 'KN'

set @secondary = @secondary + 'N'

end

else

begin

-- not e.g. 'cagney'

if substring(@original, @current + 2, 2) = 'EY'

and (@strnext1 <> 'Y')

and @SlavoGermanic = 0

begin

set @primary = @primary + 'N'

set @secondary = @secondary + 'KN'

end

else

begin

set @primary = @primary + 'KN'

set @secondary = @secondary + 'KN'

end

end

set @current = @current + 2

end

else

begin

if substring(@original, @current + 1, 2) = 'LI' -- 'tagliaro'

and @SlavoGermanic = 0

begin

set @primary = @primary + 'KL'

set @secondary = @secondary + 'L'

set @current = @current + 2

end

else

begin

if @current = 1 -- -ges-, -gep-, -gel- at beginning

and (@strnext1 = 'Y'

or substring(@original, @current + 1, 2) in ('ES','EP','EB','EL','EY','IB','IL','IN','IE', 'EI','ER')

)

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'J'

set @current = @current + 2

end

else

begin

if (substring(@original, @current + 1, 2) = 'ER' -- -ger-, -gy-

or @strnext1 = 'Y'

)

and substring(@original, 1, 6) not in ('DANGER','RANGER','MANGER')

and @strprev1 not in ('E', 'I')

and substring(@original, @current - 1, 3) not in ('RGY','OGY')

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'J'

set @current = @current + 2

end

else

begin

if @strnext1 in ('E','I','Y') -- italian e.g. 'biaggi'

or substring(@original, @current -1, 4) in ('AGGI','OGGI')

begin

if (substring(@original, 1, 4) in ('VAN ', 'VON ') -- obvious germanic

or substring(@original, 1, 3) = 'SCH'

)

or substring(@original, @current + 1, 2) = 'ET'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

end

else

begin

-- always soft if french ending

if substring(@original, @current + 1, 4) = 'IER '

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'J'

end

else

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'K'

end

end

set @current = @current + 2

end

else

begin -- other options exausted call it k sound

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

if (@strnext1 = 'G')

set @current = @current + 2

else

set @current = @current + 1

end

end

end

end

end

end

end

else

if @strcur1 = 'H'

begin

if (@current = 0 -- only keep if first & before vowel or btw. 2 vowels

or @strprev1 in ('A', 'E', 'I', 'O', 'U', 'Y')

)

and @strnext1 in ('A', 'E', 'I', 'O', 'U', 'Y')

begin

set @primary = @primary + 'H'

set @secondary = @secondary + 'H'

set @current = @current + 2

end

else

set @current = @current + 1

end

else

if @strcur1 = 'J'

begin

if substring(@original, @current, 4) = 'JOSE' -- obvious spanish, 'jose', 'san jacinto'

or substring(@original, 1, 4) = 'SAN '

begin

if (@current = 1

and substring(@original, @current + 4, 1) = ' '

)

or substring(@original, 1, 4) = 'SAN '

begin

set @primary = @primary + 'H'

set @secondary = @secondary + 'H'

end

else

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'H'

end

set @current = @current + 1

end

else

begin

if @current = 1

begin

set @primary = @primary + 'J' -- Yankelovich/Jankelowicz

set @secondary = @secondary + 'A'

set @current = @current + 1

end

else

begin

if @strprev1 in ('A', 'E', 'I', 'O', 'U', 'Y') -- spanish pron. of .e.g. 'bajador'

and @SlavoGermanic = 0

and @strnext1 in ('A','O')

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'H'

set @current = @current + 1

end

else

begin

if (@current = @last)

begin

set @primary = @primary + 'J'

set @secondary = @secondary + ''

set @current = @current + 1

end

else

begin

if @strnext1 in ('L','T','K','S','N','M','B','Z')

and @strprev1 not in ('S','K','L')

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'J'

set @current = @current + 1

end

else

begin

if (@strnext1 = 'J') -- it could happen

set @current = @current + 2

else

set @current = @current + 1

end

end

end

end

end

end

else

if @strcur1 = 'K'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

if (@strnext1 = 'K')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'L'

begin

if (@strnext1 = 'L')

begin

if (@current = (@length - 3) -- spanish e.g. 'cabrillo', 'gallegos'

and substring(@original, @current - 1, 4) in ('ILLO','ILLA','ALLE')

