Informatica® SSA-NAME3
10.5

Introduction to SSA-NAME3 User Guide

Informatica SSA-NAME3 Introduction to SSA-NAME3 User Guide
10.5
September 2022

Table of Contents

Preface

Welcome to the Informatica SSA-NAME3 Introduction Guide.

This guide is intended for users such as a business manager, system designer or DBA. This guide contains a specific section designed to get application programmers familiar with the concepts of developing an SSA-NAME3 search and match application.

Learning About Informatica SSA-NAME3

This section provides details of documentation available with the SSA-NAME3 product.

Introduction to SSA-NAME3

Provides an overview of SSA-NAME3. It is written in a way that can be read by someone who has no prior experience of the product and wants a general overview of SSA-NAME3. It explains the problems SSA-NAME3 overcomes and provides an overview of how this is done. One chapter is dedicated to providing an overview for Application Programmers.

Getting Started

This manual is intended to be the first technical material a new developer or designer reads before installing or using the SSA-NAME3 software, regardless of the platform or environment. Its goal is to help a new user get the software installed and produce a working prototype application that calls SSA-NAME3 and executes searches against their own data.

To achieve this it provides a "script" to follow which includes pointers to pertinent sections of the other manuals.

Application & Database Design

This manual contains tips and techniques useful for setting up and optimizing a name search and matching application, including database issues, and illustrates best-practice techniques, common pitfalls, and strategies regarding the subject of name and address matching.

Installation Guide

This manual provides information on how to install the SSA-NAME3 product.

SSA-NAME3 Workbench User Guide

This is a guide to using the SSA-NAME3 Workbench - a Java GUI tool that helps a programmer understand and prototype SSA-NAME3 calls. The Workbench is also used for:

- Generating Sample Program Code;
- Executing SSA-NAME3 Calls;
- Testing different SSA-NAME3 run-time options;
- Producing debugging and support information for Informatica Corporation

Note: The Workbench in itself is not a search and match application. It assists the developer build a search and match application.

API Reference

The ultimate goal of an SSA-NAME3 implementation is for application programs to be able to call SSA-NAME3's API Functions to build keys and search strategies and to compute match scores and decisions.

This manual describes a typical program process flow for building an identity search application, and also lists in detail each of the API Functions. It describes the parameters required by these functions and the information returned.

Population Override Manager User's Guide

This is a guide to using the SSA-NAME3 Population Override Manager - a Java GUI tool that allows a trained data analyst to override some of the Standard Population rules that are supplied with the product, or provided in the form of a Custom Population. The types of rules that can be overridden using this tool are:

- Edit-list rules
- Frequency tables
- Scalar Frequency Tables
- Matching Purposes

Note: Use of this tool without proper training from Informatica should not be attempted, as improper use can adversely affect the reliability and performance of the search application(s).

Edit Rule Wizard User's Guide

This is a guide to using the SSA-NAME3 Edit Rule Wizard - a Java GUI tool that helps a business user safely add certain types of Edit Rules to the Standard or Custom Population without requiring specific knowledge of SSA-NAME3 or support from a programmer or data analyst. The types of rules that can be added using this tool are:

- Discard a word or phrase when searching and matching (e.g. a new "noise" word)
- Add a new replacement word or phrase when searching and matching (e.g. a new "abbreviation", "nickname" or "acronym")
- Add a new compound name marker word

Release Notes

The Release Notes contain information about what's new in this version of SSA-NAME3. It is also used to summarize any documentation updates as they are published.

What Do I Read If. . .

I am. . .

. . . a business manager

The INTRODUCTION TO SSA-NAME3 will address questions such as "Why have we got SSA-NAME3?", "What does SSA-NAME3 do"?

I am. . .

. . . a system designer or DBA

The INTRODUCTION TO SSA-NAME3 will address questions such as "What resources are needed to implement SSA-NAME3?". The APPLICATION & DATABASE DESIGN manual will lead you through many of the design considerations of name search and matching applications.

I am. . .

. . . installing SSA-NAME3

Before attempting to install SSA-NAME3 you should read the Getting Started document. This will describe the pre-requisites and help you plan the installation and implementation of SSA-NAME3. The actual installation steps for your platform are documented in the Installation Guide.

I am. . .

. . . an Analyst or Application Programmer

A high-level overview is provided specifically for Application Programmers in the INTRODUCTION TO SSA-NAME3 manual. Before attempting to develop programs that interface with SSA-NAME3, you should also read the GETTING STARTED and APPLICATION & DATABASE DESIGN manuals, as well as experimenting with calls in the WORKBENCH USER GUIDE.

When developing the application program(s), use the API REFERENCE manual which describes a typical application and the Function parameters.

Working example programs that illustrate the calls to SSA-NAME3 in various languages are available by using the Sample Program button on the Workbench.

I want to know. . .

. . . what SSA-NAME3 does

The INTRODUCTION TO SSA-NAME3 manual gives an overview of what SSA-NAME3 does and how it does it.

I want to know. . .

. . . how to setup the database

Refer to the APPLICATION & DATABASE DESIGN manual for tips and techniques on configuring the database to store SSA-NAME3 Keys and optimizing it for searching and matching.

I want to know. . .

   . . . how to code a search application

   The INTRODUCTION TO SSA-NAME3 manual contains a specific section designed to get application programmers familiar with the concepts of developing an SSA-NAME3 search and match application.

   The API REFERENCE GUIDE details the Function calls required and their parameters. The SSA-NAME3 WORKBENCH USER GUIDE shows how to generate a sample program in a variety of programming languages.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at http://velocity.informatica.com. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through the Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
https://www.informatica.com/services-and-training/customer-success-services/contact-us.html.

To find online support resources on the Informatica Network, visit https://network.informatica.com and select the eSupport option.

CHAPTER 1

Introduction

This chapter includes the following topics:

Differing Needs for Name Search and Matching

The nature of the data used in name search and matching applications varies considerably. The name, address and identification data used in a customer information system is very different from that in a hospital patient identification system, which in turn is very different from the data used in a police persons of interest or criminal records system.

The quality of the identification data in a customer information system will be much better than the quality in a mailing file purchased to use in a marketing campaign. The data in criminal records will be of very different quality than data in an incident reporting system.

The nature of the business applications requiring name search and matching vary considerably as does the relative importance of "name search" or "name matching" between different business applications and types of users.

The need to quickly find a customer's account number given his or her name, is quite different from the need to prove that a prospective customer is not already on file. The need to ensure that a prospective customer is not involved in "known fraud", or that a person is not on a police "warrants" or "criminal record" file requires an exhaustive search.

Some online applications can be designed to let end users make matching choices from displays, while other critical or high volume online applications and batch applications can be designed with fully automated matching without human intervention.

The SSA-NAME3 software will enable an organization to use a standard set of tools, designs and strategies to satisfy all varying name search and matching needs of the organization.

Types of Data Supported by SSA-NAME3

SSA-NAME3 supports applications that search databases using the following types of data:

Data type	Description
person names	Formatted or unformatted, with or without titles and other noise, including nicknames and names with multiple entities, in any language. Formatted names include full names in title case as mentioned in official documents, and unformatted names include full names without title case as mentioned in unofficial documents or forms. For example, `Mr Michael Smith` is a formatted name, and `mr michael smith` is an unformatted name.
organization names	Standardized or unstandardized, with or without legal endings and other noise, including abbreviations and names with multiple entities, in any language.
addresses	Formatted or unformatted, standardized or in a raw form, typically using that part of an address up to but not including the locality information. Formatted addresses include complete addresses as mentioned in official documents, and unformatted addresses include addresses as mentioned in unofficial documents or forms. Formatted addresses follow the postal address format for the relevant country, and unformatted addresses do not follow the postal address format for the relevant country.
product names	Including embedded codes, sizes and part numbers.
titles	From files, books, songs, etc.
descriptions	In free text of up to 255 characters in length, including local jargon and abbreviations.

SSA-NAME3 supports applications that need to match data using one or more of the following data types:

Data type	Description
names, addresses and descriptions	Despite the error and variation and including multiple occurrences of such data fields.
identification codes	ID numbers, Social Security Numbers, Account codes, Driver's license numbers, etc., taking into account transpositions and missing digits.
dates	Which can be matched, despite errors, as equivalent, or within range or age parameters.
attributes	Phone numbers, gender, eye color, regions, etc.

SSA-NAME3 supports applications that need to match data for the following business "purposes" and more:

Data type	Description
same Name	Person or Organization names
same Address	Formatted or Unformatted Addresses

Data type	Description
same Individual	Name, ID number or date of birth, and other optional identification
same Resident	Name and address or telephone number. . .
same Household	Family name and address. . .
same Family	Family name and address or telephone number. . .
same Wide_Household	Address and family name or telephone number. . .
same Contact	Name, organization, address, telephone. . .
same Wide_Contact	Name, organization. . .
same Division	Organization name, address. . .
same Corporate_Entity	Organization name. . .

