EDQ's Name Variant Recognition

Presented By:

Mike Matthews and William Indest

ORACLE

Introduction

There may be hundreds of different ways of representing a name, with different spellings, abbreviations and language translations producing different variants; Johannes, Juan, Giovanni, João, Jean, Johann, Jannis, Hans, Ieuan, Ifan, John and Jonathan are all derived from the same Latin root. When people (or their data) cross borders their name may be spelled differently or written using a different writing system.

Conversion from one writing system to another is only part of the puzzle for matching. There are often hundreds of ways of writing names from another language. It is therefore necessary to provide Reference Data to recognize name variants in all major languages. The aim is to convert names to a standard form, but then recognize all possible variant forms in matching.

The EDQ Customer Data Services (CDS) pack includes more than 4 million rows of name variant reference data that it uses to match records with different versions of the same name. Within one writing system, the same name may be written in many different ways (e.g. there are 204 Latin versions of the name Muhammad in the knowledge base - 47 of them in common use).

Multilingual Support

User Interface Support

The EDQ (12.1.3) user interface supports the following languages:

- English
- French
- Italian
- German
- Spanish
- Brazilian Portuguese
- Chinese
- Japanese
- Korean

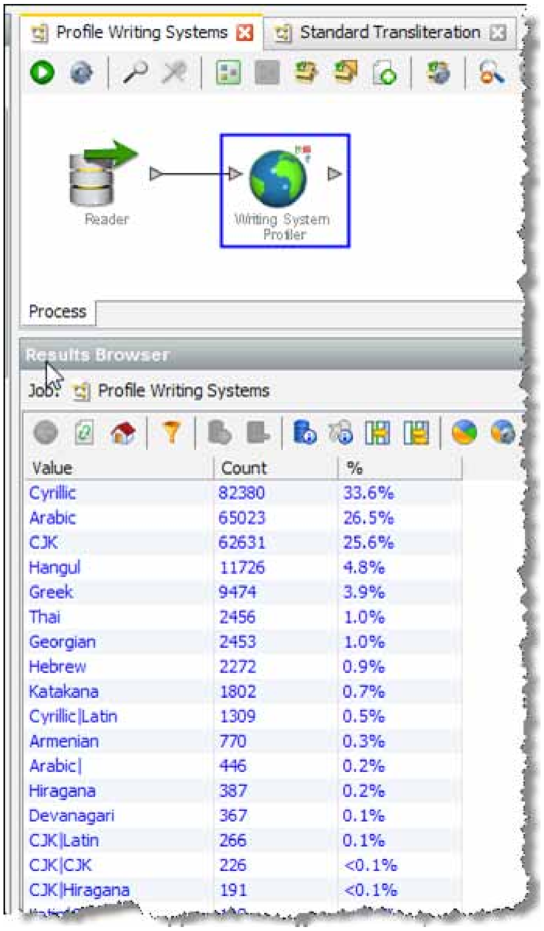
Data Support

EDQ supports data stored using different writing systems in a number of ways.

Profiling

Always profile the data first to discover which writing systems exist in your data. EDQ makes this very simple with the Writing System

Profiler. The Writing System Profiler should be used to identify the writing systems that exist in the data:



Value	Count	%
Cyrillic	82380	33.6%
Arabic	65023	26.5%
CJK	62631	25.6%
Hangul	11726	4.8%
Greek	9474	3.9%
Thai	2456	1.0%
Georgian	2453	1.0%
Hebrew	2272	0.9%
Katakana	1802	0.7%
Cyrillic|Latin	1309	0.5%
Armenian	770	0.3%
Arabic|	446	0.2%
Hiragana	387	0.2%
Devanagari	367	0.1%
CJK|Latin	266	0.1%
CJK|CJK	226	<0.1%
CJK|Hiragana	191	<0.1%

Writing System Conversion

Converting the writing system is most useful when it comes to matching records between scripts. EDQ makes use of three different techniques to convert data from one writing system to another within the Customer Data Services pack and in the Oracle Watchlist Screening application. The techniques are:

- **Transliteration** uses character-level rules.
- **Transcription** directly converts using a dictionary but does not use semantics.
- **Translation** uses a dictionary and uses semantics (convert the meaning of an item of data from one language to another) See this Wikipedia article for more on transliteration and how it differs from transcription.

