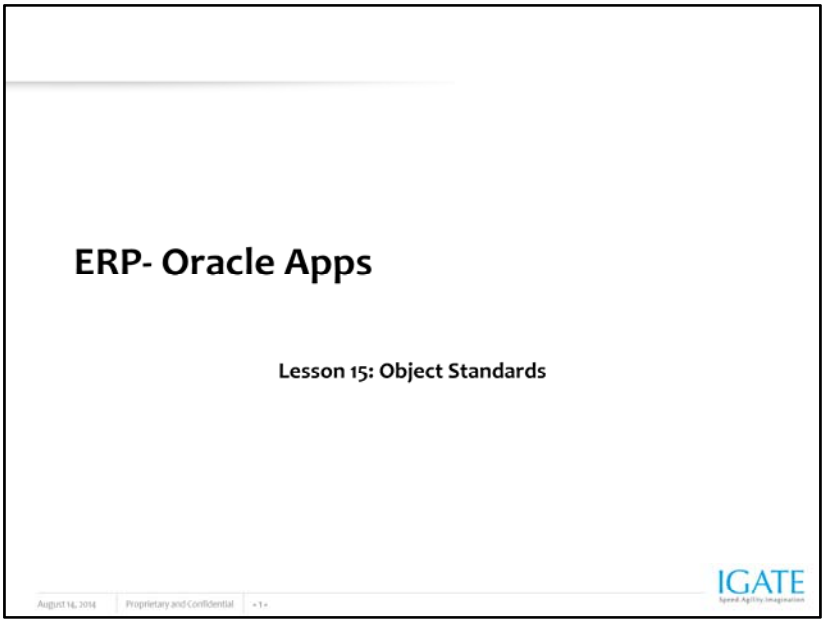
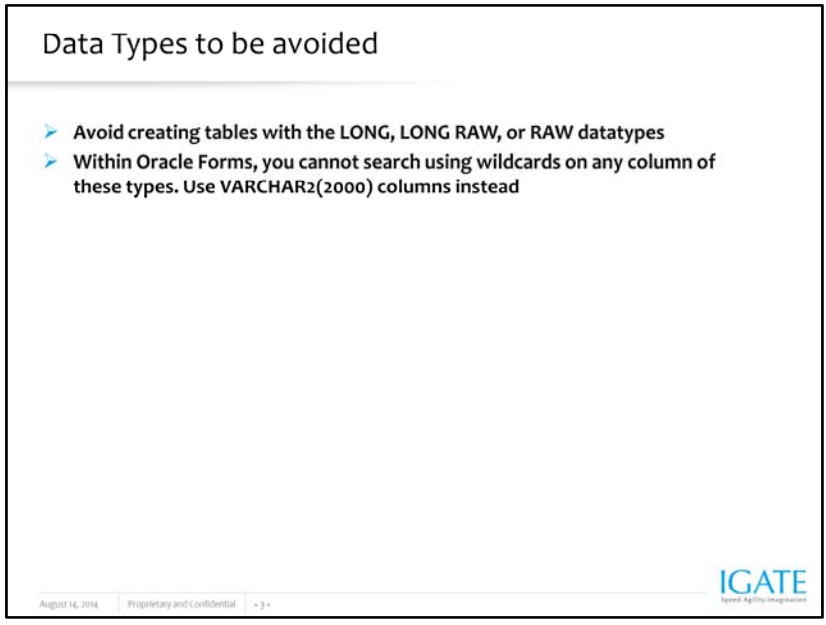
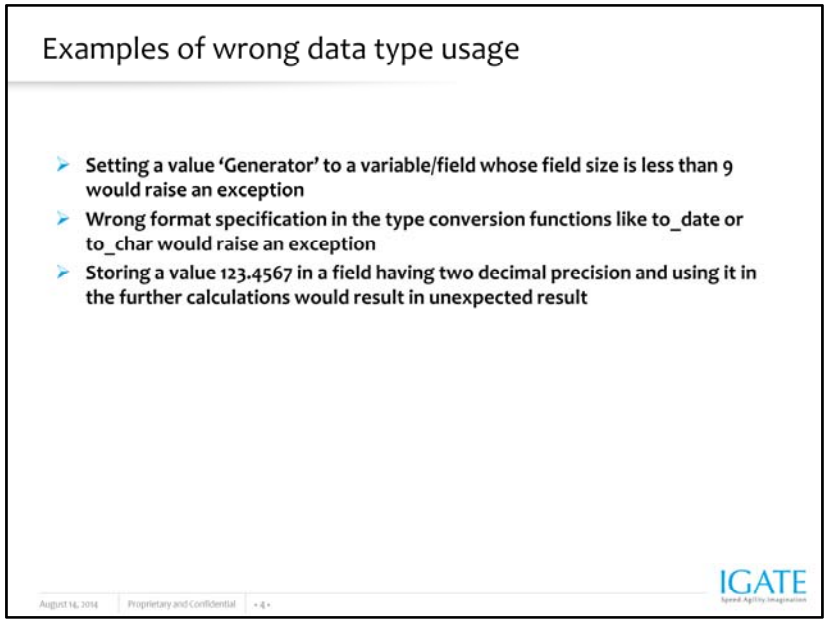
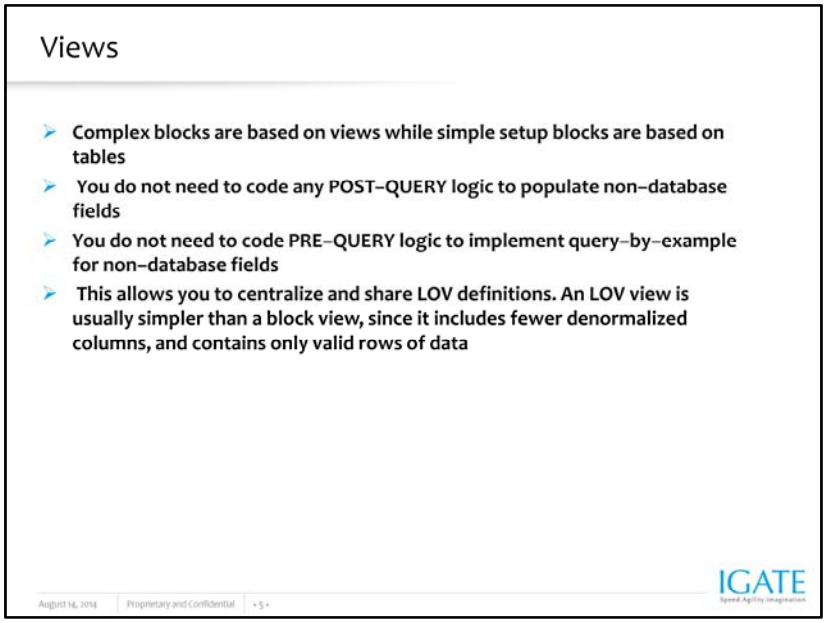