For a full list and description of the available purposes for each population, use the Workbench, select the appropriate population, then access the Population Documentation through the Help menu.

SSA-NAME3 also supports applications that need to filter records, prior to matching, based on the setting of a record flag or list of flags.

SSA-NAME3 supports search strategies on any database or access method so that:

- physical database access can be optimized
- logical database access can be simplified
- maximum quality possible can be obtained
- quality and performance can be balanced to the application needs

CHAPTER 2

The Variation Problems

There are many reasons why two reports covering the same individual person, organization, or identity end up with conflicting data stored in the system. Understanding the source of these variations will lead to an appreciation of the "search" problem.

Phonetic Variation

Where names are spoken, especially over a radio or telephone, a whole class of variations in the spelling can occur. This is usually referred to as a phonetic problem and the recognition of its existence lead to the use of historic algorithms such as SOUNDEX or those based simply on PHONETICS.

This phonetic variation is itself compounded by the fact that even when a name is spelled out by saying the letters, a degree of phonetic confusion can still occur.

The false presumption of many algorithms is that the phonetic problem is in its own right the major variation. There is in fact evidence to suggest that phonetics accounts for less than 25% of the variations.

Subconscious Correction

A common reason for variation has to do with automatic or subconscious "correction" of names that are "familiar" to the user or that are very similar to common names. For example SMITHE becomes SMITH; WILLIAM at the end of a name becomes WILLIAMS.

Physical Variation

A significant amount of error can occur when transcribing names from paper to paper or to a computer terminal. This error is often a mental one as in transposition or truncation (For example, Beth becomes Beht) or it can be keyboard dependent. For example, R instead of E on the QWERTY keyboard.

Another form of this error is the substitution of a graphically similar letter when reading handwriting. For example, G for Q, S for Z, or M for N.

Real World Variation

Many people have and use a nickname form of their first given name. In some cases, it is completely arbitrary as to when and how it is used. In other cases, the nickname will be used where less formality is required (for example, entering a contest) and the full first name where formality is expected. For example, applying for a driver's license.

A difficult problem to address is associated with the common use of multiple family names, or the addition of words over time. The familiar case arises with marriage and divorce. Another common one is associated with children of parents who retain separate family names.

Another problem arises from Anglicization (more generally localization into local language, style or dialect). In populations where people change countries it is common to adjust the pronunciation and then the spelling of a foreign name to fit into local conventions.

Truncation

Truncation of naming words occurs for a variety of reasons. Examples of truncation include formal abbreviation (Ltd, Inc, St, Rd), informal abbreviation (Mgmt, Intl, Svcs), and use of initials or acronyms. Another common reason for truncation can be traced to the design of a form, screen or database field that was not given enough space to hold all of the data.

Concatenation and Splitting of Words

The concatenation of words can be related to subconscious correction (for example, La Grande becomes Lagrande) or genuine transcription error where the space was not evident. Splitting of words can occur for similar reasons (for example, MacDonald becomes Mac Donald).

Sequence Variations

A large class of variation arrives from the fact that the protocol for saying, requesting or entering a name is not stable. Forms and screens are designed inconsistently. When providing information, we use a mixture of orders for family and given names, and often choose only to give some of the multiple words used to make up either the family name or given names.

In some cases words are left out, especially middle names. In other cases they are re-sequenced. In certain cases, the set of words used is a choice from one of two or three subsets of a group of name words.

Another common case encountered is where a person assumes they recognize a family name and therefore re-sequences the words. This situation arises frequently with names that can be either family or given names (for example, William Andrews or Joseph James), or because of the fact that a small percentage of the population is given quite unexpected first names like Adler or Brown or last names like Mister or Sister.

There are also several populations where the practice is to create multi- word family names out of both parents' family names. In certain European and South American communities this problem is exaggerated by the fact that different sequences are formally used by members of the same family when referring to the same individual, and further aggravated in some populations where it is common to also abbreviate last words in family names to an initial.

The Name Distribution Problem

It is common knowledge that there are a few family names that encompass large groups of each population. This is also true for common given names and for the many common words in addresses or product names.

It is not as obvious as to how extreme this distribution really is.

For example, in some English speaking populations, names such as SMITH or WILLIAMS may each account for in excess of 1% of the population. Thus on a 5,000,000 record file the group with that family name may exceed 50,000 entries.

What is not often realized is that, not only is the file distributed in such a skewed fashion, it is also usually true that the queries or searches will be similarly distributed. That is, in excess of 1% of the searches may include one of the common surnames. One can imagine how tempting it is to bias the design of a key to service these "common names".

Conversely the distribution has an enormous "tail" of very uncommon names each shared by only a few members of the population. Many of these names will be unusual, and because of this, it is generally true that they will attract a greater degree of error and variation. If the key design is biased towards achieving reliability out of such uncommon name searches, it may lead to performance problem for common name searches.

SSA-NAME3 specifically addresses common words, codes and tokens in a different manner than uncommon data in order to ensure the best possible characteristics at each end of the distribution.

The Online Response Time Problem

An important benchmark of a name search is the response time before the candidates or matches are presented to the user.

The "online response" performance of a name search algorithm is an important concept. An algorithm that analyses thousands of records and, after a long period, supplies a small group of candidates to a user is usually less acceptable than an algorithm that can rapidly supply the user with a few highly probable candidates, but takes quite a long time to display the low probability candidates.

An example of the response time dilemma can be seen in those algorithms that provide an exact match as a fast path to the file entries. While there is certainly a fast response to the exact records, many records that are just as significant from a business point of view are missed. For example, records with minor variations are not available unless the user chooses to widen the search. This exact match with its quick response often leads to the other relevant entries not being discovered, and consequentially this will introduce duplicates.

The important aspect to consider is that the algorithm used must not force users to either extreme. The algorithm should allow rapid access to records of a particular significance. Each search dialogue that is designed should be able to take any level or depth of search that is relevant to the problem and should not be constrained by the algorithm.

Most implementations of name search algorithms do not provide multiple levels or depths of search from a physical access point of view. Such algorithms provide one "group code", "name code" or "phonetic key" that is used to select or access a "bucket" of file entries. This whole group or bucket is then analyzed to choose what to show to the user. With large volumes these buckets or groups are themselves very large.

SSA-NAME3 allows the sets of records found to be subdivided based upon the concept of search sets, such that only the records of appropriate level are physically accessed. It also provides the choice of several predefined "search levels" for optimizing different common needs.

CHAPTER 3

SSA-NAME3

This chapter includes the following topics:

SSA-NAME3 Overview

SSA-NAME3 is a suite of software tools that enables application programs to search and match records in databases using people's names, organization names, addresses, identity numbers, dates and other identification data.

SSA-NAME3 enables high quality search and matching despite the unavoidable variation and error that normally occurs in name, address and identification data.

SSA-NAME3 also supports the implementation of critical searches that are designed, not only to overcome this natural error but also, to discover the intentional variation and error introduced by people who are attempting to defeat or fraudulently avoid the controls built into such systems.

A business application program simply "calls" the SSA-NAME3 run-time routine to invoke its services. The run-time routine is supplied as a DLL, Shared Library or Load Module, depending on the platform, and also as a server process. SSA-NAME3 can be used by applications in both "online" and "batch" environments on the popular hardware and software platforms. The SSA-NAME3 Keys can be stored in all popular databases.

SSA-NAME3 is packaged with over 50 Standard Population rule sets for different countries and languages. These rule-sets contain the rules that a typical user will require to deliver reliability and performance in their search and matching applications.

The richness of the Standard Population rule sets allows applications to be insulated from the complexity of dealing with the inevitable variation and error that will exist in name, address and identification data.

Recognizing that there are differing types of search applications (from customer lookup, to duplicate prevention and fraud investigation), the application can choose different types and levels for searching and different purposes, levels and thresholds for matching through the API. These tuning options are easily understood and implemented.

To address the need a user may have to add some custom rules, for example to cater for certain types of abbreviations used only in their data, two levels of GUI tools are provided:

- the Edit Rule Wizard, intended for the business user
- the Population Override Manager, intended for the data analyst trained in SSA-NAME3

For the user with unusual or special needs, Informatica provides a customization service that results in the generation of a Custom Population rule set. This can be plugged in and used just like a Standard Population set.

The API is standard for the different Population rule sets allowing the development of business systems that can be easily ported to multiple countries despite language and character set variation, as well as those business systems that must deal with the mixed international data from global operations.