There are two use cases:

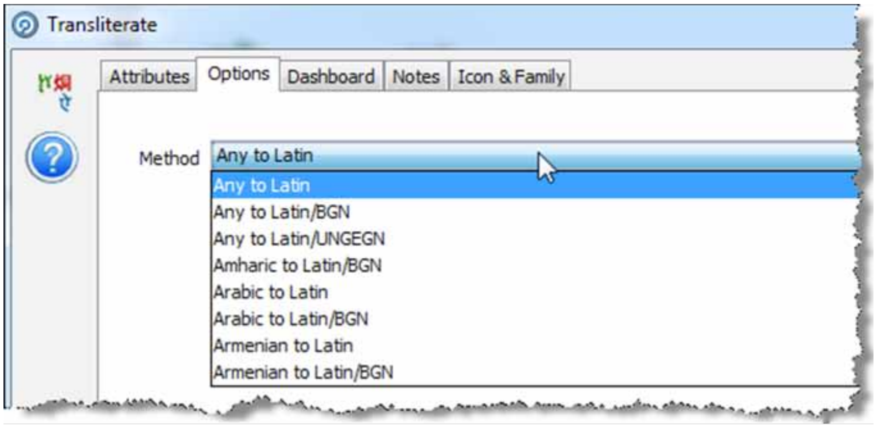
1) Match individual names between scripts

Transliteration is used for most languages and then transcription is used where transliteration does not work well enough.

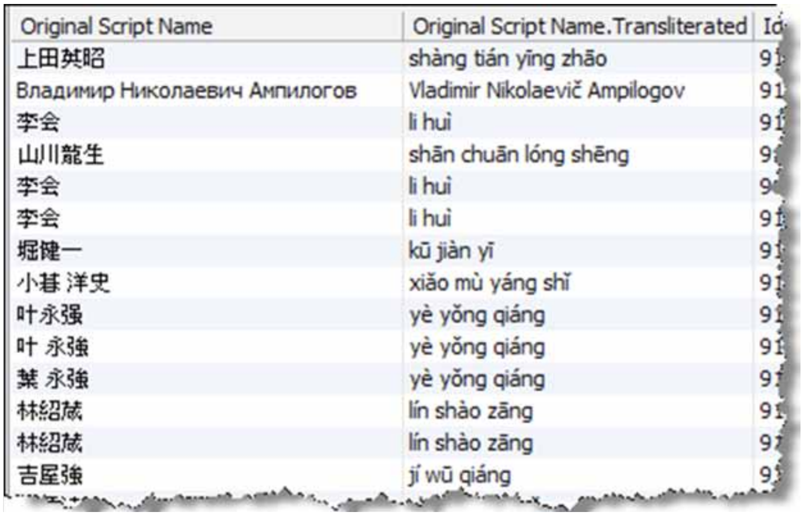
2) Match company names between scripts

EDQ uses a combination of transliteration, transcription and translation. Translation of company name data is often required for matching, but there are not many good reference data sources available. It is therefore advisable to capture names in one language where effective matching is required.

Basic transliteration can be achieved using the EDQ Transliterate processor. This processor transliterates between more than forty writing systems:



Method: Any to Latin



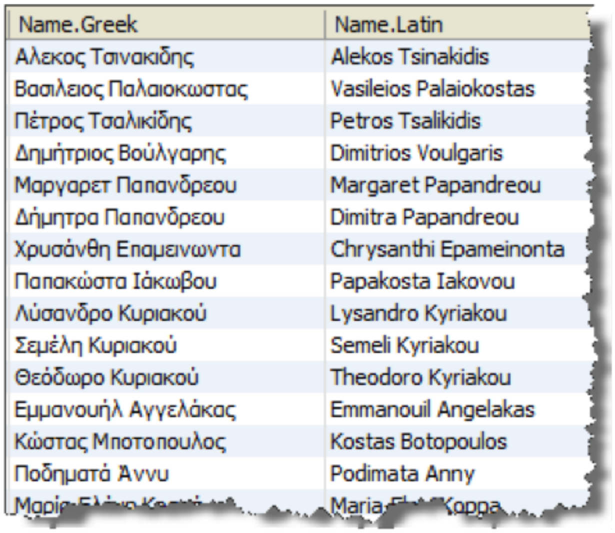
Original Script Name	Original Script Name.Transliterated	Id
上田英昭	shàng tián yīng zhāo	91
Владимир Николаевич Ампилогов	Vladimir Nikolaevič Ampilogov	91
李会	li huì	91
山川龍生	shān chuān lóng shēng	9
李会	li huì	9
李会	li huì	91
堀健一	kū jiàn yī	91
小暮 洋史	xiǎo mù yáng shǐ	91
叶永强	yè yǒng qiáng	91
叶 永強	yè yǒng qiáng	91
葉 永強	yè yǒng qiáng	9
林紹臧	lín shào zāng	91
林紹臧	lín shào zāng	9
吉屋強	jí wū qiáng	9