)

or ((substring(@original, @last - 1, 2) in ('AS','OS')

or substring(@original, @last, 1) in ('A','O')

)

and substring(@original, @current - 1, 4) = 'ALLE'

)

set @primary = @primary + 'L' --set @secondary = @secondary + ''

set @current = @current + 2

end

else

begin

set @current = @current + 1

set @primary = @primary + 'L'

set @secondary = @secondary + 'L'

end

end

else

if @strcur1 = 'M'

begin

set @primary = @primary + 'M'

set @secondary = @secondary + 'M'

if substring(@original, @current - 1, 3) = 'UMB'

and (@current + 1 = @last

or substring(@original, @current + 2, 2) = 'ER'

) -- 'dumb', 'thumb'

or @strnext1 = 'M'

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 in ('N','Ñ')

begin

set @primary = @primary + 'N'

set @secondary = @secondary + 'N'

if @strnext1 in ('N','Ñ')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'P'

begin

if (@strnext1 = 'H')

begin

set @current = @current + 2

set @primary = @primary + 'F'

set @secondary = @secondary + 'F'

end

else

begin

-- also account for 'campbell' and 'raspberry'

if @strnext1 in ('P','B')

set @current = @current + 2

else

begin

set @current = @current + 1

set @primary = @primary + 'P'

set @secondary = @secondary + 'P'

end

end

end

else

if @strcur1 = 'Q'

begin

set @primary = @primary + 'K'

set @secondary = @secondary + 'K'

if (@strnext1 = 'Q')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'R'

begin

if @current = @last -- french e.g. 'rogier', but exclude 'hochmeier'

and @SlavoGermanic = 0

and substring(@original, @current - 2, 2) = 'IE'

and substring(@original, @current - 4, 2) not in ('ME','MA')

set @secondary = @secondary + 'R' --set @primary = @primary + ''

else

begin

set @primary = @primary + 'R'

set @secondary = @secondary + 'R'

end

if (@strnext1 = 'R')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'S'

begin

if substring(@original, @current - 1, 3) in ('ISL','YSL') -- special cases 'island', 'isle', 'carlisle', 'carlysle'

set @current = @current + 1 --silent s

else

begin

if substring(@original, @current, 2) = 'SH'

begin

-- germanic

if substring(@original, @current + 1, 4) in ('HEIM','HOEK','HOLM','HOLZ')

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

else

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

end

set @current = @current + 2

end

else

begin

-- italian & armenian

if substring(@original, @current, 3) in ('SIO','SIA')

or substring(@original, @current, 4) in ('SIAN')

begin

if @SlavoGermanic = 0

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'X'

end

else

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

set @current = @current + 3

end

else

begin

if (@current = 1 -- german & anglicisations, e.g. 'smith' match 'schmidt', 'snider' match 'schneider'

and @strnext1 in ('M','N','L','W') -- also, -sz- in slavic language altho in hungarian it is pronounced 's'

)

or @strnext1 = 'Z'

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'X'

if @strnext1 = 'Z'

set @current = @current + 2

else

set @current = @current + 1

end

else

begin

if substring(@original, @current, 2) = 'SC'

begin

if substring(@original, @current + 2, 1) = 'H' -- Schlesinger's rule

begin

if substring(@original, @current + 3, 2) in ('OO','ER','EN','UY','ED','EM') -- dutch origin, e.g. 'school', 'schooner'

begin

if substring(@original, @current + 3, 2) in ('ER','EN') -- 'schermerhorn', 'schenker'

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'SK'

end

else

begin

set @primary = @primary + 'SK'

set @secondary = @secondary + 'SK'

end

set @current = @current + 3

end

else

begin

if @current = 1

and substring(@original, 3,1) not in ('A', 'E', 'I', 'O', 'U', 'Y')

and substring(@original, @current + 3, 1) <> 'W'

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'S'

end

else

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

end

set @current = @current + 3

end

end

else

begin

if substring(@original, @current + 2, 1) in ('I','E','Y')

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

else

begin

set @primary = @primary + 'SK'

set @secondary = @secondary + 'SK'

end

set @current = @current + 3

end

end

else

begin

if @current = 1 -- special case 'sugar-'

and substring(@original, @current, 5) = 'SUGAR'

