Version

3.0

pdNickname User Guide

Name and Nickname Database

A one-of-a-kind proprietary resource designed to facilitate matching given names and nicknames that are variations or phonetically similar. The package also includes extensive language of origin and use information, and the *Pro* edition includes fuzzy logic to match typographical errors. Ancestry researchers, students, teachers, and scholars benefit as well because this software is recommended for study in genealogy, onomatology, anthroponymy, ethnology, linguistics, and related disciplines.

Peacock Data, Inc.
California, USA 800-609-9231
Web address: www.peacockdata.com

TABLE OF CONTENTS

INTRODUCTION

Matching and merging names can be tricky. How do you relate William Smith with Bill Smith? The answer is **pdNickname**. It is designed to facilitate matching names that are dissimilar because one is a given name while another is a nickname or other variation.

Coverage includes hundreds of thousands of names and the package employs the best matching algorithms designed for this process. The software is a one-of-a-kind proprietary resource that for more than 20 years has been utilized by businesses and organizations around the world in applications you use every day.

As an added benefit, languages of origin and use have also been researched and included, and the enhanced version even incorporates sophisticated fuzzy logic which allows matching when lists have typographical errors.

This easy-to-use, comprehensive, and up-to-date software is of great value for businesses and organizations working with lists of names, but ancestry researchers, students, teachers, and scholars benefit as well because this software is recommended for study in genealogy, onomatology, anthroponymy, ethnology, linguistics, and related disciplines.

pdNickname is available in **Pro** and **Standard** editions. This guide covers both versions.

- ✓ **PRO**: software includes more than 40 million regular first name and nickname variation records and almost 10 million fuzzy logic records based on more than 397,000 name formations, with the relationship identified for name pairs along with languages of origin and use.

- ✓ **STANDARD**: software includes more than 40 million regular first name variation records based on more than 397,000 name formations, with the relationship identified for name pairs along with languages of origin and use. It has all the features of the *Pro* version except the fuzzy logic records.

IMPORTING DATA INTO YOUR SYSTEM

pdNickname is designed to be compatible with any database system. It comes in multiple file formats, uses only the ANSI character set, and has a well-defined layout.

INCLUDED DATABASE FILES

pdNickname has two data sets, a names database and a relationship file.

Included files are:

NAMES DATABASE

This data set lists more than 397,000 given name and nickname formations. The relationship file is based on these names. The names database also includes a name ranking, archaic flag, and languages of origin and use.

RELATIONSHIP FILE

This data set has more than 40 million (*Standard* edition) or 50 million (*Pro* edition) name-pair records based on the given names and nicknames in the names database. It includes the pairs of related names and nicknames along with extensive information about their onomastic and phonetic relationships. A scoring system is provided to order matches from most likely to least likely. The relationship file is divided into sections for user convenience and to allow users more options when setting up their system.

FILE FORMATS

The database is available in three common file formats. Each format contains the same data.

Available file formats are:

CSV-COMMA SEPARATED VALUES

Files in Comma Separated Values (CSV) format (also known as Comma Delimited) separate fields with commas, and alpha/numeric character fields are usually delimited with double quotes (in case some of the field content includes commas). This format is the most commonly used. It is a native format for Microsoft Excel and is compatible with nearly all database management systems and spreadsheets.

TXT-FIXED LENGTH

Files in Fixed Length (TXT) format (also known as Standard Data Format or SDF) use constant field positions and lengths for all records. In other words, each field starts and ends at the same place in the text file and each record is on a separate line. While not as popular as comma separated values, this format is preferred by many due to its input precision and is widely used to transfer data between different software programs. It is compatible with most database management systems and spreadsheets.

DBF-DATABASE

Files in DBF database format (also known as xBase) are native to Microsoft FoxPro and Visual FoxPro, dataBased Intelligence dBase, Alaska Software XBase++, Apollo Database Engine, Apycom Software DBFView, Astersoft DBF Manager, DS-Datasoft Visual DBU, Elsoft DBF Commander, GrafX Software Clipper and Vulcan.NET, Multisoft FlagShip, Recital Software Recital, Software Perspectives Cule.Net, and xHarbour.com xHarbour. They are also compatible with any database management system that can import the DBF (xBase) format, such as Microsoft Access, Microsoft SQL Server, and numerous others.

CHARACTER SET

The ANSI character set is utilized for all database records. This includes ASCII values 0 to 127 and extended values 128 to 255. These are also known as the extended Latin alphabet. Some users may needs to configure their database system to import the extended values. In many cases the option will be labeled the “Latin-1” character set.

FILE LAYOUTS AND DATA DEFINITIONS

Below are the complete layout specifications and data definitions of all files provided with *pdNickname*.

Each line below contains the following information: FIELD NUMBER: field position number. FIELD NAME: name of field. FIELD TYPE: field data type; "*Chr*" = alpha/numeric characters, "Num" = numbers. FIELD LENGTH: length of field. DECIMAL PLACES: number of decimal places (if any). START POSITION: field starting position. END POSITION: field ending position. DESCRIPTION: data definition of field contents.

LAYOUT OF PDNICKNAME NAMES DATABASE

Field Count: 11
Total Length: 485
Record Count: Pro and Standard: 397,847

FIELD NUMBER	FIELD NAME	FIELD TYPE	FIELD LENGTH	DECIMAL PLACES	START POSITION	END POSITION	DESCRIPTION
1	PEACOCK_ID	Chr	17		1	17	Unique identifier for each record
2	NORMAL	Chr	35		18	52	Normalized name spelling
3	STANDARD	Chr	35		53	87	Standardized name spelling
4	NAME	Chr	35		88	122	Stylized name spelling
5	GENDER	Chr	1		123	123	Gender flag: *M = Male* *F = Female*
6	GIVEN	Chr	1		124	124	Given name flag: G = Is a given name
7	NICK	Chr	1		125	125	Nickname flag: N = Is a nickname
8	RANK	Num	5	0	126	130	Name rank in the United States
9	ARCHAIC	Chr	1		131	131	Archaic name flag: *A = Archaic*
10	LANGUAGE	Chr	254		132	385	Language or languages of origin and use
11	SPECIAL	Chr	100		386	485	Special and unique origins

LAYOUT OF PDNICKNAME RELATIONSHIP FILE

Field Count: 13
Total Length: 82
Record Count: Pro: 51,391,682; Standard: 41,653,922

FIELD NUMBER	FIELD NAME	FIELD TYPE	FIELD LENGTH	DECIMAL PLACES	START POSITION	END POSITION	DESCRIPTION
1	GENDER1	Chr	1		1	35	Gender of the first name in the related name pair
2	NAME1	Chr	35		2	36	First name in the related name pair
3	GENDER2	Chr	1		37	37	Gender of the second name in the related name pair
4	NAME2	Chr	35		38	72	Second name in the related name pair
5	REL	Chr	1		73	73	Relationship flag: *1 = Close onomastic variant* *2 = Near onomastic variant* *3 = Distant onomastic variant* *S = Short form nickname* *D = Diminutive nickname* *P = Phonetic match* *X = Opposite gender match* *F = Fuzzy logic match* (*Pro* only)
6	SCORE	Chr	2		74	75	Match quality score: *01 (best) to 99 score; 99 is reserved for archaic matches; 00 is entered for opposite gender and fuzzy logic matches*
7	DMP	Chr	1		76	76	Double Metaphore: *P = Primary line match* *S = Secondary line match*
8	MP	Chr	1		77	77	Metaphone: *M = match*
9	NY	Chr	1		78	78	New York State Identification and Intelligence System (NYSIIS): *N = match*
10	CP	Chr	1		79	79	Caverphone: *C = match*
11	SX	Chr	1		80	80	Soundex: *S = match*
12	DMSX	Chr	1		81	81	Daitch–Mokotoff Soundex: *P = Primary line match* *S = Secondary line match*
13	DIR	Chr	1		82	82	Name pair direction flag: *A = Name pair is in standard order* *R = Name pair is in reverse order*

USING THE NAMES DATABASE

***pdNickname* has two data sets—a names database listing all included given names and nicknames, and a separate file with name-pair relationships. This section discusses the names database.**

The names database lists all the given names and nicknames in the relationship file along with additional names for which no onomastic or phonetic connection exists. It is located at:

- Names Database\pdNickname_NamesDatabase.CSV [or .TXT, .DBF]

For a large majority, the language or languages of origin and use have also been researched and included, and names from the United States provide a national popularity ranking.