The SSA-NAME3 software includes algorithms for building special Keys for the names and addresses stored in the user's database tables. These special keys would be stored in a new database table by the user's application and must be indexed to provide optimum retrieval performance.

The SSA-NAME3 key building algorithms are also used for building special Key Ranges, to be used by the user's search application(s) to navigate that index and retrieve candidate records for a given search name or address. Different Search Strategies are supported, each generating different types of Key Ranges. For example, two types of Search Strategy are the Narrow search and the Exhaustive search.
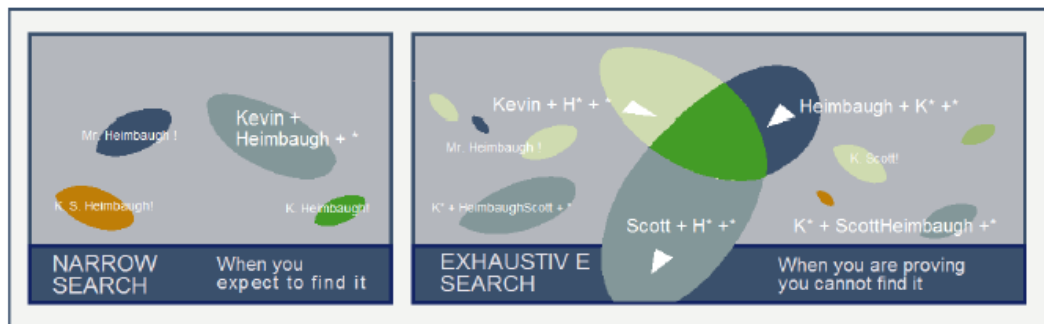
SSA-NAME3 includes different Match "Purposes" to allow the user's application to filter, match, rank or group the candidates returned from a search in a way that matches the business requirements of the system. These Match Purposes map different entity types including individuals, households, families, organizations, addresses, residents and contacts.
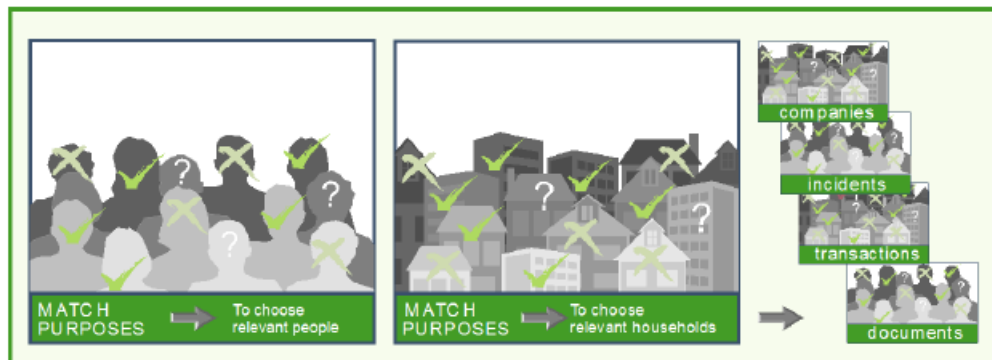
The SSA-NAME3 algorithms are available through a "call" interface or API.
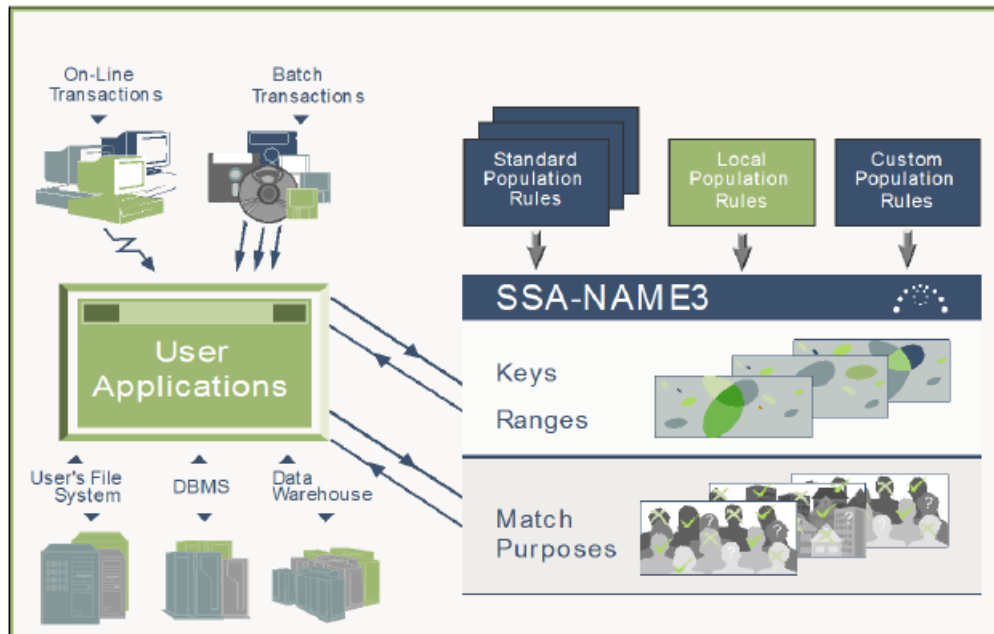
Examples of Search and Match Applications

The following are some examples of search and match applications using SSA-NAME3. The screens and reports shown below are not generated by SSA-NAME3, rather by the user developed program that calls SSA-NAME3:

- Example 1 shows an online person name search for the purpose of finding and matching the same "resident"
- Example 2 shows a batch address search for the purpose of matching the same "household"
- Example 3 shows a batch name search for the purpose of matching the same "individual"

Example 1

This first example shows the results from an online inquiry used to find records about the same person using the name as a search argument and both the name and address data to choose the match. This is an example of the Match Purpose "same Resident".

SEARCH	William Stuart Harison 117- 2a Jacksen Rd., Easthartford CT 06987
William Stewart Harison	117 Jackson Road., Hartford, CT 06987 Suite 2A
Doctor Bill Harisen jr	117 Jacson, Room 2a, Hartford East, CT 06984
Harrisen William Doctor	2a Jackson Rd, Hartford CT 06987-4573#174978
W. S. Harrison II PhD	Medical Center,117/2A Jackson Rd E. Hartford, NY16987 #17497

A similar online inquiry showing international data.

SEARCH	William Kwok Ki Ho Block C, 4th Floor, Unit 7, 234 Wan Chai Road, Hong Kong
William Kwok	4/F BlkC 234 Wan Chai Rd Hongkong
Kwok Ki Hoh	7/234 C-4, Wan Cha Rd, HK
Mr. K. H. Kwok	F7 234 Won Chai Rd (Block C 4th Flr) Hong Kong
Ki Ho Kok	Level 4 C Block 7/234 Wan Chai St, HK

Example 2

This example shows results from a batch process to discover members of the same "household" in records from multiple files, with data from many countries.

Co unt	ID	Name	Street	City	Area
1	DB01	JOHN ABITZ	1550/12 MLK BLVD	FOSTER	CA000000
	TX11	JOHN & SHEILA	ABITZ 1550 MLK BOULEVARD, STE12	FSTRCTY	CA30906
	TX11	ABITZ, SHEILA	1550 MLK BOULEVARD SUITE12		CA

Co unt	ID	Name	Street	City	Area
2	DB01	GONZALES GEORGE	CARGO CTRE, FRANKFURTAI RPORT		
	DB02	GONZALES GEORGE & ROBERT INC	LUFTRACHT ZENTRUM	FRANKFORTAM-MAIN	
	TX06	ROBERT GONZALES	CARGO CENRER, FLUGPATZ	FRANKFORT	
3	DB02	ANGIOLINI ALDA GABRIELLA	PUTNAM AVE, 1445 2F	OLD GREENWICH	CT06870
	DB05	GABRIELLA ALDA ANGIOLINI	1455 EAST PURNAM (2F - 1445)	OLD GRNWICH	CT6870
4	DB01	SMITH, DONALD DBA PRIMESTAR CONS CO	35 OLD KINGS RD, BISHOPSGATE	LONDON	EC3(UK)
	TX02	BRZEZINSKY,D OF PRIMESTAR CONS	BISHOPSGATE 35 OLDKINGS RD	LONDON	(UK)
	DB01	BRITTON, PAUL PRIMESTAR CONS CO	35 OLD KINGS RD	BISHOPSGATE LOND	EC3
5	DB01	PENNSYLVANIA DEPT.OF TRANSPOTATIO(LANCASTER CONTRCTG CORP.)	20 MAIN	HARRISBURG	PA23330
	TX03	PENN. STATE DEPT. OF TRANSP	20 MAIN (LANCASTER CORP)MAIN	HARRISBURG	PA23330
	TX22	LANCASTER CONTRACTING CORP (PENN DOT)	20 MAIN STREET	HARRISBURG	PA23330
6	TX02	Take 6 Arts Co-Operative	38 Queen Elizabeth Avenue	MARSGATE Kent	CT93LQ
	TX02	Take Six Arts Co Opritive	38 Wellis Gardens	Margate Kent	CT93LQ
7	DB01	MCCLISH SYLVIA	RURAL ROUTE 23 W ENCANTO	ALBEQUERQE	NM
	DB02	FUENTES MC LISH SILVIA	ENCANT RR 23	ALBUQUERQUE	NM

Example 3

This example shows results from a batch process to identify multiple records about the same person in a file containing names, dates of birth and identity numbers. This is also a demonstration of the "same Individual" Match Purpose.