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'S'

set @current = @current + 1

end

else

begin

if @current = @last -- french e.g. 'resnais', 'artois'

and substring(@original, @current - 2, 2) in ('AI','OI')

set @secondary = @secondary + 'S' --set @primary = @primary + ''

else

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

if @strnext1 in ('S','Z')

set @current = @current + 2

else

set @current = @current + 1

end

end

end

end

end

end

end

else

if @strcur1 = 'T'

begin

if substring(@original, @current, 4) = 'TION'

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

set @current = @current + 3

end

else

if substring(@original, @current, 3) in ('TIA','TCH')

begin

set @primary = @primary + 'X'

set @secondary = @secondary + 'X'

set @current = @current + 3

end

else

if substring(@original, @current, 2) = 'TH'

or substring(@original, @current, 3) = 'TTH'

begin

if substring(@original, @current + 2, 2) in ('OM','AM') -- special case 'thomas', 'thames' or germanic

or substring(@original, 0, 4) in ('VAN ','VON ')

or substring(@original, 0, 3) = 'SCH'

begin

set @primary = @primary + 'T'

set @secondary = @secondary + 'T'

end

else

begin

set @primary = @primary + '0'

set @secondary = @secondary + 'T'

end

set @current = @current + 2

end

else

begin

if @strnext1 in ('T','D')

begin

set @current = @current + 2

set @primary = @primary + 'T'

set @secondary = @secondary + 'T'

end

else

begin

set @current = @current + 1

set @primary = @primary + 'T'

set @secondary = @secondary + 'T'

end

end

end

else

if @strcur1 = 'V'

if (@strnext1 = 'V')

set @current = @current + 2

else

begin

set @current = @current + 1

set @primary = @primary + 'F'

set @secondary = @secondary + 'F'

end

else

if @strcur1 = 'W'

begin

-- can also be in middle of word

if substring(@original, @current, 2) = 'WR'

begin

set @primary = @primary + 'R'

set @secondary = @secondary + 'R'

set @current = @current + 2

end

else

if @current = 1

and (@strnext1 in ('A', 'E', 'I', 'O', 'U', 'Y')

or substring(@original, @current, 2) = 'WH'

)

begin

if @strnext1 in ('A', 'E', 'I', 'O', 'U', 'Y') -- Wasserman should match Vasserman

begin

set @primary = @primary + 'A'

set @secondary = @secondary + 'F'

set @current = @current + 1

end

else

begin

set @primary = @primary + 'A' -- need Uomo to match Womo

set @secondary = @secondary + 'A'

set @current = @current + 1

end

end

else

if (@current = @last -- Arnow should match Arnoff

and @strprev1 in ('A', 'E', 'I', 'O', 'U', 'Y')

)

or substring(@original, @current - 1, 5) in ('EWSKI','EWSKY','OWSKI','OWSKY')

or substring(@original, 0, 3) = 'SCH'

begin

set @secondary = @secondary + 'F' --set @primary = @primary + ''

set @current = @current + 1

end

else

if substring(@original, @current, 4) in ('WICZ','WITZ') -- polish e.g. 'filipowicz'

begin

set @primary = @primary + 'TS'

set @secondary = @secondary + 'FX'

set @current = @current + 4

end

else

set @current = @current + 1 -- else skip it

end

else

if @strcur1 = 'X'

begin

if not (@current = @last -- french e.g. breaux

and (substring(@original, @current - 3, 3) in ('IAU', 'EAU')

or substring(@original, @current - 2, 2) in ('AU', 'OU')

)

)

begin

set @primary = @primary + 'KS'

set @secondary = @secondary + 'KS'

end --else skip it

if @strnext1 in ('C','X')

set @current = @current + 2

else

set @current = @current + 1

end

else

if @strcur1 = 'Z'

begin

if (@strnext1 = 'Z')

set @current = @current + 2

else

begin

if (@strnext1 = 'H') -- chinese pinyin e.g. 'zhao'

begin

set @primary = @primary + 'J'

set @secondary = @secondary + 'J'

set @current = @current + 2

end

else

begin

if (substring(@original, @current + 1, 2) in ('ZO', 'ZI', 'ZA'))

or (@SlavoGermanic = 1

and (@current > 0

and @strprev1 <> 'T'

)

)

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'TS'

end

else

begin

set @primary = @primary + 'S'

set @secondary = @secondary + 'S'

end

end

set @current = @current + 1

end

end

else

set @current = @current + 1

end

return cast(@primary as char(5)) + cast(@secondary as char(5))