PEACOCK_ID FIELD

FIELDS

PEACOCK_ID | Unique identification number (primary key)
Each record has a 17-character alphanumeric primary key that uniquely distinguishes it from all other records in the table.

The first field in the names database is PEACOCK_ID. It provides a unique primary key identifier for each record. Each begins with the character "n" to identify the software product. Each identification number has three parts and each part is separated with a hyphen.

The following is the first PEACOCK_ID in the names database:

- **n0000001-001-001F** is a complete PEACOCK_ID; no other record has this same exact identification

PARTS OF PEACOCK_ID

The PEACOCK_ID primary key is made up of the following three parts:

n0000001 is the first part of a PEACOCK_ID; it identifies each normalized name spelling; multiple records can have this same number.

n0000001-001 are the first and second parts of a PEACOCK_ID; together they identify each standardized name spelling; multiple records can have this same number.

n0000001-001-001F are all three parts of a PEACOCK_ID; together they identify each unique stylized name spelling and gender; multiple records cannot have this same number.

NAME FIELDS

FIELDS

NORMAL | Normalized first name spelling
STANDARD | Standardized first name spelling
NAME | Stylized first name spelling
Each record has three up to 35-character alphabetic names that provide the normalized, standardized, and stylized spelling of the same name. All names are in UPPER CASE.

Each record lists the same given name or nickname spelled in up to three different ways—stylized, standardized, and normalized. These are different ways of writing the same name. All are in UPPER CASE. **The relationship file is based on the standardized version of the name.**

Note that fewer than nine percent of the names in the names database contain any styling or special characters so most of the time all three versions of the name will be identical.

You can find examples of stylized, standardized, and normalized name spellings at the end of this section.

NAME (STYLIZED)

These are given names and nicknames as they are commonly spelled on name lists with any styling intact. Names sometimes have hyphens, periods, spaces, and other special characters. When a name exists in multiple styles, it is grouped and listed separately for each stylization. Different styling and the use of special characters can indicate different languages of origin and use. All names are in UPPER CASE.

STANDARDIZED

These are the same as stylized first names, but with all spaces, periods, hyphens, and apostrophizes removed. **This formation is used to build and compare the names in the relationship file.** Because names can be written in a number of ways, removing these characters makes the name data easier to match. Most database systems have simple commands to accomplish this. Most names, particularly English and Americanized names, do not require standardization. Standardization transforms the following characters:

From	To	Description
-	*	Hyphen
.	*	Period

* = Removed

From	To	Description
‘	*	Left single quote/apostrophe
’	*	Right single quote/apostrophe

* = Removed

From	To	Description
'	*	Apostrophe
	*	Blank space

* = Removed

NORMALIZED

These are the same as standardized first names, but with all non-English alphabetic letters and glyphs converted to English A—Z letters, including grave accents, acute accents, umlauts, and other values in the extended ANSI character set. Additionally, names starting with a "SAINT" or "STE" prefix are converted to "ST". **Users do not need to normalize name information to match against the relationship file, but the provided special character database will need to be installed if lists have extended ANSI characters.** Most names, particularly English and Americanized names, do not require normalization. Normalization transforms the following characters:

From	To	Description
À	A	A-grave
Á	A	A-acute
Â	A	A-circumflex
Ã	A	A-tilde
Ä	A	A-diaeresis (umlaut)
Å	A	A-ring
Æ	AE	Æsc (grapheme)
Ç	C	C-cedilla
Ð	D	Eth
È	E	E-grave
É	E	E-acute
Ê	E	E-circumflex
Ë	E	E-diaeresis (umlaut)

From	To	Description
Ì	I	I-grave
Í	I	I-acute
Î	I	I-circumflex
Ï	I	I-diaeresis (umlaut)
Ñ	N	N-tilde
Ò	O	O-grave
Ó	O	O-acute
Ô	O	O-circumflex
Õ	O	O-tilde
Ö	O	O-diaeresis (umlaut)
Ø	O	Ø-vowel
Œ	OE	Œ (grapheme)
Š	S SH	S-caron (grapheme)**

From	To	Description
ß	SS	Eszett/Sharp S
Ù	U	U-grave
Ú	U	U-acute
Û	U	U-circumflex
Ü	U	U-diaeresis (umlaut)
Ý	Y	Y-acute
Þ	Y TH P	Þorn (Thorn)[þ]
Ž	Z	Z-caron (grapheme)
¸	*	Spaced cedilla
`	*	Spaced grave accent
´	*	Spaced acute accent
SAINT	ST	SAINT-prefix***
STE	ST	STE-prefix***

* = Removed

** The Š (S-caron) grapheme can be interpreted in two ways, as an "S" or "SH". The "S" transformation in provided in the NORMAL field of the names database, but in the relationship file, the S-caron is transformed and related in both ways so it matches regardless of the interpretation.

*** The SAINT/STE to ST transformations should only be performed when they are being used as prefixes and are followed by a period, hyphen, or space.

[þ] The Þorn (Thorn) can be interpreted in three ways. Some transform it to a "P" because it obviously looks a lot like the letter. But in the days of the Anglo-Saxons it was used to pronounce what the Norman French would later introduce as the digraph "TH". However, after the invention of the printing press, parts of the Þorn were abbreviated or dropped to the point it resembled a "Y", which is the reason for such articulations as "Ye Olde". In

fact, the “Ye” pronunciation is still used informally today in Hiberno-English (Irish English). In homage to Old English, the “Y” transformation in provided in the NORMAL field of the names database, but in the relationship file, the Þorn is transformed and related in all three ways so it matches regardless of the interpretation.

EXAMPLES

The following are examples of stylized, standardized, and normalized names. Note when the stylized, standardized, and normalized names are all different, all the same, and when the stylized name is different but the standardized and normalized names the same:

Normalized	Standardized	Stylized	Gender	Language
ABDULAZIZ	ABDULAZIZ	ABDUL 'AZIZ	Male	Arabic
ANAMARIA	ANAMARÍA	ANA MARÍA	Female	Spanish
FERNANDOJOSE	FERNANDOJOSÉ	FERNANDO JOSÉ	Male	Portuguese, Spanish
FREDERICK	FRÉDÉRICK	FRÉDÉRICK	Male	French
JOYCE	JOYCE	JOYCE	Female	English
KATHERINE	KATHÉRINE	KATHÉRINE	Female	Swedish
MARYLOUISE	MARYLOUISE	MARY LOUISE	Female	English
ROGNVALD	RÖGNVALD	RÖGNVALD	Male	Icelandic
STGRELLAN	SAINTGRELLAN	SAINT GRELLAN	Male	Old Irish
SANDEEPKUMARA	SANDEEPKUMARA	SANDEEP KUMARA	Male	Hindi
YIGYEONG	YIGYEONG	YI-GYEONG	Female	South Korean
ZYRIEL	ZYRIEL	ZYRIEL	Female	West Pacific Filipino Tagalog

GENDER

FIELDS

GENDER | Male or female gender flag
M = Male
F = Female
Each record has a one-character alphabetic codes that indicate the gender associated with the name.

