
Anonymous Oracle Applications HR Data

A Net 2000 Ltd. White Paper

Abstract

Information in many test and development Oracle Applications HR systems must be rendered anonymous for security purposes. Such data masking helps prevent the unauthorized visibility of the data content and secures the data against accidental release. This White Paper is an overview of the primary Oracle Applications HR tables which require such data sanitization and the actions which should be performed on them.

Implementation Details

This paper is a general discussion of the tables in the Oracle Applications HR schema and their masking requirements. Net 2000 Ltd., the authors of this paper, sell a software tool called [Data Masker](#) which includes a pre-prepared set of rules for Oracle Applications HR schemas. These rules resolve the issues discussed herein and make the masking of the information a simple and repeatable process.

Having said that, this paper really is a generic survey of the sanitization requirements of a Oracle Applications HR schema and there will be no further reference to any software. If you wish to know more, or have any questions about the issues and techniques described below please contact us.

Net 2000 Ltd.
Info@Net2000Ltd.com
<http://www.Net2000Ltd.com>

Table of Contents

Disclaimer	1
Why Provide Anonymous Information in Test Oracle HR Schemas?	2
Sanitizing the Oracle Applications HR Schema	3
Overview	3
Data Sanitization Techniques	3
Data Scrambling Issues	3
Oracle Applications HR Tables Which Require Masking	4
Important Note	4
The PER_ALL_PEOPLE_F Table	5
Overview	5
Decisions	5
Specific actions on PER_ALL_PEOPLE_F Columns	6
The PER_ALL_ASSIGNMENTS_F Table	8
Overview	8
Specific actions on PER_ALL_ASSIGNMENTS_F Columns	8
The HR_COMMENTS Table	8
Overview	8
Specific actions on HR_COMMENTS Columns	8
The PER_PHONES Table	8
Overview	8
Specific actions on PER_PHONES Columns	8
The PER_ADDRESSES Table	9
Overview	9
Specific actions on PER_ADDRESSES Columns	9
The PAY_EXTERNAL_ACCOUNTS Table	9
Overview	9
Specific actions on PAY_EXTERNAL_ACCOUNTS Columns	9
Summary	10
Data Sanitization Techniques	10
Data Scrambling Issues	10

Disclaimer

The contents of this document are for general information purposes only and are not intended to constitute professional advice of any description. The provision of this information does not create a business or professional services relationship. Net 2000 Ltd. makes no claim, representation, promise, undertaking or warranty regarding the accuracy, timeliness, completeness, suitability or fitness for any purpose, merchantability, up-to-dateness or any other aspect of the information contained in this paper, all of which is provided "as is" and "as available" without any warranty of any kind.

Oracle HR schemas vary widely and each has a unique configuration. Readers should take appropriate professional advice prior to performing any actions.

Masking Oracle HR Schemas

Why Provide Anonymous Information in Test Oracle HR Schemas?

The information in an HR schema is sensitive and nearly all countries place a legal obligation on the data holder which requires its protection. Besides the legal consequences, an escape of such information will usually cause considerable public relations damage.

So the information must be protected – of that there is no doubt. The issue then becomes a matter of implementation. In general, a policy of *minimum required access* is usually adopted.

In a production environment it is usually possible to protect the information by restricting access to the underlying data. Strict controls are in place and carefully designed user interfaces present a managed view. Test and development systems are different. They present an environment in which access is usually much wider. Information is visible to more people and those people often have greater privileges and low level access. From a data visibility standpoint, test schemas have exactly the same data security requirements as production systems yet they usually contain far more relaxed controls.

In general, a reasonable security assumption is that the more people who have access to the information, the greater the inherent risk of the data being compromised. If, as is typical with test systems, it is not possible to restrict the number of people working on the data then a useful technique to provide enhanced security is to modify the data so that no sensitive information remains. The process of modifying the data to remove data sensitivity issues is known by a number of names – data masking, data sanitization or data scrubbing.

Irregardless of the name used, the general technique is to modify the existing data in such a way as to remove all identifiable and distinguishing characteristics thus rendering the data anonymous - yet still usable as a test system.

It is important to realise that in the case of Oracle Applications HR schemas, the data is intricately interlinked and highly denormalized. Each masking operation applied to the schema must preserve the relationships between the data storage entities. The methodology required is, in many cases, extremely subtle.

Sanitizing the Oracle Applications HR Schema

Overview

This paper discusses some of the specific actions required to sanitize the data in Oracle Applications HR schemas. No claim is made as to its completeness – in fact, it is highly probable given the version variations and typical site specific customisations, that data in tables and columns other than the ones mentioned here will require sanitization.