Count	Social Security Number	Full Name	DOB
1	920994387	BROWN-WOOD CHRISTINA	
	920-94387	TINA BROWN	1958-NOV-17
	UNKNOWN	KRISTINA WOOD	581117
2	118505475	CORBETT AYANNA	1952-MAY-28
	000000000	CORBET ANN AY	
	118505475	AYANNA A CORBETT	520928
3	513254585	JAMES CHRIS	610513
	513425585	CHRISTOPHER JAMES	19610513
4	840-58-32	JOHNSON STEVIERAY	180107
	084058320	JANSON STEPHEN R	770118
5	386995162	CAPRIATI AUGUST V	690825
	386995612	AUGUSTUS VCA	690801
6	999999999	SCHMITT RONNIE	620118
	962545640	SCMIDT JR RONALD NOMIDDLENAME	620118
	962534644	SMITH RONNY	620909
7	068706902	ROBERTS LEE E	1944-JUN-24
	068706902	LEE ROBERT E	1945-JUN-24

Problems Addressed by SSA-NAME3

For the typical application SSA-NAME3 addresses:

- Errors made in spelling the spoken word.
- Transcription and keying errors for written names and codes.
- Missing words, initials, numbers or codes.
- Mixed usage of first names and initials.
- Mixed usage of 1, 2, etc. with. .. one, two . . . 1st, 2nd, .. First, Second, etc.

- Nicknames, formal and informal abbreviations, synonyms, language variation of common words.

- Concatenation or splitting of words and codes.

- Extra words and word sequence variations.

- Presence of irrelevant "noise words" in the data.

- Missing or "null" data.

- Presence of "foreign" names and addresses.

- Failures to find all parts of compound or account names where multiple entities are present in one name field.

- Anglicization (Localization) of names causing variation between formal name, as on a Passport or Driver's License, and less formal names on other transactions.

- The problems created by the frequent use of certain common last and first names, or use of common words and numbers in organization names and addresses.

- The fact that many names can be made from title or "noise" words. For example, Sister J Bishop, The Limited, The Company Inc.

For the higher volume or more sophisticated system, SSA-NAME3 also addresses the following issues:

- Length of response time of the system before an answer is available.

- The need to balance performance and quality.

- The problem that weaker name search approaches either miss relevant names, or conversely show far too many names for the user to be able to make relevant choices.

- That certain common nicknames and synonyms are better handled by "secondary searching". For example, Tina can be Chris but not Christopher, yet Chris can be Christopher.

- The design of dialogues so that neither the operator nor the system comes too quickly to the conclusion that there is not a relevant match. That is the volume can cause data to be missed.

- That increasing the width of the search to discover records with a larger amount of variation significantly aggravates the response time and performance.

- That progressive refinement of the system by addressing special cases introduces undiscovered problems elsewhere and progressively degrades the system.

- That mixing data from different systems in an integrated search leads to new error and frustration for users because of variations in the name handling between the source systems.

- That ad hoc changes to a name search system may improve specific transaction quality yet, because re-testing is not exhaustive, can cause previously satisfied cases to fail.

C H A P T E R   4

Major Concepts

This chapter includes the following topics:

Multiple Keys

The initial step in a system that needs to support searching on names or addresses is to build keys from the name or address data.

These keys must be able to overcome the error and variation in the data, including missing words, extra words and word sequence variations. To provide this level of reliability, multiple keys must be generated for each name or address.

Out-of-the-box, SSA-NAME3 provides three levels of keying. Standard keys are used by the typical user. Extended keys are used by the user with critical search needs. Limited keys are used by the user who is concerned with disk space and is willing to trade some reliability for savings in space.

The keys must also support efficient access. The Algorithms used to build SSA-NAME3 Keys are designed to provide efficient access, even for common names.

In addition, to get optimal performance it is necessary for the user to store these keys in a new table that contains not only the SSA-NAME3 Keys and foreign key to the source record, but also any additional identity data that will be used for matching, filtering or display purposes. This is so that the search application does not need to perform table joins at search time to get the data necessary for matching and display.

The SSA-NAME3 Key column must be indexed.

Search Strategies

The first step in an online or batch process that has to find records about a person, organization or address is to get "candidates" from the database using the keys defined for the names or addresses.

The "search strategy" used to achieve this goal must balance the conflicting objectives of making sure it does not miss possible candidates, yet at the same time not slow the process with too many irrelevant candidates. SSA-NAME3 provides a variety of "search strategies" or access paths to data using names or addresses as search keys.

Applications dealing with relatively reliable and complete data can use a high performance strategy, while applications dealing with less reliable data or with more critical problems need more complex strategies. For any name or address, SSA-NAME3 provides the business application with a logical access path to the set of records that are most likely to include relevant matching records. Out-of-thebox, four search strategies or "search levels" are provided. These are Narrow, Typical, Exhaustive and Extreme.

For example, an Exhaustive search for the name "John William Smith" may result in:
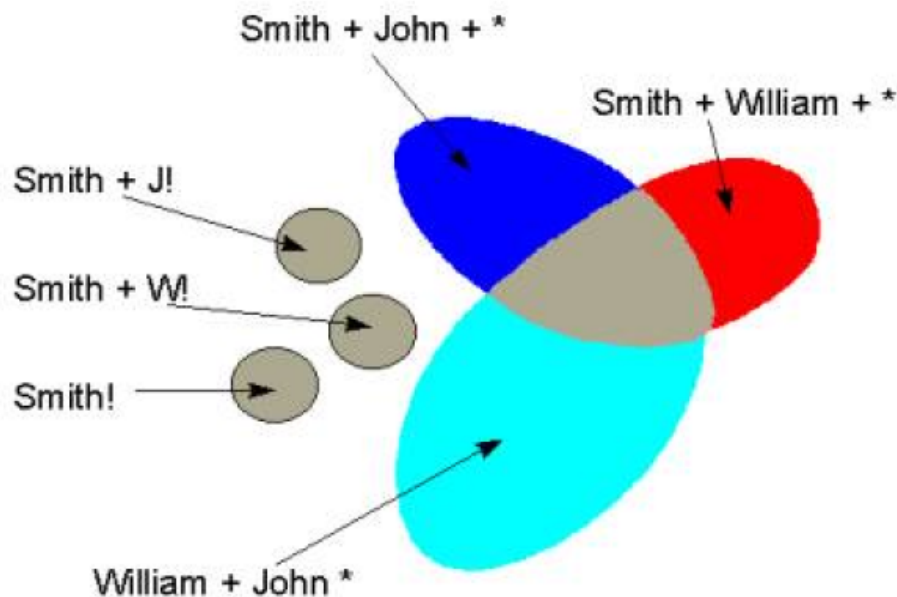
The above diagram is an of a search strategy where all records containing two similar words are found, as well as the major word (in this case "Smith") together with explicitly only the initials well as a set of records that contain only the major word.

Match Purposes

The second step in most search and matching applications requires that the system can automate or assist the operator in "choosing" the relevant matching records. In online transactions, where the operator makes the final choice, this Matching process can be used to eliminate the entirely irrelevant records, or rank them in order of relevance. In many transactions the application system can be made more reliable, if instead of allowing the operator the choice, the Matching process is fully automated. In the majority of batch systems the Match decisions have to be fully automated.

To allow the easy implementation of such high quality Matching processes, the SSA-NAME3 software provides a set of Match Purposes to "mimic" the matching decisions made by the very best business users. A Match Purpose will compare two records, typically the search record and a file record, and compute a Score and Match Decision.

The out-of-the-box Match Purposes provided are used for determining same Person_Name, Individual, Resident, Household, Family, Wide_Household, Address, Organization, Division, Corporate Entity, Contact and Wide_Contact. In addition, three different Match Levels can be chosen in conjunction with each Match Purpose. These Match Levels are Conservative, Typical and Loose.

Each Match Purpose will have one or more mandatory fields and a number of optional fields. For each Match Purpose and Match Level, special rules are defined for each field of available data that control how the results from each field should be combined into the overall Match Score. Pre-set Score thresholds deliver a Match "Decision" as either Accept, Undecided, or Reject. The user's application can override these thresholds, or use the raw Score to determine its own thresholds.