The gender associated with the name is entered in the GENDER field, “M” for male and “F” for female. When the same spelling of a name belongs to both genders, there will be two records, one for each gender. These are unisex names.

GIVEN AND NICK FIELDS

FIELDS

GIVEN | Given name flag
Each record has a one-character alphabetic code that indicates if the name is a given name:
G = Given
Blank = Not a given name

NICK | Nickname flag
Each record has a one-character alphabetic code that indicates if the name is a nickname:
N = Nickname
Blank = Not a nickname

The names database lists all the given names and nicknames in the relationship file along with additional names for which no onomastic or phonetic connection exists. Given names are flagged with a “G” in the GIVEN field, while nicknames are flagged with an “N” in the NICK field.

Names can be flagged both as a given name and as a nickname. It is not uncommon for a nickname over time to be accepted independently as a given name. For example, “Kate”, a nickname for “Katherine” and “Katarina”, is now also considered a proper given name.

The names database contains all the first name spellings gathered and published by the U.S. Census Bureau and Social Security Administration between 1800 and the present time, related nicknames, and ethnic given names and nicknames not found in the United States. About 75 percent of the given names and nicknames can be found in the United States, and the remainder only found outside the United States.

About 42% of the names in the names database are classified as given names; about 56% are classified as nicknames; and about 2% are classified as both a given name and a nickname.

Given names and nicknames are defined as follows:

GIVEN NAMES

These are formal first names (also known as personal name and forenames), normally bestowed upon, or given by parents, at or near the time of birth. This contrasts to surnames which are normally inherited and shared with other immediate family members. There are frequently numerous variations of the same given name often in multiple languages. In addition to regular given names, the names database also provides translations from other languages, and transcriptions, which are variations spelled phonetically as they sound to the person hearing and transcribing the name.

NICKNAMES

These are substitutes for the formal given names that express familiarity or endearment. For example, “Mike” is a nickname for the given name “Michael”. Nicknames can include shortened forms of the given names or diminutives.

NAME RANK

FIELDS

RANK | United States name rank
Records have an up five digit numeric value indicating the rank of first names occurring in the United States 1915 to the present as published by the U.S. Social Security Administration.

The U.S. Social Security Administration, in association with the U.S. Census Bureau, publishes a list of first names occurring 5 times or more in the United States from 1880 to the present. The names database includes all of these names and numerically ranks those with entries occurring from 1915 to the present in order of popularity, starting with the male name “James” in the #1 position, with 4,855,084 entries; and ending with 14,752 names all ranked #88,434, with five entries each. Names not occurring since 1915 or not in the Social Security data set are not ranked.

ARCHAIC NAMES

FIELDS

ARCHAIC | Archaic flag
Each record has a one-character alphabetic code that indicates if the first name is archaic and no longer in use:
A = Archaic name
Blank = Not an archaic name

About 0.8 percent of the names in the names database are archaic. They are included because of their onomastic significance. Archaic names are flagged with an “A” in the ARCHAIC field. Note that names can be archaic for one gender and in use for the opposite gender.

LANGUAGE

FIELDS

LANGUAGE | Language string
Each record has an up to 254-character alphabetic list that indicates the language or languages of origin and use of the first name. Multiple languages are entered as a comma-delimited list with the languages in alphabetical order.

The language or languages of origin and use are identified in the LANGUAGE field. If there is more than one language, they are listed alphabetically in a comma delimited string; for example, "Basque, Catalan, Portuguese, Spanish". None of the languages were derived algorithmically and the provided information represents years of extensive onomastic research. If the language has not been identified but is found in U.S. Social Security Administration or U.S. Census Bureau listings, it is flagged as "American". When different sources list different origins and usages they may be combined depending on the reliability of the source and the reasonability of the information. The languages apply to the stylized name. Differently styled names can have different language values.

Language coverage is extensive. The list exceeds 500 languages, language families, and dialects. Some languages refer to ethnic groups. For example, Bosniak Bosnian referrers to a South Slavic Muslim ethnic group inhabiting mainly Bosnia and Herzegovina.

TOP 30 LANGUAGES

These following are the top 30 languages, out of a total of more than 500 languages, with the number of occurrences in the names database out of a total of 397,000 names. The language count is one for each unique name formation and not one for each relationship (which would be many more):

1.	English	225,000	11.	Spanish	3,300	21.	Czech	2,000
2.	Arabic	46,700	12.	Italian	3,100	22.	Russian	1,900
3.	Turkish	6,700	13.	Bengali	3,100	23.	Dutch	1,800
4.	Punjabi	6,700	14.	German	3,100	24.	Hungarian	1,800
5.	French	5,900	15.	Pashto	3,000	25.	Portuguese	1,700
6.	Iranian	5,800	16.	Norwegian	3,000	26.	Malaysian Malay	1,700
7.	Urdu	4,900	17.	Danish	2,900	27.	Albanian	1,700
8.	Afghan Arabic	4,400	18.	Korean	2,700	28.	Japanese	1,700
9.	Swedish	4,100	19.	Egyptian Arabic	2,400	29.	Bosniak Bosnian	1,500
10.	Finnish	3,600	20.	Polish	2,000	30.	Icelandic	1,400

Note that the counts are rounded to the lower 100.

Also note that the Arabic and Muslim name section is very large due to the many different variations and ways of writing these names. These include theophoric combination names such as those with the religious prefix "Abdul". Both common and uncommon possibilities are included, and the use of Sun Letters in Arabic and Maltese is accounted for.

A list of all the identified languages with counts is included with the software as a Microsoft Excel (XLSX) file. The language names chosen are detailed and easy to search for. For example, to select all 40 African languages, search for “Africa” in the language string, and exclude “African American”. To select all 18 Pacific Island languages, search for “Pacific” and “Oceania”. To select all 12 Indian languages, search for “Indian”, “Hindi”, and “Urdu”. To select all 38 Native American languages, search for “Native American”. Many other useful query words exist which can be determined from the provided list.