A practical implementation of a set of data sanitization operations will use a number of techniques and will require a variety of issues to be addressed. This document refers to these techniques and issues by names written in an italicised font such as “*Substitution*”, “*Variance*”, “*Table Internal Synchronization*” or “*The Isolated Case Phenomena*” etc. The names used are ours and were composed in order to provide a short descriptor for easy reference. As far as we know there is no widely agreed nomenclature.

No attempt is made herein to provide any overview regarding specific issues or techniques. For a comprehensive discussion of all italicised terms please see the companion white papers:

Data Sanitization Techniques

http://www.DataMasker.com/datasanitization_whitepaper.pdf

Data Scrambling Issues

<http://www.DataMasker.com/datascramblingissues.pdf>

Oracle Applications HR Tables Which Require Masking

Every organization has unique HR requirements. Hence there are considerable variations in the configuration of each Oracle HR implementation. In addition there are differences, both significant and minor, between the various versions of the Oracle HR schema. The tables discussed in this and the following section are current as of Oracle Applications version 11.5.8.

Some of the critical tables in an Oracle HR schema which contain user identifiable information are:

```
PER_ALL_PEOPLE_F  
PER_ALL_ASSIGNMENTS_F  
PER_PHONES  
PER_ADDRESSES  
PAY_EXTERNAL_ACCOUNTS  
HR_COMMENTS  
PER_ANALYSIS_CRITERIA
```

As a bare minimum, the data in the above tables will need to be sanitized. Usually other tables will also be required as well. The advice in the following section is only a suggestion as to a masking approach for some of the more tricky columns in the above tables. Doubtless there are other methodologies. Please be aware the discussion of the columns for the above tables is *not* complete! There are numerous important columns that have not been discussed for space reasons – for the most part their data sanitization requirements are pretty straightforward.

A careful analysis of each table and its contents will be required in order to completely mask the data. Typically, multiple operations are required for each table. The PER_ALL_PEOPLE_F table, for example, is a particularly complex case. Be sure to check the flex fields in the tables to see if they require masking. Since the usage of these columns are implementation defined, the only way to determine if they require sanitization is to look at them and find out what sort of data is in there.

Important Note

Needless to say, (but we are going to say it anyways), only mask rows in test instances. Even then, only mask schemas that you can recreate when necessary. In general, masking operations are not reversible - there is no “undo” other than a complete restore from backup.

The PER_ALL_PEOPLE_F Table

Overview

The PER_ALL_PEOPLE_F table holds personal information for just about everybody that comes into contact with the HR department. This includes employees, applicants, ex-employees, ex-applicants, contacts and usually other site specific categories of people.

In general, each distinct individual in the table is distinguished by an internal Oracle identifier called PERSON_ID which is unique to the individual. However, there is always more than one row per PERSON_ID in the table. The primary key is (PERSON_ID, EFFECTIVE_START_DATE, EFFECTIVE_END_DATE) and each time the user changes an assignment, a new row is added to PER_ALL_PEOPLE_F with a updated effective date range. The earlier rows for the PERSON_ID remain as a history of previous assignments. This means that if the LAST_NAME field is masked for one row, the same last name will be required in each row with an identical PERSON_ID. Getting this detail wrong really messes up some of the front end screens and reduces the functionality of the resulting test instance. This is an example of the *Table-Internal Synchronization* issue discussed in the [Data Scrambling Issues](#) white paper and it appears constantly in Oracle Applications HR schema tables.

Decisions

Due to its many synchronization issues we tend to use PER_ALL_PEOPLE_F as the driver. That is, we modify PER_ALL_PEOPLE_F and synchronize every other table to it. This means some decisions have to be made regarding the PER_ALL_PEOPLE_F table.

Do you wish to preserve gender in the PER_ALL_PEOPLE_F table? In other words, should a record listed with a gender of 'M' necessarily remain an 'M' in the masked schema. It's a judgement call that has to be made based on the distribution of data within the schema. Typically, if there is a reasonable distribution of male and female records, one does not bother with masking the gender – the other fields will be enough to sanitize things. However, if there is only one or very few records of one type then the genders might need to be masked to prevent the identification via the *Isolated Case Phenomena*.

Do you wish to mask the EMPLOYEE_NUMBER column? Usually this is the case since people know their own employee number and would readily be able to identify their own records and possibly those of others. If all other columns associated with the record are masked, disguising the EMPLOYEE_NUMBER may not be required. If this column is masked, be aware that the EMPLOYEE_NUMBER column requires both *Table-Internal Data Synchronization* internally and *Table-Table Data Synchronization* with other tables (see the discussion of the PER_ASSIGNMENTS table for an example).