Standard Populations

The rules that define how the Key Building, Search Strategies and Match Purposes operate on a particular population of data are packaged with SSA-NAME3 in what are called Standard Population sets. There is one Standard Population set for each country, language or population that Informatica supports.

Each Standard Population set supports:

- Key Building and Searching on Names (Person and Organization) and Addresses
- Selectable Key and Search levels
- Matching for Purposes such as Same Person Name, Same Organization, Same Contact, Same Individual, Same Resident, Same Household, Same Division and Same Address
- Selectable Match levels and thresholds.

Custom Populations

For the user with unusual or special needs, Informatica provides a customization service that results in the generation of a Custom Population set. This can be plugged in and used just like a Standard Population set.

This Customization Service might be used, for example, by the user that needs to search on an entity other than a name or address, for example a Song Title.

Overriding Population Rules

SSA-NAME3 includes the ability to override certain types of Population Rules. There are two levels at which these overrides can be managed.

Using the **Population Override Manager**, a data analyst trained in the syntax of the SSA-NAME3 Edit rules, and versed in the consequences of making changes that may effect search and match quality and performance, and possibly require indexes to be rebuilt, can add or remove Edit rules from the Population set.

Through this utility, the analyst can also replace the packaged Frequency tables with ones built from the organization's own data.

Using the **Edit Rule Wizard**, a business or non-IT user can safely add certain types of Edit rules, typically new nicknames or synonyms, new noise words, or new phrase replacement rules, without requiring involvement from IT or the need to re-build indexes.

Multi-Country Support

Because SSA-NAME3 is ready to deal with various populations of data, it is practical to build systems in a manner that allows the same system to be deployed for many countries.

It is also practical to build systems where the data in one database is mixed from several countries. It is possible to "tune" critical systems to handle the fact that the searches are being generated in one country's language and character set yet the database has been built according to international rules so that it is usable in many countries.

In countries like Greece, or Israel, where both Roman character forms of names and addresses as well as the local country language and character forms are common, SSA-NAME3 allows searching and matching from Roman form to local language and vice versa.

SSA-NAME3 comes with Standard Population rule sets for over 50 countries and languages. These Standard Population sets include all of the Key Building, Search Strategy and Matching services required to effectively search and match on identity data sourced from that country and character set.

Informatica can also generate Custom Populations for new countries or for unusual searching and matching requirements.

SSA-NAME3 CJK-SUPPORT

SSA-NAME3 CJK-SUPPORT is the name given to a separately licensable product that expands the full facilities of SSA-NAME3 to handle the special nature of Chinese, Japanese and Korean data. Its features include the ability to recognize and encode double-byte characters in names and addresses and handle special representations of Chinese numbers. It also supports the mixed use of local phonetic (for example, Katakana) and Roman, often used to record foreign names and addresses in these countries.

Unicode Support

SSA-NAME3 also supports Unicode source data. For more information on how to use and specify Unicode input, see the *API REFERENCE Guide.*

C H A P T E R 5

SSA-NAME3 Implementation

This chapter includes the following topics:

SSA-NAME3 Implementation Overview

SSA-NAME3 is a software tool-kit that requires implementation by an analyst programmer/developer.

Because its functionality can often be shared by multiple systems and applications, higher-level design input into its implementation is often beneficial. In this way, the search and matching functionality might be implemented as a shared server process, forming part of the infrastructure of a system.

The implementation of SSA-NAME3 will normally also involve a Database Administrator. This is because the SSA-NAME3 Keys must be stored in a new database table.

A description of the SSA-NAME3 components will help identify the technical aspects of the product's implementation.

SSA-NAME3 Components Overview

The SSA-NAME3 software consists of a Callable Routine, Standard Populations, the Developer's Workbench, the Population Override Manager, the Edit RuleWizard, and the documentation needed to support these facilities, provided as HTML files, in PDF format and hard-copy.

The Callable Routine contains the proprietary algorithms and processes used for building keys and search strategies and for computing match decisions for a calling application. The Callable Routine is provided as a shareable, re-entrant object, such as a DLL or Shared Library, and also as a server process. It can be invoked through the documented API.

The Standard Populations contain the out-of-the-box Country and Language rules that are applied to the internal algorithms to support different countries, character sets, data types and application needs. The Standard Populations are accessible only through the Callable Routine, or via one of the SSA-NAME3 GUI Clients.

The SSA-NAME3 Developer's Workbench client is a Java GUI tool that helps a programmer prototype SSA-NAME3 calls. The Workbench is also used for Browser-based Help and Documentation, Generating Sample Program Code, Executing SSA-NAME3 Calls, Testing different SSA-NAME3 parameters and Producing debugging information for Informatica.

The Workbench can be run on any computer that has the correct version of Java installed and that can communicate with the computer where the SSA-NAME3 Callable Routine is situated. If the Workbench is installed on a different computer to the Callable Routine, the Callable Routine will need to be started as the SSA-NAME3 Server process, and the Workbench will communicate with it through TCP/IP. The Workbench can also run on the same computer as the SSA-NAME3 run-time environment.
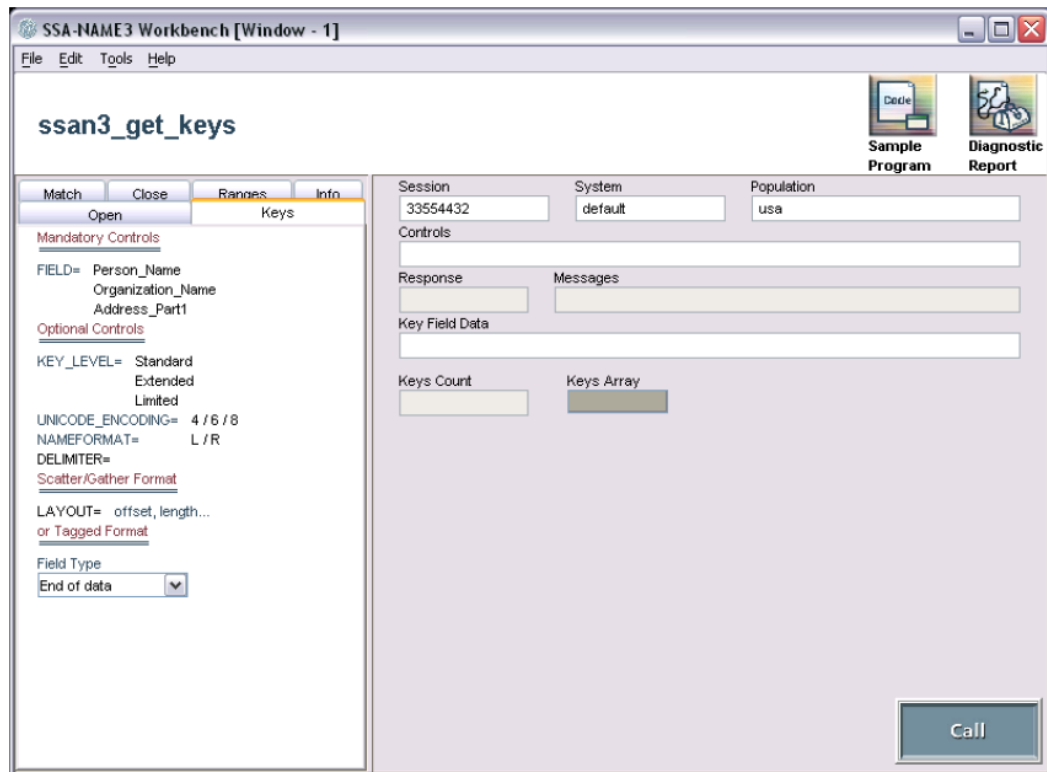

The **Population Override Manager** allows a data analyst trained in the syntax of the SSA-NAME3 Edit rules to add or remove Edit rules from the Population set. Using this utility, the analyst can also replace the packaged Frequency tables with ones built from the organization's own data.