A SHORT HISTORY OF FIRST NAMES

First names have been with us for millennia, much longer than last names, which are a relatively modern development. Some of the names in the names database originated during Antiquity while others arose during the subsequent Middle Ages or during modern times. The following is a primer on recognizing these origins and the periods they represent:

ANTIQUITY

Important ancient languages include:

- **Ancient Egyptian**: attested from 3400 BC making it one of the earliest known written languages (along with Sumerian)
- **Sumerian**: the language of ancient Sumer, spoken in southern Mesopotamia (modern Iraq) and closely related to Akkadian, it is attested from 3350 BC making it one of the earliest known written languages (along with Egyptian); it was slowly replaced by Akkadian between the 3rd and 2nd millennia BC, but continued as a classical language until about 100 AD
- **Akkadian**: spoken in ancient Mesopotamia from the 29th through 8th centuries BC, including during the Akkadian Empire (ca. 2334–2193 BC), it is closely related to and replaced Sumerian, and is the earliest attested Semitic language; academic and liturgical use continued until about 100 AD
- **Assyrian**: a dialect of Akkadian spoken in upper-Mesopotamia from about 25th century BC
- **Old Aramaic**: the earliest stages of Aramaic, a Northwest Semitic language subfamily which includes Hebrew and Phoenician, dating from the 10th century BC
- **Avestan**: an Iranian language dating from the Late Bronze Age (1570–1200 BC) known only from its use as the language of Zoroastrian scripture
- **Greek**: spoken on the Balkan Peninsula since the 3rd millennium BC, and the oldest recorded living language, its earliest attested written evidence is the Linear B clay tablet found in Messenia which dates to between 1450 and 1350 BC
- **Hebrew**: a West Semitic language, closely related to Phoenician, historically regarded as the tongue of the Israelites (meaning, “Children [or Sons] of Israel”; its earliest attested written evidence, in form of primitive drawings, dates from the 10th century BC; it was nearly extinct as a spoken language by late Antiquity, but continued to be used as a literary language and as the liturgical language of Judaism, until its revival as a spoken language in the late 19th century
- **Phoenician**: a Northwest Semitic language, closely related to Hebrew, originally spoken in the ancient coastal Mediterranean region of Canaan (roughly corresponding to the Levant) and attested from the 10th until the early 4th century BC

- **Etruscan**: spoken by Etruscan civilization (768—264 BC), in Italy, in the ancient region of Etruria (modern Tuscany plus western Umbria and northern Latium) and in parts of Campania, Lombardy, Veneto, and Emilia-Romagna (where the Etruscans were displaced by the Gauls); it influenced Latin, but was eventually superseded by it.
- **Roman**: spoke Archaic Latin during the Roman Kingdom (753—509 BC) through most the Roman Republic (509—27 BC), replaced by Classical Latin around 75 BC; due to Roman conquests, Latin spread to many Mediterranean and some northern European regions; although considered a "dead" language, Latin is still used in the creation of new words and names in modern languages
- **Koine Greek**: a dialect of Greek spoken in the Eastern Roman Empire from 300 BC to 300 AD; it is also known as Alexandrian dialect, common Attic, and Hellenistic Greek
- **Illyrian**: a family of languages spoken in the western part of the Balkans by a group of Indo-European tribes called the Illyrians; it is attested from about 500 BC
- **Alanic**: spoken by Iranian nomadic pastoral people known as the Alans from the 1st century AD
- **Gallo-Roman**: spoken by the Gauls under provincial rule in the Roman Empire from the 1st century BC to the 5th century AD
- **Proto-Germanic**: dating to the Nordic Bronze Age in Scandinavia (ca. 1700–500 BC) through Antiquity
- **Proto-Celtic**: dating from the British Iron Age (ca. 600 BC—100 AD) through Antiquity
- **Proto-Norse**: a northern dialect of Proto-Germanic from the 2nd century AD until it evolved into Old Norse at the beginning of the Viking Age about 800 AD
- **Primitive Irish**: the oldest known Goidelic language from around the 4th century AD until it evolved into Old Irish about 600 AD; it is only known from fragments, mostly personal names, inscribed on stone in the Ogham alphabet in Ireland and western Great Britain

Names from languages prefixed with "Proto-" are reconstructed and are generally unattested in any documented form.

Late Greek and Late Roman names date from late Antiquity and the early Byzantine period. Late Antiquity is generally considered from the end of the Roman Empire's crisis of the 3rd century (ca. 235–284) to the re-organization of the Eastern Roman Empire under Byzantine Emperor Heraclius and the Islamic conquests during the early and mid 7th century.

Coptic Egyptian is the later stages of the Egyptian language spoken from the 2nd until the 17th century. Today Egyptians mainly speak a dialect of Modern Standard Arabic. Coptic Egyptian is still used as the liturgical language of the Coptic Church.

MIDDLE AGES

Almost all historians agree the Middle Ages began when the political structure of Western Europe changed at the end of the united Roman Empire (476 AD). In the database names dating from the Early Middle Ages (which followed the decline of the Western Roman Empire and is sometimes called the Dark Ages due to the relative scarcity of literary and cultural output during most of the era) are usually prefixed with "Old" such as Old High German and Old Spanish (which still continues as a liturgical language but with a modernized pronunciation). An exception is Old English which is labeled "Anglo-Saxon". Another exception is Old Aramaic which is ancient.

Many languages went through significant changes during the High Middle Ages (a period of rapid population growth and social and political change in Europe from about the 11th through the 13th century) or by the Late Middle Ages (when prosperity and growth in Europe came to a halt and the population experienced a series of famines and plagues). Languages developing in this period are prefixed with “Middle”, or in some cases “Medieval” depending on the accepted terminology.

After Duke William II of Normandy conquered England and killed King Harold II at the Battle of Hastings (1066), the invading Normans and their descendants replaced the Anglo-Saxons as the ruling class of England. French influences were absorbed into the English language, and Old English slowly evolved into Middle English between the 12th and 15th century, additionally aided by influences from the Latin language of the church and the invention of the printing press. Nevertheless, Old English was still used throughout the Plantagenet era (1154–1485), a few years beyond the time Constantinople was finally captured by the Ottoman Turks marking the final end of the Roman Empire (1453), the conclusion of the Middle Ages in the minds of many historians. Of course others cite the Battle of Bosworth Field which established the Tudor dynasty and an era of expansion for England (1485), the conquest of Granada and its annexation by Castile ending Islamic rule (1492), the discovery of the Americas by Christopher Columbus (also 1492), the death of Queen Isabella I of Castile (1504), the death of her spouse King Ferdinand II of Aragon (1516), and the Protestant Reformation (1517) as more appropriate cutoff points, often influenced by the nationality of the historian.

There was no similar revision during the High or Late Middle Ages in many languages, including Spanish, and they do not have a generally recognized middle variety.

Tiberian Hebrew is the canonical pronunciation of the Hebrew Bible (or Tanakh) committed to writing by Masoretic scholars living in the Jewish community of Tiberias in ancient Palestine (ca. 750–950). It is written in a form of Tiberian vocalization dating from the 8th century, but the oral tradition it reflects has ancient roots. Tiberian pronunciation of Hebrew is considered by textual scholars to be the most exact and proper pronunciation of the language as it preserves the original Semitic consonantal and vowel sounds of ancient Hebrew.

Much is unknown about the origin of the Yiddish, a High German language written in the Hebrew alphabet, because most speakers were exterminated in the Holocaust. The consensus among scholars is it emerged among the Ashkenazi Jews in Central Europe between the 10th and 12th centuries and later spread to Eastern Europe in the 16th century.

MODERN

Language formations after the Middle Ages are usually know as modern.

There is frequently confusion about the development of the three modern strains of Gaelic: Irish, Scottish, and Manx. All three sprang from Middle Irish which came from Old Irish.

SPECIAL AND UNIQUE ORIGINS

FIELDS

SPECIAL | Special origin string
Record have an up to 100-character alphabetic list that indicates special and unique characteristics about the origin of the name. Multiple characteristics are entered as a comma-delimited list with the elements in alphabetical order. This field is also used as a flow-over field for the language string.

Many records provide information about special and unique origins. The special and unique characteristics of origin are identified in the SPECIAL field. If there is more than one element, they are listed alphabetically in a comma delimited string; for example, “Biblical, Latinized Greek Mythology, Roman cognomen, Surname”.