For some languages, such as *Russian* using the *Cyrillic* script, it is preferable to use specific transliteration logic; by using simple reference data and the Character Replace processor, we gain more control over the transliteration process.

Arabic data is not effectively converted using transliteration, so we use a transcription approach instead. This utilizes a prepared version of [The CJK Institute's (CJKI)]() Database of Arabic Names in Arabic (DANA). This recognizes more than 200,000 names and name variants, but where a match is not found, we fallback to using transliteration.

Transliteration Example: Greek

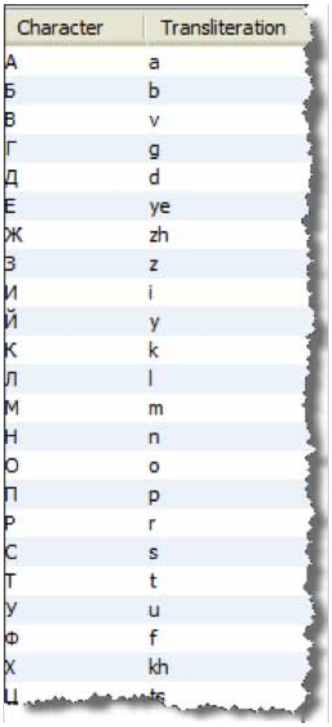
Use the EDQ Transliterate processor with the Greek to Latin option and a few simple additional rules to remove 'Diacritics':



Name.Greek	Name.Latin
Αλεκος Τσινακιδης	Alekos Tsinakidis
Βασιλειος Παλαιοκωστας	Vasileios Palaiokostas
Πέτρος Τσαλικίδης	Petros Tsalikidis
Δημήτριος Βούλγαρης	Dimitrios Voulgaris
Μαργαρετ Παπανδρεου	Margaret Papandreou
Δήμητρα Παπανδρεου	Dimitra Papandreou
Χρυσάνθη Επαμεινωντα	Chrysanthi Epameinonta
Παπακώστα Ιάκωβου	Papakosta Iakovou
Λύσανδρο Κυριακού	Lysandro Kyriakou
Σεμέλη Κυριακού	Semeli Kyriakou
Θεόδωρο Κυριακού	Theodoro Kyriakou
Εμμανουήλ Αγγελάκας	Emmanouil Angelakas
Κώστας Μποτοπουλος	Kostas Botopoulos
Ποδηματά Άννυ	Podimata Anny
Μαρί[illegible]	Maria [illegible] Koppa

Transliteration Example: Russian

For some writing systems, it is preferable to use specific transliteration logic, rather than use the Transliterate processor. For Russian, the standard approach uses simple Reference Data and the Character Replace processor to transliterate at the character level. This allows greater control over the transliteration process:



Character	Transliteration
А	a
Б	b
В	v
Г	g
Д	d
Е	ye
Ж	zh
З	z
И	i
Й	y
К	k
Л	l
М	m
Н	n
О	o
П	p
Р	r
С	s
Т	t
У	u
Ф	f
Х	kh
Ц	ts

Transliteration Example: Arabic

For Arabic, transliteration is not effective enough for proper matching as محمد is transliterated by most methods to ‘**mhmd**’ but realy represents ‘**Muhammad’** so EDQ uses transcription from a prepared version of DANA (CJKI’s Database of Arabic Names in Arabic) which recognizes over 200,000 names. When the name dictionary is not hit, an exception can be flagged and transliteration can be used as a fallback on the outliers.