The Population Override Manager can be run on any computer that has the correct version of Java installed and that can communicate with the computer where the SSA-NAME3 Callable Routine is situated. Unlike the Workbench, the Population Override Manager must communicate directly with the SSA-NAME3 Server process, even if the Callable Routine is installed on the same computer. For this reason, it requires that TCP/IP is installed and enabled.
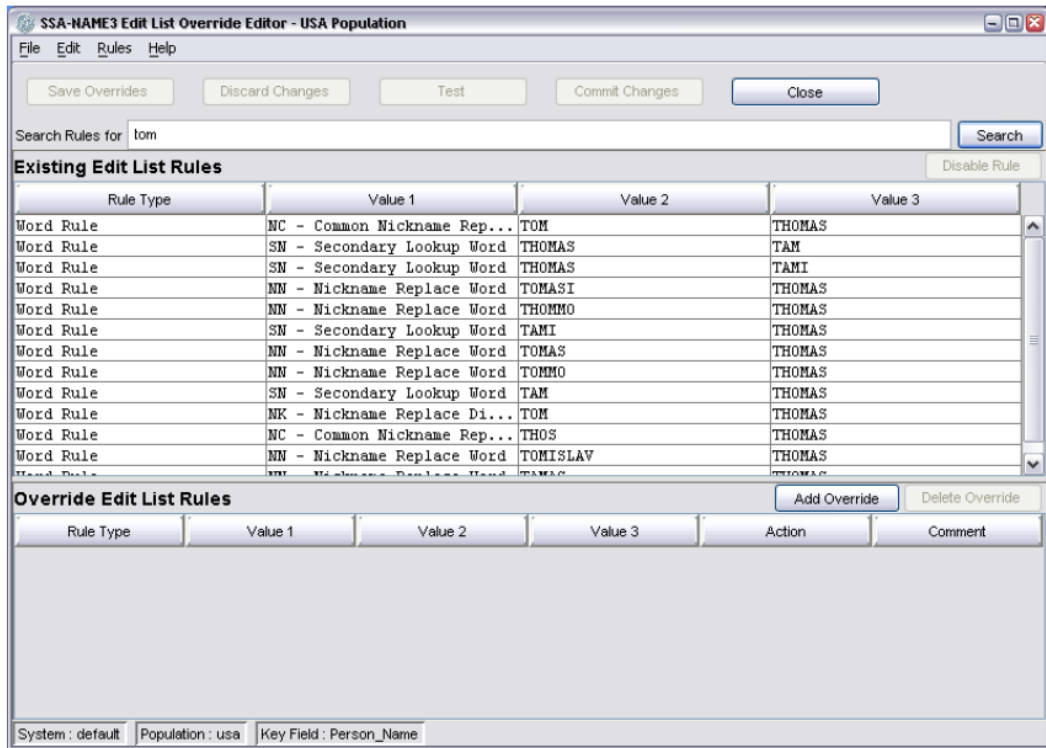
The **Edit Rule Wizard** allows a business or non-IT user to safely add certain types of Edit rules, typically new nicknames or synonyms, new noise words, or new phrase replacement rules, without requiring involvement from IT or the need to re-build indexes.
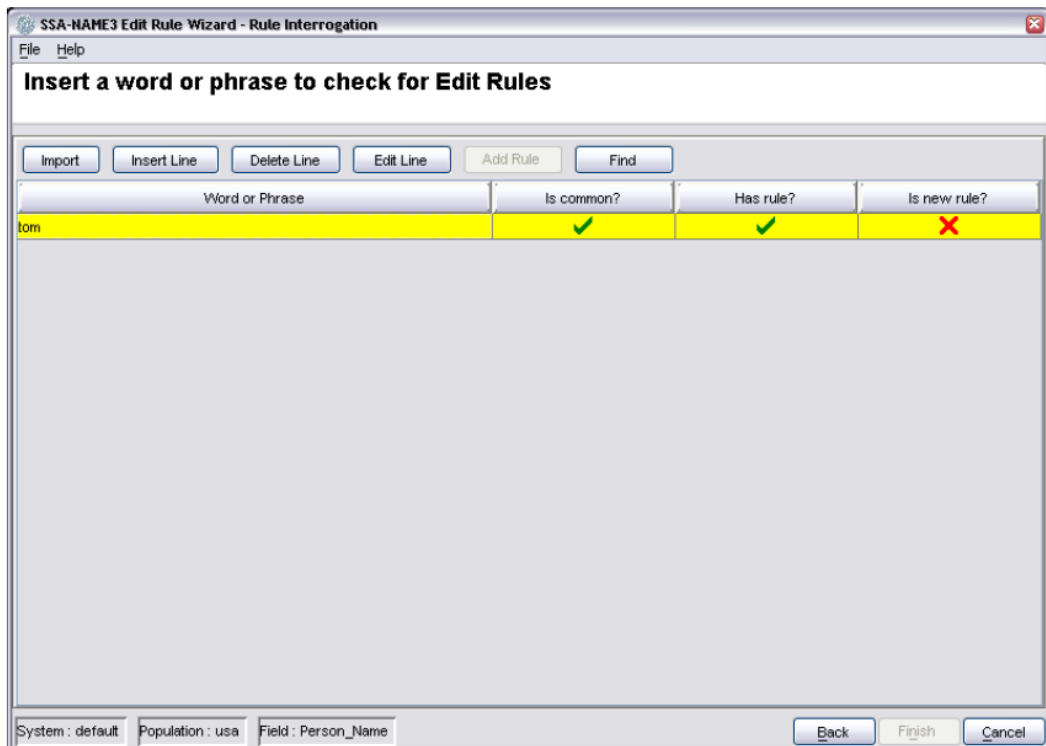
Implementation Architecture

The SSA-NAME3 Callable Routine can be invoked either through a socket interface using TCP/IP, or locally using shared (dynamic) libraries.

During development, this design allows the Workbench to be situated on a Windows client, accessing a Callable Routine on a server. It also means that the application program can be developed on a Windows client and access a single copy of the Callable Routine on the server.

This client/server architecture is not recommended to be the basis for a production implementation of an SSA-NAME3 application. It is recommended that the application that calls SSA-NAME3 and the SSA-NAME3 Callable Routine, even if the Callable Routine is started as a Server process, both reside on the same server in a production environment. Significant performance degradation may occur if this design is not followed.

Ssome applications will have to follow this architecture though. For example, the only way possible to use the Address Standardization Module from many platforms is to call the server callable routine on AIX, Solaris or Windows (all 32 bit). Another example is (even when co-resident on the same machine) the use of 64 bit client code with a 32 bit server or vice versa. A user's production application only requires access to the Callable Routine and the particular Population rule-set(s) used.

The Workbench is a development only component. The Population Override Manager is also intended to be used in a development environment. However it is possible to use the Edit Rule Wizard in a production environment.

Application Design

User applications use SSA-NAME3 for three major purposes, Key Building, Building Search Strategies and Matching.

Key Building

Key Building is the process of building keys for the name or address records to be searched. The user application calls the **ssan3_get_keys** function of SSA-NAME3, passing it a name or address and indicating which key field (`Person_Name`, `Organization_Name` or `Address_Part1`) it wishes to use. SSANAME3 passes back one or more keys ("Keys Array") which should be stored by the application in the user's database. These keys are typically 8-byte character keys; however, a 5-byte binary option is also available. Applications making use of this service are commonly the initial key-load program, and the name or address online update or batch maintenance programs.