Additionally, this field is also used as a flow-over field for the language string when the number of languages exceeds the LANGUAGE field length. In the handful of cases this happens, the additional languages are entered at the beginning of the SPECIAL field following a plus (“+”) sign. If there are also special origins, they are entered after the languages following a semicolon (“;”); for example, “+ Slovene, Swedish; Biblical, Greek byname, Surname”.

Special and unique origins include:

- Names from religion; identifications are:
  - **Biblical**
  - **Quranic**
  - **Sanskrit**
- Bynames: a familiar name for a person, similar to a nickname, that is often used as a replacement for a personal name—for example, Rocky is a common byname for boxers; identifications are:
  - **American English byname**
  - **Ancient Germanic byname**
  - **Anglo-Saxon byname**
  - **English byname**
  - **French byname**
  - **German byname**
  - **Greek byname**
  - **Italian byname**
  - **Middle English byname**
  - **Middle High German byname**
  - **Old French byname**
  - **Old High German byname**
  - **Old Irish byname**
  - **Old Norman French byname**
  - **Old Norse byname**
  - **Old Welsh byname**
  - **Polish byname**

  - **Roman byname**
  - **Welsh byname**
- **History** = Names that became known through historical events
- **Literature** = Literary names created by authors, composers, and poets
- Names from mythology and legend; identifications are:
  - **Anglicized Egyptian Mythology**
  - **Anglicized Greek Mythology**
  - **Anglicized Judeo-Christian Legend**
  - **Anglicized Roman Mythology**
  - **Anglicized Welsh Arthurian Legend**
  - **Anglo-Saxon Arthurian Legend**
  - **Anglo-Saxon Mythology**
  - **Baltic Mythology**
  - **Breton Mythology**
  - **Celtic Mythology**
  - **Egyptian Mythology**
  - **English Arthurian Legend**
  - **European Mythology**
  - **Far Eastern Mythology**
  - **Finnish Mythology**
  - **French Arthurian Legend**
  - **German Arthurian Legend**
  - **Germanic Mythology**
  - **Greek Mythology**
  - **Hebrew Mythology**
  - **Hellenized Egyptian Mythology**
  - **Hellenized Near Eastern Mythology**
  - **Hellenized Persian Mythology**
  - **Hindu Mythology**
  - **Irish Mythology**
  - **Islamic Mythology**
  - **Judæo-Christian Legend**
  - **Latinized Egyptian Mythology**
  - **Latinized Germanic Mythology**
  - **Latinized Greek Mythology**
  - **Latinized Irish Mythology**
  - **Latinized Near Eastern Mythology**
  - **Mayan Mythology**
  - **Near Eastern Mythology**
  - **Norse Mythology**
  - **Northwest Semitic Mythology**
  - **Orphic Mythology**
  - **Persian Mythology**

  - **Proto-Germanic Mythology**
  - **Roman Mythology**
  - **Russian Mythology**
  - **Scottish Arthurian Legend**
  - **Slavic Mythology**
  - **Spanish Arthurian Legend**
  - **Welsh Arthurian Legend**
  - **Welsh Mythology**
- **Roman cognomen** = (plural cognominia) were originally nicknames that were later utilized to augment family names to identify a particular branch within a family or family within a clan
- **Roman gens** = (plural gentes) identified a family consisting of all those individuals who shared the same nomen and claimed descent from a common ancestor
- **Roman nomen** = (plural nominia) were hereditary surnames that identified a person as a member of a distinct gens
- **Roman praenomen** = (plural praenomina) are early personal names chosen by the parents of a Roman child originally bestowed the eighth day after the birth of a girl, or the ninth day after the birth of a boy; the praenomen would then be formally conferred a second time when girls married, or when boys reaching manhood and assumed the toga virilis (which in the case of Romans boys was about age 14 or 15)
- **Surname** = Name is also a surname in the *pdSurname* companion software product

USING THE RELATIONSHIP FILE

***pdNickname* has two data sets, a names database listing all included given names and nicknames, and a separate file with name-pair relationships. This section discusses the relationship file.**

The relationship file is made up of name pairs. The name pairs show:

- Nicknames for given names—short forms and diminutives
- Nickname variations
- Onomatologically related given names—onomastic variants are either close, near, or distant
- Phonetic matches—names that sound or are spelled similarly
- *Pro* edition only: fuzzy logic matches—misspelled names

EXAMPLES

Here are twenty examples of related names:

	Gender #1	Name #1	Gender #2	Name #2	Relation
Example 1	Female	BEATRICE	Female	BEA	Short form nickname
Example 2	Male	GABRIEL	Male	GABE	Short form nickname
Example 3	Male	PHILLIP	Male	PHILL	Short form nickname
Example 4	Female	REBECCA	Female	BECCA	Short form nickname
Example 5	Male	ANTHONY	Male	TONY	Diminutive nickname
Example 6	Male	IGNACIO	Male	NACHO	Diminutive nickname
Example 7	Female	NICHOLLE	Female	NIKKI	Diminutive nickname
Example 8	Female	OKSANA	Female	OKSANOCHKA	Diminutive nickname
Example 9	Male	ALAN	Male	ALLEN	Close onomastic variation
Example 10	Female	DELORES	Female	DELORIS	Close onomastic variation
Example 11	Female	MONIQUE	Female	MONIQUA	Near onomastic variation
Example 12	Male	MATTHEW	Male	MATTHIEU	Near onomastic variation
Example 13	Male	BENJAMIN	Male	BINYAMIN	Distant onomastic variation
Example 14	Female	WILLAMINA	Female	WILHELMINA	Distant onomastic variation
Example 15	Female	JULIA	Female	JOLEAH	Phonetic match
Example 16	Male	TERENCE	Male	TORENCE	Phonetic match
Example 17	Female	FRANCISCA	Female	FANCISCA	Fuzzy logic match
Example 18	Male	OLIVIER	Male	OLIVEIR	Fuzzy logic match
Example 19	Male	DANIEL	Female	DANIELLA	Opposite gender match
Example 20	Male	RASHEED	Female	RASHEEDA	Opposite gender match

All given names and nicknames in the relationship file are drawn from the standardized spellings in the names database. Standardizing requires all spaces, periods, hyphens, and apostrophizes be removed.

The relationship file does not have a special primary key, but the combined gender and name fields are unique and all the names and genders except fuzzy spelling are in the names database. If users have a special purpose, they may need to create a primary key field. Normally individual records are not edited or deleted in the file.

SECTIONS IN THE RELATIONSHIP FILE

The relationship file is divided into 12 (*Standard* edition) or 18 (*Pro* edition) smaller sections for user convenience and to allow users more options when setting up their system. All sections may not be needed for a particular project, and the divisions make it easy to build a custom database from selected sections. All sections have the exact same structure and can be stacked on top of each other and easily sorted.