Writing System Conversion Efficacy

- **High** - Russian, Simplified Chinese, Traditional Chinese, Korean, Russian, Other Cyrillic, Greek, Georgian, Armenian, Hiragana, Katakana, Bengali, Devanagari, Oriya, Gurmukhi, Telugu, Thaana, Amharic
- **Medium** - Arabic, Kanji (with no reading), Tamil, Malayalam, Kannada
- **Low** - Thai
- **No Capability** - Hebrew. Mongolian, Nepali, Khmer, Lao, Urdu

Customer Data Services Pack

The Customer Data Services Pack or CDS may be downloaded by licensees of EDQ. It provides language and writing system-specific extensions for use with both EDQ and Oracle Watchlist Screening:

- Processors to transliterate between writing systems
- Processors/processes to generate standardized data for matching purposes
- Dictionaries to use to derive name culture(s) or gender(s), and to use in matching to recognize name variants
- Dictionaries of anonymous values (to strip in matching), for example: 1) Business name trailer information , and; 2) Names that are frequently inserted

The Customer Data Servicd Pack deploys a Best Practice approach to transliteration and transcription. CDS automatically routes records to the best conversion method (transliteration or transcription) based on the writing system used; this may be derived from the data. Where a writing system is shared (such as CJK for Chinese and Japanese data), an input language setting is used to route records. The language setting may be derived from the data (for example Nationality/Country)

Best Practices

Name dictionaries are normally simply applied within matching as another matching technique, in a similar way to current Name Standardization. Exceptions are made for complex languages where we need to prepare standardized versions of names, for instance, Latin versions of Arabic names require some additional preparation.

We do not overwrite the source data; original names are retained and used in matching. Using name variant recognition with loose text matching is not recommended as it will generate large numbers of false positives - consider disabling weaker textual match rules once name variant recognition is enabled

Name Dictionaries

The Name Dictionaries are just data and can be processed like any other type of data. This means that they are NOT a black box and users can easily view them to understand why variants matched, or did not match.
The dictionaries have attributes for:

- Gender derivation (including ambiguity detection)
- Culture derivation
- Frequency of variant (used to filter the data down to the most important variants)

The data is compiled, verified and licensed by professional linguists, such as The CJK Institute. They are regularly updated and includes recognition of where a name is a variant of different root names.

Why Use Name Dictionaires?

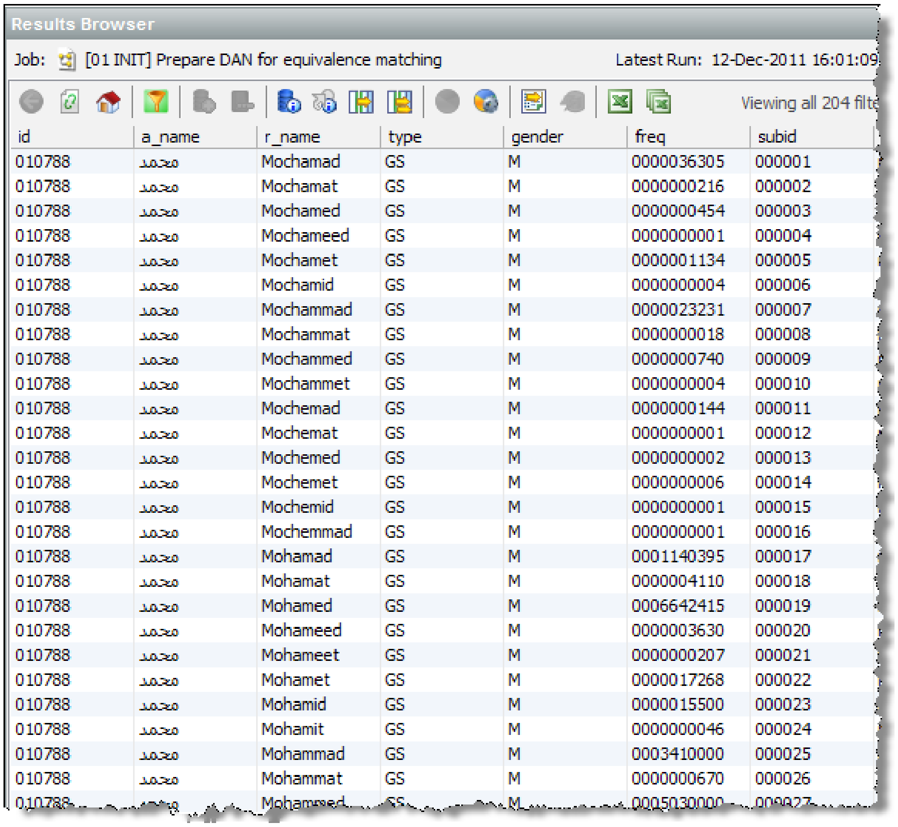
The same name in one writing system may be written in many ways in another, making matching difficult. For example, there are 204 Latin versions of the Arabic name:

محمد

which is Muhammad in the knowledge base and even filtering this down leaves 47 frequently used variants.