An example of the program flow in a key building program is shown below:
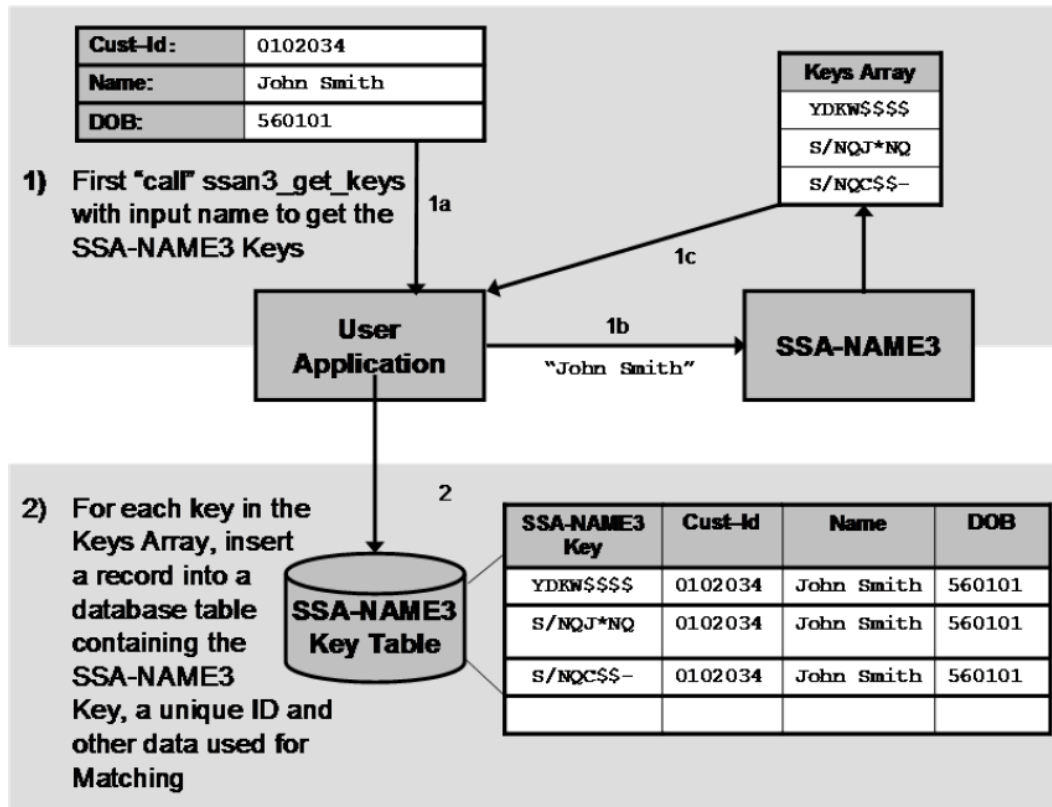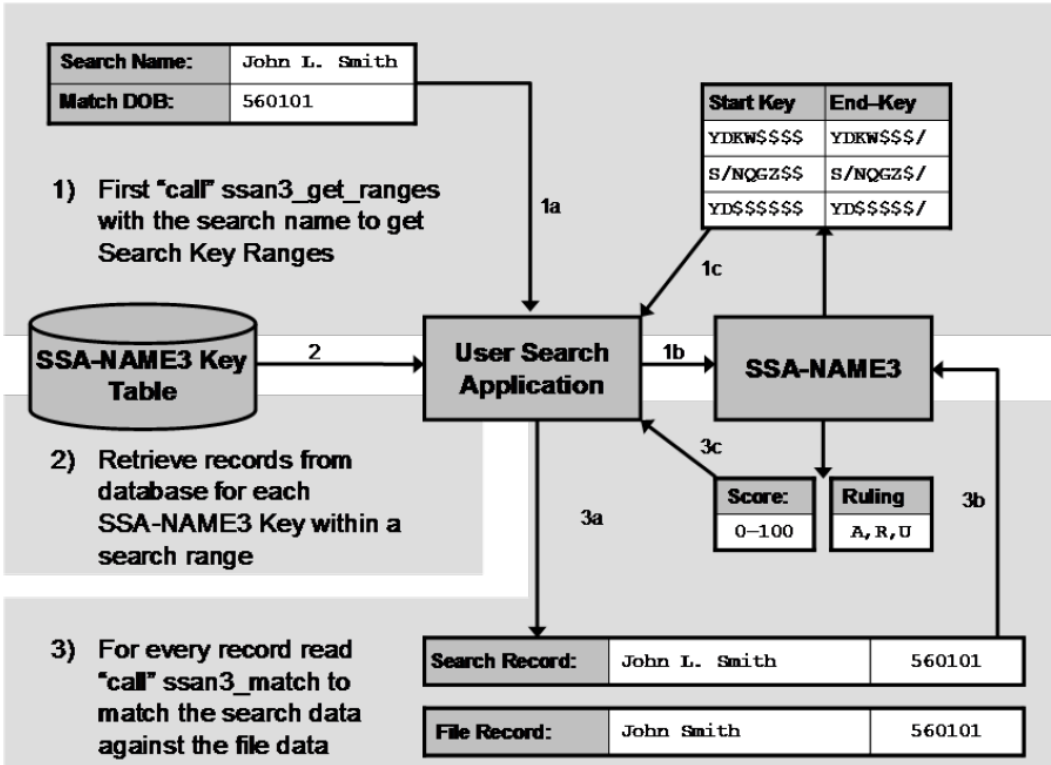
Building Search Strategies

The process of building a search strategy returns an array of search key ranges for a search name or address. The user's search application calls the **ssan3_get_ranges** function of SSA-NAME3, passing it the name or address and indicating which field and Search Strategy (Search Level) to use. SSANAME3 passes back an array of search key ranges that define the sets of records that are candidates for this name or address search. The user application then does a Select between start and end key or a key sequential access of its SSA-NAME3 key index for all of the key ranges returned. It then fetches the name or address and other data from the database to be used for matching.

The diagram below in the "Matching" section also shows the Search Strategy call.

Matching

Matching compares two records (a search record and a file record). After reading a candidate record in the search, the user application calls the **ssan3_match** function of SSA-NAME3 indicating which Matching "Purpose" and "Level" to use. The user application also passes the search name or address and any other identity data entered, plus the file name or address and the identity data retrieved for the file record. SSA-NAME3 compares the two records and passes back a Score between 0 and 100 and a Match Decision (Accept, Reject or Undecided). The user application may then use the Score and/or the Decision to match, filter or rank the records.

An example of the program flow for the search and matching application:

Search Name:	John L. Smith
Match DOB:	560101

Start Key	End–Key
YDKW$$$$	YDKW$$$/
S/NQGZ$$	S/NQGZ$/
YD$$$$$$	YD$$$$$/

1) First "call" ssan3_get_ranges with the search name to get Search Key Ranges

SSA-NAME3 Key Table

User Search Application

SSA-NAME3

2) Retrieve records from database for each SSA-NAME3 Key within a search range

Score:	Ruling
0–100	A,R,U

3) For every record read "call" ssan3_match to match the search data against the file data

Search Record:	John L. Smith	560101
File Record:	John Smith	560101

Part I: Glossary of Terms

This section provides information of terms used in SSA-NAME3.

Accept Limit

Accept-Limit is the score above which a candidate record is considered an accepted match. The Match Decision returned is set to "A". It is combined with Reject Limit such that records attaining a score between the Accept and Reject limits have a Match Decision set to U (Undecided). It is pre-defined in a Population rule-set, and can be overridden by the search application.

Candidates

The set of records returned from a name search. Each candidate should be compared with the original search record using the ssan3_match function and the accepted, and optionally the suspect, records displayed or otherwise further processed.

Compound Name

A name field which explicitly refers to more than one simple name, for example:

```
INFORMATICA CORPORATION dba IDENTITY SYSTEMS
JOHN SMITH and GEORGE BROWN
```

Controls

The keywords, values and parameters that govern how an API Function will operate and which are passed through the API. The format and contents of the controls will depend on the type of function being used. More information can be found in the Controls section of the *API REFERENCE* manual.

Custom Population

Standard Population rules modified by Informatica for a user with special search and matching requirements and re-packaged as a Custom Population. These may also have been built from scratch for a totally new type of population.

Delimiter

Special characters contained within the data which separate distinct fields or keywords.

Developer's Workbench

A Java-based GUI used by the developer for understanding and testing the various API Functions, accessing online documentation, and generating sample programs.

Edit Rule Wizard

A Java GUI tool that helps a business user safely add certain types of Edit Rules to the Standard or Custom Population without requiring specific knowledge of SSA-NAME3 or support from a programmer or data analyst. The types of rules that can be added using this tool are:

- Discard a word or phrase when searching and matching (e.g. a new "noise" word)

- Add a new replacement word or phrase when searching and matching (e.g. a new "abbreviation" or "acronym")

- Add a new compound name marker word

Efield

A field that is used for Matching, and not used for Key or Range building, that supports Edit-list rules. Examples of this are:

- ID numbers

- Telephone Numbers

- Postal Codes

- Efields benefit from Edit-lists to overcome such problems as when an ID number that contains all 9's should be considered a "null" number. Therefore, a rule would be required to treat all 9's as a noise word.

Error Message

A description of any error which may have occurred during an API Function call.

Extended Keys

For high-risk and critical search applications, this is the Key Level to be used when generating SSANAME3 Keys. In contrast to Standard Keys, Extended Keys include keys built from additional token concatenation.

File Data

The data retrieved from the file as a result of finding a set of candidate records using the key ranges returned from the **ssan3_get_ranges** function. The File Data is compared with the Search Data by the **ssan3_match** function to calculate a Score and Match decision.

Filtering

The process of discarding candidates that fail to meet a certain Score threshold or are deemed "Rejected". This reduces the number of records that need to be passed back over the network, shown to the user or further processed by a program.

Function

An SSA-NAME3 API Function which is called from the application to perform a distinct task. For example, **ssan3_get_keys** will generate SSA-NAME3 Keys; **ssan3_get_ranges** will generate a Key Ranges Array; **ssan3_match** will match a pair of records and return a Score and Match Decision. These functions and more are defined in detail in the *API REFERENCE* manual.

Fuzzy Keys

In the context of SSA-NAME3, Fuzzy Keys is a term that refers to the special SSA-NAME3 Keys built from names or addresses that have been treated by a variety of techniques to overcome the error and variation in the data.

Initial

A single character word or the first character of a word.

Key Field

The field used to build SSA-NAME3 Keys using the **ssan3_get_keys** function. In SSA-NAME3 Standard Populations, the supported Key Fields are Person_Name, Organization_Name and Address_Part1.