Section of the relationship file and their locations are:

Folder	File Name	Description
Relationship File	pdNickname_USA_Relationship.*	United States given name variations, nicknames, and phonetic matches
	pdNickname_International_Relationship.*	International given name variations, nicknames, and phonetic matches
	pdNickname_SpecialCharacter_Relationship.*	Given name variations, nicknames, and phonetic matches with extended ANSI characters**
Relationship File\ Reverse	pdNickname_USA_Relationship_Rev.*	United States given name variations, nicknames, and phonetic matches in reverse order
	pdNickname_International_Relationship_Rev.*	International given name variations, nicknames, and phonetic matches in reverse order
	pdNickname_SpecialCharacter_Relationship_Rev.*	Given name variations, nicknames, and phonetic matches with extended ANSI characters in reverse order**
Opposite Gender	pdNickname_USA_OppositeGender.*	United States opposite gender matches
	pdNickname_International_OppositeGender.*	International opposite gender matches
	pdNickname_SpecialCharacter_OppositeGender.*	Opposite gender matches with extended ANSI characters**
Opposite Gender\ Reverse	pdNickname_USA_OppositeGender_Rev.*	United States opposite gender matches in reverse order
	pdNickname_International_OppositeGender_Rev.*	International opposite gender matches in reverse order
	pdNickname_SpecialCharacter_OppositeGender_Rev.*	Opposite gender matches with extended ANSI characters in reverse order**
PRO EDITION ONLY Fuzzy Logic	pdNickname_USA_FuzzyLogic.*	United States fuzzy logic matches
	pdNickname_International_FuzzyLogic.*	International fuzzy logic matches
	pdNickname_SpecialCharacter_FuzzyLogic.*	Fuzzy logic matches with extended ANSI characters**
PRO EDITION ONLY Fuzzy Logic\ Reverse	pdNickname_USA_FuzzyLogic_Rev.*	United States fuzzy logic matches in reverse order
	pdNickname_International_FuzzyLogic_Rev.*	International fuzzy logic matches in reverse order
	pdNickname_SpecialCharacter_FuzzyLogic_Rev.*	Fuzzy logic matches with extended ANSI characters in reverse order**

* All files are provided in comma separated values (.CSV), fixed length (.TXT), and standard database (.DBF) formats.

** Includes name pairs where at least one name has an extended ANSI character.

UNITED STATES NAMES

Given names and nicknames found in the United States are grouped in sections with "**_USA_**" in the file name.

INTERNATIONAL NAMES

Given names and nicknames found outside the United States are grouped in sections with "**_International_**" in the file name.

NAMES WITH EXTENDED ANSI CHARACTERS

Given names and nicknames with extended ANSI characters are grouped in sections with "**_SpecialCharacter_**" in the file name.

OPPOSITE GENDER NAMES

Given names pairs of opposite gender are grouped in sections with "**_OppositeGender_**" in the file name.

FUZZY LOGIC NAMES (PRO ONLY)

Given names and nicknames matched to fuzzy logic spellings are grouped in sections with "**_FuzzyLogic_**" in the file name.

NAME AND GENDER FIELDS

FIELDS

GENDER1 | Male or female gender flag for Name #1
M = Male
F = Female
NAME1 | Name #1
GENDER2 | Male or female gender flag for Name #2
M = Male
F = Female
NAME2 | Name #2
Each record has a pair of related names in fields that can be up to 35 alphabetic characters each and a pair of corresponding fields with one-character alphabetic codes that indicate the gender associated with the names.

Each record has two sets of name information, a GENDER1+NAME1 side and a GRNDER2+NAME2 side. These fields contain given names and nicknames, along with their corresponding genders, which are related either onomatologically, phonetically or, in the *Pro* edition only, can be fuzzy logic matches.

Users can match this name information with records in their lists to establish if two or more records on their lists are the same person with the first name entered differently, such as one record showing a formal given name and another record a nickname, or one record presenting one variation of a name and another record a different variation.

RELATIONSHIP FLAG

FIELDS

REL | Relationship flag

Each record has a one-character alphabetic code that indicates the relationship between name pairs:

1 = Close onomastic variant
2 = Near onomastic variant
3 = Distant onomastic variant
S = Short form nickname
D = Diminutive nickname
P = Phonetic match
X = Opposite gender match
F = Fuzzy logic match (*Pro* edition only)

The relationship between name pairs can be important to how users filter results. This information is entered in the REL field. The relationship may be as a short form nickname (REL = "S"); or diminutive nickname (REL = "D"); or a close (REL = "1"), near (REL = "2"), or distant (REL = "3") onomastic variation; or a phonetic match (REL = "P"); or an opposite gender match (REL = "X"); or, in the *Pro* edition only, a fuzzy logic match (REL = "F").

Some names serve both as a given name and as a nickname, such as the name "Kate", which is a nickname for "Katherine" and "Katarina", and is now also considered a proper given name. Because of this, some name pairs could be flagged both as an onomastic variation and as a nickname. In these cases the nickname designation supersedes the variation designation, and the record is flagged as either a short form nickname or a diminutive nickname. Users can link back to the names database to determine if a name pair is an onomastic variation as well. This will be true when "G" is entered for the nickname in the names database GIVEN field.

Definitions of the name relationships are:

ONOMASTIC VARIATIONS

These are related formal given names, either in the same language or translated into another language. The onomastic distance between variants is rated on a 1 (closest) to 3 scale. This value is determined by tabulating or estimating the number of lines separating the names on a name tree.

SHORT FORM NICKNAMES

These are nicknames designed to show familiarity with a person that are shorten or abbreviated forms of the associated name. For example, "Matt" is a short form nickname for the given name "Matthew". Notice that the short form is simply the given name abbreviated to first four letters. Short form nicknames are more often taken from the first syllable of the associated name, but can be taken from any part of the name. For example, the short form "Belle" for the given name "Isabelle" is taken from the end of the name.

DIMINUTIVE NICKNAMES

These are nicknames designed to show endearment and sometimes intimacy with a person that are usually, but not always, based on the root of the associated name, and frequently, but not always, conclude with a diminutive suffix such as “Y” or “IE”. For example, “Kathy” and “Kathie” are both diminutive nickname for both the given names “Katharine” and “Kathleen”. Notice that the diminutive is taken from the root of the given names, but adds the suffixes “Y” and “IE”, respectively. Diminutive nicknames are more often taken from the first syllable of the associated name, but can be taken from any part of the name, or can be altogether different. For example, the diminutive “Sandy” for the given names “Alexander” and “Alexandra” is based on the second half of the name. While the diminutive “Diesel” for the given name “Matthias” has no phonetic association with the name at all, but rather derives from a German nickname.

PHONETIC MATCHES

These are first names that are spelled or pronounced similarly, such as “Garry” and “Gerry” or “Lana” and “Lona”. The system is designed to pick out similar names that are not onomatologically related, but it also matches many names that are not listed as related names in onomastic documentation, often due to the rarity of the spelling, but are doubtlessly derived from the same name formation. Many thousands of unlisted variations are picked up with the phonetic algorithms.

OPPOSITE GENDER MATCHES

These are given names that are related to given names of the opposite gender. For example, the given names “Davida” and “Davina” are feminine forms of the male given name “David”. All other relationship files have the same gender in both the GENDER1 and GENDER2 fields. The opposite gender file is the only one that does not. While opposite gender relationships are typically not utilized in name matching, they can be useful in some circumstances. If users already have gender in their list, they can match against opposite gender names to determine if a gender was entered incorrectly or the wrong formation of the name was used (female and male names are often quite similarly spelled).

FUZZY LOGIC MATCHES (PRO ONLY)

These are first names that are matched to spellings of the same name written with typographical errors. This is the only file that includes names that are not real spellings. For example, “Abgel” and “Abigell” are not real names, but rather misspellings of the real given name “Abigail”. And “Stanlley” is not a real name, but rather a misspelling of the real given name “Stanley” with the “L” accidently doubled.

SCORE

FIELDS

SCORE | Match quality score
Record have a two-character numeric code that indicates on a 01 (best) to 99 scale the quality of each name-pair match. The score of "99" is reserved for archaic matches, and "00" means the match is not scored (which only pertains to fuzzy logic and opposite gender pairs).