Often, the name variants are not textually similar and there is no standard for writing names and we also find nicknames, even in formal data.

What do the name dictionaries look like?



Job: [01 INIT] Prepare DAN for equivalence matching Latest Run: 12-Dec-2011 16:01:09

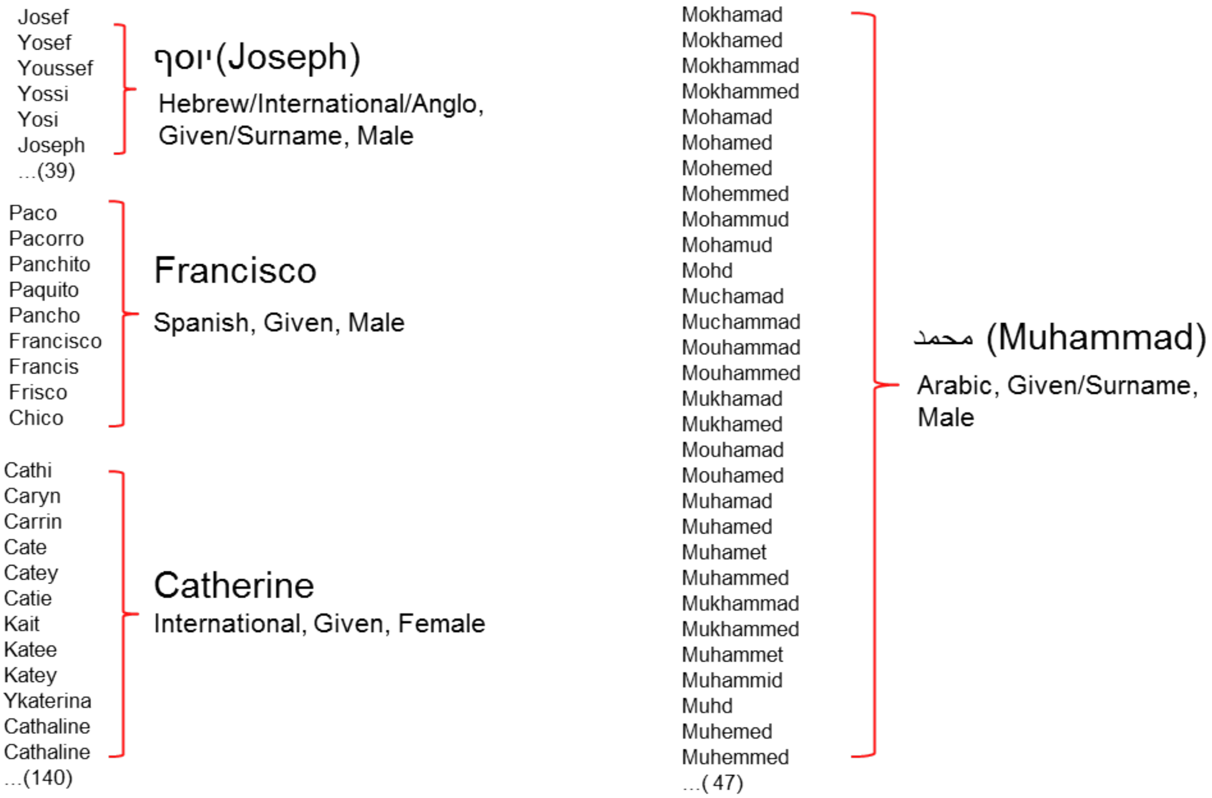
id	a_name	r_name	type	gender	freq	subid
010788	محمد	Mochamad	GS	M	0000036305	000001
010788	محمد	Mochamat	GS	M	0000000216	000002
010788	محمد	Mochamed	GS	M	0000000454	000003
010788	محمد	Mochameed	GS	M	0000000001	000004
010788	محمد	Mochamet	GS	M	0000001134	000005
010788	محمد	Mochamid	GS	M	0000000004	000006
010788	محمد	Mochammad	GS	M	0000023231	000007
010788	محمد	Mochammat	GS	M	0000000018	000008
010788	محمد	Mochammed	GS	M	0000000740	000009
010788	محمد	Mochammet	GS	M	0000000004	000010
010788	محمد	Mochemad	GS	M	0000000144	000011
010788	محمد	Mochemat	GS	M	0000000001	000012
010788	محمد	Mochemed	GS	M	0000000002	000013
010788	محمد	Mochemet	GS	M	0000000006	000014
010788	محمد	Mochemid	GS	M	0000000001	000015
010788	محمد	Mochemmad	GS	M	0000000001	000016
010788	محمد	Mohamad	GS	M	0001140395	000017
010788	محمد	Mohamat	GS	M	0000004110	000018
010788	محمد	Mohamed	GS	M	0006642415	000019
010788	محمد	Mohameed	GS	M	0000003630	000020
010788	محمد	Mohameet	GS	M	0000000207	000021
010788	محمد	Mohamet	GS	M	0000017268	000022
010788	محمد	Mohamid	GS	M	0000015500	000023
010788	محمد	Mohamit	GS	M	0000000046	000024
010788	محمد	Mohammad	GS	M	0003410000	000025
010788	محمد	Mohammat	GS	M	0000000670	000026
010788	محمد	Mohammed	GS	M	0005030000	000027