Do you need to worry about the TITLE field? Many sites use the Oracle supplied standard titles such as Mr. Mrs. Ms. etc and there is no real requirement to mask these fields. However, if you only have one "Reverend" in the system then you have a classic *Isolated Case Phenomena* to worry about. There are a number of sites (police

forces, military and other para-military) that have a variety of titles, the quantity of which gets progressively fewer as the rank increases. For example, there is probably no need to mask the title “Constable” but the single “Chief Constable” record might get re-titled back to an already existing type, or simply just deleted, to prevent identification.

Specific actions on PER_ALL_PEOPLE_F Columns

PERSON_ID – This column forms part of the primary key. It is an internal Oracle identifier and is generally meaningless. It can be masked, but there will be *Table-Table Data Synchronization* issues with over 60 tables. These tables will not be listed here, but if you wish to have this information just execute the query below:

```
select table_name from user_tab_columns
where column_name='PERSON_ID' order by table_name;
```

SEX – Decide if this column is to be masked. If it is, then typically every record gets set to ‘M’ then a defined percentage (50% perhaps) is updated to an ‘F’ value. Typically the PREVIOUS_LAST_NAME field needs to be synchronized so that the majority of not null PREVIOUS_LAST_NAME columns are associated with ‘F’ entries since this mirrors reality. It may also be necessary to update other tables containing details (such as pregnancy leave taken) to correlate with the re-gendered rows.

EMPLOYEE_NUMBER – Usually this column is rendered anonymous to prevent simple lookups on known values. This is a varchar2 field and its structure is site defined – any replacement value must conform to the same formatting otherwise the *Intelligent Key* issue will manifest itself and many things in the resulting schema will break. Care must also be taken not to update these values to collide with existing ranges otherwise unrelated rows can become associated with each other. The EMPLOYEE_NUMBER will need to be synchronized *Table-Internal* and also *Table-Table* with the PER_ALL_ASSIGNMENTS_F table at minimum and usually others depending on the implementation of the system.

FIRST_NAME – This sensitive column needs *Row-Internal* synchronization with the SEX column so that ‘M’ records get male first names and ‘F’ records get female first names. Also requires *Table-Internal* synchronization with the other rows with the same PERSON_ID.

MIDDLE_NAMES – Always masked and usually needs *Row-Internal* synchronization with the SEX column so that ‘M’ records get male names and ‘F’ records get female names. Also requires *Table-Internal* synchronization with the other rows with the same PERSON_ID.

LAST_NAME – This column requires *Table-Internal* synchronization with the other rows with the same PERSON_ID.

KNOWN_AS – A sparsely populated column that is always sanitized. A useful way of approaching this column is to null all values and substitute in a random percentage. Usually needs *Row-Internal* synchronization with the SEX column so that ‘M’ records

get male first names and 'F' records get female first names. May require *Table-Internal* synchronization with the other rows with the same PERSON_ID depending on how picky you wish to be.

PREVIOUS_LAST_NAME – Always masked, is sparsely populated and usually contains the previous last name of married (or divorced) females. A useful way of approaching this column is to null all values and substitute in a random percentage. May require *Table-Internal* synchronization with the other rows with the same PERSON_ID.

NATIONAL_IDENTIFIER – Usually a governmental ID data item which must be rendered anonymous (for example: SSN in the USA and NI number in the UK). This column will require *Table-Internal* synchronization with the other rows. Masking these types of ID usually involves coping with the *Intelligent Key* problem discussed in the [Data Scrambling Issues](#) white paper.

EMAIL_ADDRESS – Needs to be cleansed in some manner. Probably should be updated with something random that looks like an email address.

DATE_OF_BIRTH – Often overlooked, but readily known and is a unique identifier in many cases. Probably should be masked, but take care not to put in values which are too old or too young. It is unlikely there are many 5 year olds on most HR systems and values which are out of range may well introduce validity issues on the front end screens. It is sometimes important to preserve the distribution of the information in this type of field. If the birth dates are replaced by a range of random dates then there will probably be an equal number of 40 year old employees as there is 20 year old employees. This may or may not matter to the end users of the test system – if it does matter then this is a case of a *Distribution Preservation Issue* and it is usually addressed using *Variance* techniques.

START_DATE – A possible candidate. Make sure that this date is not sanitized to be unreasonable given the DATE_OF_BIRTH. It makes the internal HR validity checks unhappy if people are employed before they are born. This field may also exhibit a case of the *Distribution Preservation Issue*.

TITLE – May or may not be required to be masked – see the discussion in the Decisions section above. Masked TITLE columns may have to be synchronized with the SEX field as appropriate and will require *Table-Internal* synchronization based on distinct PERSON_ID's.

FULL_NAME – A denormalized field requiring Row-Internal synchronization which contains the formatted contents of the LAST_NAME, FIRST_NAME, KNOWN_AS and TITLE fields. This data item must be built after the masking operations on the dependent fields are complete – see the discussion of the *Sequential Operations* issue in the [Data Scrambling Issues](#) white paper.