Key Field Data

The value(s) of the field used to build SSA-NAME3 Keys using the **ssan3_get_keys** function call. The keys generated from the call are stored by the user's application in a user-defined "SSA-NAME3 Key Table" within the database. The SSA-NAME3 Key Table is designed by the user's DBA. The SSANAME3 Keys column must be indexed.

When using the **ssan3_get_ranges** function from a search application, the Key Field Data will contain the value(s) of the field used to initiate the search.

Key Field Data may consist of one value or several repeating values. Examples of repeating values are: a person's name and their maiden name; a residential address and a mailing address.

Key Generation

The process whereby a user application calls the **ssan3_get_keys** function to generate SSA-NAME3 Keys from the Key Field Data (typically a name or an address). The application will then store these keys in a database table referred to as the SSA-NAME3 Key Table.

Keys Count

The number of keys returned from the **ssan3_get_keys**function call. This value is used by the application code to ensure that all of the keys returned are stored in the SSA-NAME3 Key Table.

Key Level

Refers to the number and variety of keys to be generated by an **ssan3_get_keys** call. The three Key Levels are Standard keys, Extended keys and Limited keys.

Limited Keys

If disk space is limited, SSA-NAME3 can generate "Limited" SSA-NAME3 Keys. Limited keys are a subset of Standard keys. However, the designer/developer should be aware that the use of Limited keys, while saving on disk space, may also reduce search reliability.

Local Population

A Standard Population or Custom Populationthat has been modified locally via either the Population Override Manageror Edit RuleWizard.

Major Word

The word in a Name identified as being the most significant word. In some Search Strategies, it is used as the primary part of the Search key ranges, and for extra weighting in some Match Purposes (e.g. family name in a Household Purpose).

Matching

The process whereby a user application calls the **ssan3_match** function to compare two records, usually a Search and a File record, and compute a Score and Match Decision.

Match Purpose

The ultimate business purpose of the search/match application. This will be provided as a parameter to the **ssan3_match** function. Examples of Match Purposes are "same name", "same individual", "same resident", "same household", "same organization", "same division", "same corporate entity", "same contact", "same address".

Match Decision

A 1-byte character value which identifies the judgment on the matched records. Values are "A" for Accept, "U" for Undecided and "R" for Reject. The thresholds by which these decisions are chosen can be varied by the user.

Match Level

Used in defining the level of Matching to be performed for a particular search application. In most Standard Populations, possible values are Conservative, Typical and Loose. The three possible values allow adjustment to the "tightness" of the match.

Minor Word

Any word in a Name which is not the Major word.

Name

The name of a person, company, business or organization; an address; a product title, song title or book title; any short description. A name consists of a number of words and optionally codes, each with a limit of 24 characters.

Name Format

An internal setting that specifies at what end of a name or address (Left or Right) the Major Wordcan be found. This can be overridden by an application program in certain API calls.

Noise Words

Words that do not contribute to, and can impede, a search or match function. Such words are removed when SSA-NAME3 processes a name through an **ssan3_get_keys**, **ssan3_get_ranges** or **ssan3_match** call. Examples of Noise Words are Personal Titles (e.g. Mr., Mrs.), Street Types (, Rd.) and Company legal endings (e.g. Inc., Ltd.). As with other Edit rules, Noise Words are population specific and vary according to what Standard Population is being used.

Population

This typically refers to the country and language of the data to be used in the search system; however, populations can be both super- and sub-sets of country and language populations. An example of super-set population is the combination of all Western European populations into the one population for searching and matching. An example of a sub-set population is the USA's OFAC list of Specially Designated Nationals.

Population Override Manager

A Java GUI tool that helps a trained data analyst override some of the Standard Population rules that are supplied with the product, or provided in the form of a Custom Population. The types of rules that can be overridden using this tool are:

- Edit-list rules
- Frequency tables
- Scalar Frequency Tables

Note: Use of this tool without proper training from Informatica is not recommended, as improper use can adversely affect the reliability and performance of the search application(s).

Ranges Array

Returned from a call to the **ssan3_get_ranges** function. The Ranges Array is a set of "Start" and "End" SSA-NAME3 Key values. These should be used by the user's search application to form a set of SQL select statements that retrieve records within those ranges.

Ranges Count

The number of in the Ranges Array. This is the number of ranges which the calling program must process.

Ranking

The process of sorting the Matched candidates in descending order by Score in order to display the records to the user in descending order of their likeness to the search identity.

Reject Limit

Reject-Limit is the score below which a candidate record is considered a rejected match. The Match Decision returned is set to "R". It is combined with Accept Limit such that records attaining a score between the Accept and Reject limits have a Match Decision set to U (Undecided). It is pre-defined in a Population rule-set, and can be overridden by the search application.

Reliability

A measure of the likelihood that a Search Strategy will find a name in the database if one exists that should be considered a match to the search name.

Required Keys

Refers to the Standard, Extended or Limited SSA-NAME3 Keys computed when a name or address is processed by the **ssan3_get_keys** function. They are referred to as "required" because all of the keys must be stored in the database table.

The default SSA-NAME3 Keys are 8 bytes in length and consist of printable characters. An option exists to generate 5-byte binary keys if your database supports such keys. The application program will store these key values in a separate table within the database specifically designed and optimized by your DBA for searching and matching. This table will be sorted and indexed on the column storing the SSA-NAME3 Keys.

Response Code

Gives an indication of the validity of a call to SSA-NAME3. A Response Code value of zero indicates a successful call. If the Response Code is not zero, then a description of the problem will be reported in the Error Message parameter.

Scatter / Gather Data Format

This is a method of formatting the input data when using **ssan3_get_keys**, **ssan3_get_ranges**and **ssan3_match** function calls.

Score

A numeric value between 000 and 100 returned from the **ssan3_match** call. It indicates how close a match was achieved after comparing the Search Data and File Data. The actual Score returned will depend on the Match Purpose, Match Level and the Search and File Data being compared.

Score Limit

A numeric value between 000 and 100 that defines the threshold for the Match Decisionfor a specific Match Purposeand Match Levelfor a given Population rule-set. Score Limits are pre-defined in the Population rule-sets, and can be overridden by the calling program.

Search Data

The transaction data which contains the search information. It will contain the field value used to drive the search (that is, used in the **ssan3_get_ranges** call) as well as all of the available data to be compared with the File Data during the **ssan3_match** call.

Search Dialogue

The method by which a search application receives search data from an input screen, processes the Ranges Array generated from the search data, and displays the ranked records back to the user.

Search Level

Used in defining the type of Search Strategyto use for a particular search application. In most Standard Populations, possible values are Narrow, Typical, Exhaustive and Extreme. The four possible values allow adjustment to the "thoroughness" of the search. The wider the search, the more candidates are typically returned, which may increase the reliability of the search but also use more resources and take longer.

Search Strategy

The combination of Key Field and Search Level passed to the **ssan3_get_ranges** function to generate the Ranges Array.

Selectivity

The percentage of the database (that is, number of candidates / total number of database rows) that is retrieved to satisfy a particular search.

SSA-NAME3 Keys

SSA-NAME3's intelligent keys are computed when a name or address is processed by the **ssan3_get_keys** function. SSA-NAME3 Keys can be of 3 types: Standard Keys, Extended Keys or Limited Keys. The Keys are 8 bytes in length and consist of printable characters. An option exists to generate 5-byte binary keys. The application program will store these key values in a new table within the database or in a new indexed file specifically designed and optimized for searching and matching. This table will be sorted and indexed on the column storing the SSA-NAME3 Keys.

Standard Keys

For typical applications, this is the Key Level to be used when generating SSA-NAME3 Keys. Standard Keys overcome more variation than Limited Keys while using less disk space than Extended Keys. High-risk and critical applications, however, should use Extended keys.

Standard Population (SP)

Standard algorithms which support various searching and matching rules and requirements, typically for a specific language and country. Note: all Standard Populations are delivered with the product, however a separate license is required to use the double-byte character sets covered by the `SSANAME3- CJK-SUPPORT` product.

System

Describes the use for the Name Search, for example, your project name. The System name is used to define the name of a folder or sub-directory where the Standard or Custom Population for this system should be stored and secured.

Tagged Data Format

This is a method of formatting the input data when using **ssan3_get_keys**, **ssan3_get_ranges** and **ssan3_match** function calls.

In Tagged Format, the offsets and lengths of the data fields being passed do not need to be specified. Instead, a notation of labels and delimiters is used to break up the fields. By default the delimiter is an asterisk but it can be user defined.

Token

The word or code components of a Name or Address.

Server Platform

The combination of hardware and operating system that will host the application that calls SSANAME3 and accesses the database.

Workbench

See the *WORKBENCH USER* guide for more information.

INDEX