The overall quality of each name-pair match is quantified on a scale of 01 (best) to 99. The number of matches from a query can sometimes be very numerous, and the score is effective in ordering the output for filtering. Users will find this a major advantage with our system. The scoring considers several factors:

- How closely the names are onomatologically linked
- If the match is a nickname or given name variant—nicknames are generally scored higher, but not always
- If a nickname match is a short form or diminutive—short forms are generally scored higher, but not always, such as when a diminutive is known to be very popular
- If a nickname matches the beginning syllable of an associated name or another part of the name—matches to the beginning are generally scored higher, but not always, such as when a nickname matched to another part is known to be very popular
- How closely the languages match
- How closely the names are spelled and pronounced
- The popularity of the names involved in the match

Note that some archaic matches are included for their onomastic significance. The score of "99" is reserved for these and only matches.

Also note that opposite gender matches and fuzzy logic matches (*Pro* edition only) are not scored, and instead are flagged with a "00" in the SCORE field. This is because not enough of the criteria necessary for scoring are present for these matches.

OPEN SOURCE PHONETIC ALGORITHMS

FIELDS

DMP | Double Metaphone | *P = Primary line match; S = Secondary line match*
MP | Metaphone | *M = match*
NY | New York State Identification and Intelligence System (NYSIIS) | *N = match*
CV | Caverphone | *C = match*
SX | Soundex | *S = match*
DMSX | Daitch–Mokotoff Soundex | *P = Primary line match; S = Secondary line match*
Each record has up to six one-character alphabetic codes that indicates if a particular open source phonetic match was achieved. Flags are indicated above.

As part of our phonetic indexing process we include matches from six open source algorithms most data engineers are familiar with. These matches are flagged in a series of fields. Not all open-source matches are included because many are junk matches.

Nicknames matches, opposite gender matches, and fuzzy logic matches (*Pro* edition only) are not flagged in these fields because these types of matches are not generally conducive to phonetic algorithms.

The open source phonetic algorithms utilized are:

SOUNDEX

This is the original phonetic algorithm. It was developed by Robert C. Russell and Margaret King Odell and patented in 1918 and 1922. The process was the first to index names by sound, as pronounced in English. The algorithm mainly encodes consonants. A vowel is not encoded unless it is the first letter.

METAPHONE

This is considered the first advanced phonetic algorithm. It was published in 1990 by Lawrence Philips and improved on Soundex by using information about variations and inconsistencies in English spelling and pronunciation to produce more accurate coding.

DOUBLE METAPHONE

This algorithm, also published by Lawrence Philips, is called “Double” because it can return both a primary and a secondary code for a name string. The algorithm takes into account spelling peculiarities of a number of languages in addition to English.

NEW YORK STATE IDENTIFICATION AND INTELLIGENCE SYSTEM (NYSIIS)

This algorithm was developed in 1970 and is similar to Soundex except it maintains relative vowel positioning and handles some phonemes and sequential letters better. The accuracy increase over Soundex has been cited as 2.7 percent.

CAVERPHONE

This algorithm was first developed by David Hood in the Caversham Project at the University of Otago in New Zealand in 2002 and revised in 2004. It was created to assist in data matching between late 19th century and early 20th century New Zealand electoral rolls.

DAITCH–MOKOTOFF SOUNDEX

This algorithm was developed in 1985 by Jewish genealogists Gary Mokotoff and Randy Daitch. It is a refinement of Soundex algorithms designed to allow greater accuracy in matching of Eastern European and Ashkenazi Jewish names with similar pronunciation but differences in spelling. While specifically developed for matching surnames, it is often useful for matching first names and other words as well.

REVERSE RECORDS

FIELDS

DIR | Name pair direction flag
Each record has one-character alphabetic codes that indicates if the names in the name pair are entered in the standard direction or reversed.
A = Name pair is in standard direction
R = Name pair is in reverse direction

Which name in the relationship file is in the NAME1 field and which is in the NAME2 field is not randomly chosen, but rather specifically entered depending on the type of information provided in the record.

All relationship file records are provided with the NAME1 and NAME2 information entered in a standard direction as well as separately in the reverse direction. Everything else about the record remains the same, only the names are reversed.

All reversed records are in folders named "**\Reverse**" and have "**_Rev**" at the end of the file name just preceding the file extension. For example, the file "Relationship File\pdNickname_USA_Relationship.csv" is entered in standard direction and the file "Relationship File\Reverse\pdNickname_USA_Relationship_Rev.csv" is the same file entered in the reverse direction.

STANDARD DIRECTION

The following describes the standard direction for entering names in name pairs:

- **Given name variants**: the NAME1 and NAME2 information is in alphabetical order
- **Nicknames**: the name receiving the nickname is in NAME1 and the nickname itself is in NAME2
- **Phonetic Matches**: the NAME1 and NAME2 information is in alphabetical order
- **Opposite Gender Matches**: the masculine form is in NAME1 and the feminine form is in NAME2
- **Fuzzy Logic Matches**: the real name is in NAME1 and the fuzzy name is in NAME2

REVERSE DIRECTION

The following describes the reverse direction for entering names in name pairs:

- **Given name variants**: the NAME1 and NAME2 information is in reverse alphabetical order
- **Nicknames**: the nickname itself is in NAME1 and the name receiving the nickname is in NAME2
- **Phonetic Matches**: the NAME1 and NAME2 information is in reverse alphabetical order
- **Opposite Gender Matches**: the feminine form is in NAME1 and the masculine form is in NAME2
- **Fuzzy Logic Matches**: the fuzzy name is in NAME1 and the real name is in NAME2

FUZZY LOGIC (PRO ONLY)

This section applies to pdNickname Pro only.

If you typed "**Garfeild**" into a word processor, it would probably be underlined with a squiggly red line signifying a misspelling. It is the name "**Garfield**" with the "**IE**" reversed to "**EI**"—a common mistake.

The fuzzy logic technology in the *Pro* edition of this software allows matching name data that has typographical errors. If users look at the fuzzy logic records, they are likely to see errors they have repeatedly made or seen. In many cases you will have to look close to see the difference, but they are different. There are almost 10 million fuzzy logic records.

The fuzzy logic file uses the same format as the other relationship files. The fuzzy logic portion is broken into six smaller sections for user convenience and to allow users more options when setting up their system. All sections may not be needed for a particular project, and the divisions make it easy to build a custom database from selected sections. All sections have the exact same structure and can be stacked on top of each other and easily sorted.

Section of the fuzzy logic file and their locations are:

Folder	File Name	Description
PRO EDITION ONLY Fuzzy Logic	pdNickname_USA_FuzzyLogic.*	United States fuzzy logic matches
	pdNickname_International_FuzzyLogic.*	International fuzzy logic matches
	pdNickname_SpecialCharacter_FuzzyLogic.*	Fuzzy logic matches with extended ANSI characters**
PRO EDITION ONLY Fuzzy Logic\ Reverse	pdNickname_USA_FuzzyLogic_Rev.*	United States fuzzy logic matches in reverse order
	pdNickname_International_FuzzyLogic_Rev.*	International fuzzy logic matches in reverse order
	pdNickname_SpecialCharacter_FuzzyLogic_Rev.*	Fuzzy logic matches with extended ANSI characters in reverse order**

* All files are provided in comma separated values (.CSV), fixed length (.TXT), and standard database (.DBF) formats.

** Includes name pairs where at least one name has an extended ANSI character.

UNITED STATES NAMES

Given names and nicknames found in the United States are grouped in sections with "**_USA_**" in the file name.

INTERNATIONAL NAMES

Given names and nicknames found outside the United States are grouped in sections with "**_International_**" in the file name.