The PER_ALL_ASSIGNMENTS_F Table

Overview

The PER_ALL_ASSIGNMENTS_F table holds information regarding employee assignments. Employees must have at least one employee assignment at all times in a period of service, and each assignment must have a unique number. The ASSIGNMENT_ID field here is the same as the EMPLOYEE_NUMBER in the PER_ALL_PEOPLE_F table and needs to be synchronized with it. There are multiple assignments for any given PERSON_ID or EMPLOYEE_NUMBER so *Table-Internal Data Synchronization* is also called for.

Specific actions on PER_ALL_ASSIGNMENTS_F Columns

ASSIGNMENT_ID – If the EMPLOYEE_NUMBER column is masked in the PER_ALL_PEOPLE_F table then this value must also be masked. Each distinct PERSON_ID in the PER_ALL_PEOPLE_F table must have its EMPLOYEE_NUMBER match that of the ASSIGNMENT_ID for the same PERSON_ID in the PER_ALL_PEOPLE_F table.

The HR_COMMENTS Table

Overview

The HR_COMMENTS table contains a number of non-date dependent textual HR information.

Specific actions on HR_COMMENTS Columns

COMMENT_TEXT – Highly sensitive textual information associated with personnel records. Absolutely must be sanitized and exhibits the *Free Format Data* issue discussed in the [Data Scrambling Issues](#) white paper. A typical action is to null the entire column or replace it with random textual gibberish. Carefully hand sanitized realistic looking data can be substituted into selected records if required.

The PER_PHONES Table

Overview

The PER_PHONES table holds phone numbers for current and ex-employees and other contacts.

Specific actions on PER_PHONES Columns

PHONE_NUMBER – Highly sensitive information which really must be replaced with realistic looking random data.

The PER_ADDRESSES Table

Overview

The PER_ADDRESSES table holds address information for current and ex-employees and other contacts. Be aware that there are some issues regarding the treatment of the primary address – see the Oracle documentation for detailed information.

Specific actions on PER_ADDRESSES Columns

ADDRESS_LINE_[1, 2, 3] – This is the employee address and is highly sensitive. Typically the first line is given a realistic looking random street address and the remainder set to null unless there is a requirement for multi-line street addresses.

TOWN_OR_CITY – It is much more useful to the end users if this column can be set from a list of town or city names rather than just random text.

REGION_[1, 2, 3] – These are usually information such as a state or county designator. As with the street address group of columns, the first value is given a realistic looking state name and the remainder set to null.

COUNTRY – It is debatable whether this column needs to be masked and is probably a decision that can be taken at implementation time. If this column is masked, be aware that the value used is required by the front end screens to be present in a pre-approved list. Usually this field is updated to a common value to eliminate all occurrences of the *Isolated Case Phenomena*.

POSTAL_CODE – This field needs to be masked and is usually an *Intelligent Key*. This means any replacement data must satisfy the validity checks or the front end screens will work improperly.

TELEPHONE_NUMBER_[1, 2, 3] – These are the telephone numbers of the employee and as such are highly sensitive. Typically the first line is given a realistic looking phone number and the rest are given null values.

The PAY_EXTERNAL_ACCOUNTS Table

Overview

The PAY_EXTERNAL_ACCOUNTS table stores bank account information for employees. All bank account details are stored in the flex fields so check these very carefully.

Specific actions on PAY_EXTERNAL_ACCOUNTS Columns

SEGMENT_[1...20] – Highly sensitive information which really must be replaced with realistic looking random data.

Summary

Given the legal and organizational operating environment of today, many test and development Oracle Applications HR databases will require some form of sanitization in order to render the information content anonymous.

There are a variety of techniques available, and an even larger number of issues of which to be aware. Some of the most critical issues from an Oracle Applications HR schema perspective are the *Row-Internal*, *Table-Internal* and *Table-Table Data Synchronization* requirements.

The following companion white papers provide a detailed discussion of data masking techniques and issues:

Data Sanitization Techniques

http://www.DataMasker.com/datasanitization_whitepaper.pdf

Data Scrambling Issues

<http://www.DataMasker.com/datascrumblingissues.pdf>

The demands of the Oracle HR schema require a sophisticated approach to the problem. In this paper the `PER_ALL_PEOPLE_F` table is the focus for the majority of the really complex masking requirements. A number of decisions for the `PER_ALL_PEOPLE_F` table were discussed and it was noted that the outcome of these decisions has a major effect on the type of sanitization performed.

Some of the other important tables in the Oracle HR schema (from a data sanitization point of view) were listed and a discussion of how masking operations might be performed on selected columns from these tables was undertaken.