The Name Dictionaries are just data and as such, can be processed like any other type of data. The name dictionaries are NOT a black box so users can easily understand and comprehend why variants matched or did not match. The data dictionaries have attributes for:

- Gender derivation (including ambiguity detection)
- Culture derivation
- Frequency of variant which is used to filter the data down to the most important variants

The name dictionaries are compiled, verified by, and licensed from professional linguists. They include recognition of where a name is a variant of different root names. The name dictionaries are updated regularly.

Name Variant Examples



Josef
Yosef
Youssef
Yossi
Yosi
Joseph
...(39)

יוסף(Joseph)
Hebrew/International/Anglo, Given/Surname, Male

Paco
Pacorro
Panchito
Paquito
Pancho
Francisco
Francis
Frisco
Chico

Francisco
Spanish, Given, Male

Cathi
Caryn
Carrin
Cate
Catey
Catie
Kait
Katee
Katey
Ykaterina
Cathaline
Cathaline
...(140)

Catherine
International, Given, Female

Mokhamad
Mokhamed
Mokhammad
Mokhammed
Mohamad
Mohamed
Mohemed
Mohemmed
Mohammud
Mohamud
Mohd
Muchamad
Muchammad
Mouhammad
Mouhammed
Mukhamad
Mukhamed
Mouhamad
Mouhamed
Muhamad
Muhamed
Muhamet
Muhammed
Mukhammad
Mukhammed
Muhammet
Muhammid
Muhd
Muhemed
Muhemmed
...(47)

محمد (Muhammad)
Arabic, Given/Surname, Male

Name Variant Coverage

- Arabic: ~250,000 name variants (distilled from >6.5m)
- Japanese: ~330,000 name variants (distilled from >3.4m)
- Chinese: ~43,000 name variants (distilled from >200,000)
- Korean: ~20,000 name variants (distilled from >40,000)
- Anglo: ~8,000 name variants

- International: ~7,000 name variants
- Hebrew: ~5,000 name variants
- German: ~1,500 name variants
- French: ~1,000 name variants
- Russian: ~1,000 name variants
- Gaelic/Irish: ~1,000 name variants
- Indian: ~500 name variants
- Spanish: ~500 name variants
- Portuguese: ~500 name variants
- Italian: ~500 name variant

Summary

EDQ's name variant capabilities are some of the best in the industry. No black box so the user has complete transparency of how names are handled across multiple writing systems. With over four million variants, EDQ has the most complete solution in the market today.

Capabilities are provided out of the box so the user can deploy quickly and begin understanding the state of their data with confidence that the quality of their information will improve.

Title: EDQ’s Name Variant Recognition
Authors: Mike Matthew and William Indest

EOF

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.
Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.