NAMES WITH EXTENDED ANSI CHARACTERS

Given names and nicknames with extended ANSI characters are grouped in sections with "**_SpecialCharacter_**" in the file name.

USING THE FUZZY LOGIC FILES

The fuzzy logic file is utilized exactly like the other relationship files, except fuzzy logic deals with misspelled names instead of variations and nicknames. It attempts to duplicate real errors created while entering names into databases. The most likely typographical errors are determined based on the number of letters, the characters involved, where they are located in the name, the language, and other factors. None of the fuzzy spellings formulate a real name already in the database.

The biggest advantage in our technology is in its ability to work with language rules that indicate how individuals of various nationalities may hear and spell names.

A score of "00" is entered for all fuzzy logic matches and phonetic algorithms are not run against them because we already known they are the same exact name, with one name misspelled. All fuzzy matches have a relationship of "F".

Some fuzzy logic spellings have one typographical error while others have multiple issues, so the technology is suited for even the worst typists and transcribers. The algorithms have five layers:

PHONETIC MISSPELLINGS

These algorithms look at digraphs, trigraphs, tetragraphs, pentagraphs, hexagraphs, and even a German heptagraph, "SCHTSCH", used to translate Russian words with the "SHCHA" or "SHCH" (romanticized) sound. These are, respectively, two to seven letter sequences that form one phoneme or distinct sound. Most of letter sequences trigraph and above are Irish who have more language rules than you can shake a stick at.

Many misspellings occur as transcribers enter the sounds they hear. The character sequences and the sounds they produce are different for each language and situation, such as before, after, or between certain vowels and consonants, so our substitutions are language-rule based. Furthermore, our algorithms consider both how a name may sound to someone who speaks English as well as how it may sound to someone who speaks Spanish, which is often different. Take the digraph "SC". Before the vowels "E" or "I" it is most likely to be misspelled by an English speaker as "SHE" or "SHI" while a Spanish speaker may hear "CHE" or "CHI" and sometimes "YE" or "YI". Our library includes over 80,000 language-based letter sequence phonetic rules. Phonetic misspelling examples:

	Real name	**Fuzzy name**	**Gender**
Example 1	BARTHOLOMEW	BARTHOLOMUE	Male
Example 2	DAWNETTE	DAUNETTE	Female
Example 3	NATHANIEL	NATHANAIL	Male
Example 4	PHYLLIS	FYLLIS	Female
Example 5	SIGOURNEY	SIGOURNI	Female
Example 6	XAVIER	XAVAR	Male

REVERSED DIGRAPHS

These algorithms look for misspellings due to reversed digraphs (two letter sequences that form one phoneme or distinct sound) which are a common typographical issue, such as “IE” substituted with “EI”. The character sequences and the sounds they produce are different for each language and situation, such as before, after, or between certain vowels and consonants, so our substitutions are language-rule based. Reversed digraph examples:

	Real name	**Fuzzy name**	**Gender**
Example 7	ANNABETH	ANNABEHT	Female
Example 8	CAETLIN	CEATLIN	Female
Example 9	EUGENE	UEGENE	Male
Example 10	FRIEDRICH	FREIDRICH	Male
Example 11	RAQUEL	RAUQEL	Female
Example 12	VICKTOR	VIKCTOR	Male

DOUBLE-LETTER MISSPELLINGS

These algorithms look for misspellings due to double letters typed as single letters and single letters that are doubled. The most common typographical issues occur with the characters, in order of frequency, “SS”, “EE”, “TT”, “FF”, “LL”, “MM”, and “OO”. Double-letter misspelling examples:

	Real name	**Fuzzy name**	**Gender**
Example 13	EMANNUEL	EMMANNUEL	Male
Example 14	KASSANDREA	KASANDREA	Female

MISSED LETTERS

These algorithms look for missed keystrokes and provide fuzzy logic matches with missing letters. Unlike the other algorithms, these are not language specific. Keystrokes can be missed in any language. Missed letter examples:

	Real name	**Fuzzy name**	**Gender**
Example 15	ABDUL	ADUL	Male
Example 16	MARGARET	MRGARET	Female

STRING MANIPULATIONS

These algorithm changes letters and syllables in a variety of ways. They are less guided by language rules and more guided by randomness. String manipulation examples:

	Real name	**Fuzzy name**	**Gender**
Example 17	CYNTHIA	CYNTTHA	Female
Example 18	GERALD	GERLLD	Male

COMPATIBILITY

To ensure compatibility with any operating system and database platform, **pdNickname** is provided in multiple file formats and utilizes only the ANSI character set (ASCII values 0 to 127 and extended values 128 to 255).

USING PDNICKNAME WITH PDSURNAME AND PDGENDER

pdNickname, pdSurname, and *pdGender* make excellent partners. They have been developed to be fully compatible. The name pair format in *pdNickname* is very similar to the *pdSurname* database except *pdNickname* is used to match give names and nicknames while *pdSurname* matches last names. *pdGender* is based on the first name database and is designed to apply gender identification to first name records. Note that *pdSurname* and *pdGender* are not required to use *pdNickname* but they are highly attuned to work together.

USER GUIDE UPDATES

User guides are updated based on information gained from user experience. It is suggested that users regularly check the Support section of the Peacock Data website for updates. Look for a date newer than the date below:

The publication date of this guide is: May 10, 2016.

DATABASE VERSION NUMBER

Depending on the file format, the version number of each copy of *pdNickname* is written in the first or second row of the first or second column of all database files in **X.X.X** format. The first number is the main version number of the release. The number after the first dot is the update for the version indicated. The number after the second dot references a minor revision.

SITE LICENSE

Peacock Data's site licenses are designed to be fair. They are broader than most software licenses in that they allow installation on not one but all computers in the same building within a single company or organization. We ask users to honor these simple rules so Peacock Data can continue bringing great products to users.

THE USE OF *PDNICKNAME* IS GOVERNED BY THE FOLLOWING SITE LICENSE

I. This Site License grants to the Licensee the right to install the licensed version of **pdNickname** (hereinafter, 'information') on all computers in the same building within a single company or organization. Separate Site Licenses must be purchased for each building the information is used in.

II. The information may only be used by the employees of the Licensee. If the Licensee is an educational institution, the data may only be used by enrolled students, faculty, teaching assistants, and administrators.

III. Temporary employees, contractors, and consultants of the Licensee who work on-site at the Licensee's facility may also use the information in connection with the operation of the business of the Licensee. Any copies of the information used by temporary employees, contractors, and consultants must be removed from such individual's computers once they cease working at the Licensee's facility.

IV. The information cannot be used to provide services or products to customers or other third parties, whether for-profit or given away. A Developer License must be purchased separately by the Licensee to incorporate the information in for-profit services and products.

V. The Licensee is required to use commercially reasonable efforts to protect the information and restrict network or any other access to the information by anyone inside or outside of the Licensee's facility who is not authorized to use the information.

VI. The Licensee owns the media on which the information is recorded or fixed, but the Licensee acknowledges that Peacock Data, Inc. and its licensors retain ownership of the information itself.

VII. The Licensee may not transfer or assign its rights under this license to another party without Peacock Data, Inc.'s prior written consent.

VIII. Peacock Data, Inc. may revoke the rights granted by this license upon a violation of any provision herein by the Licensee.

IX. This Site License is governed by Peacock Data, Inc.'s Terms of Service and Privacy Policy, and the laws and regulations of the United States and the State of California.

COPYRIGHT NOTICE