ERP- Oracle Apps
Lesson 15: Object Standards
August 14, 2014
Proprietary and Confidential
- 1 -
IGATE

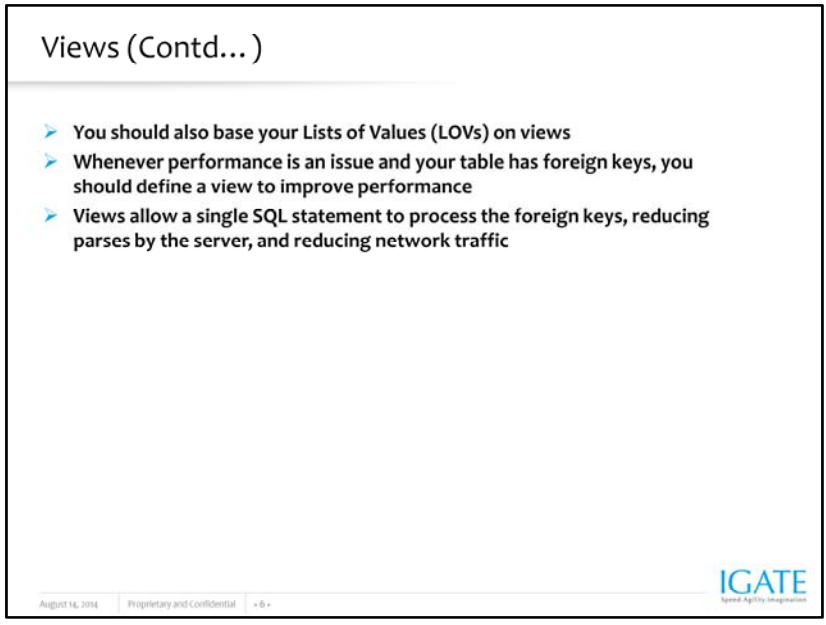
Lesson Objectives

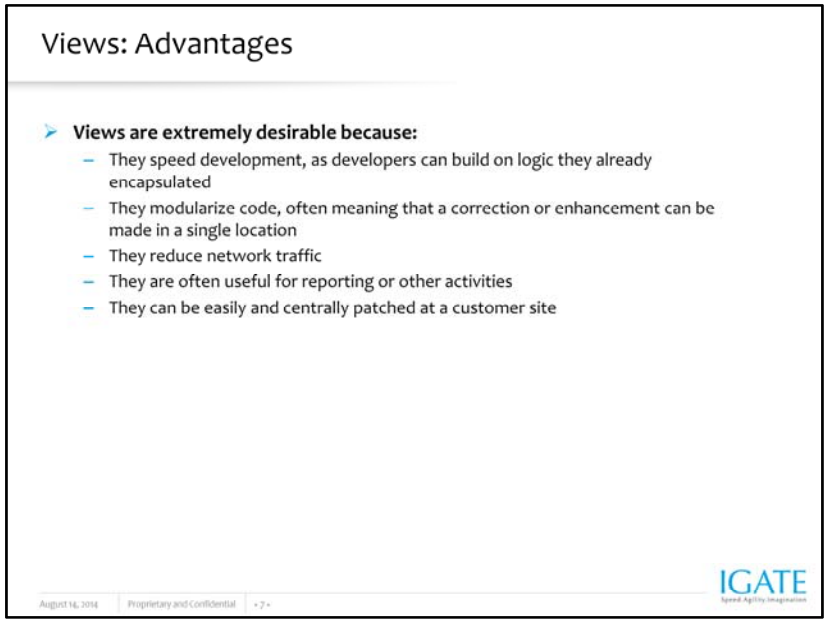
- **At the end of the session you will be able to:**
  - Understand the correct usage of data types
  - Work with Oracle Applications Views
  - Sequences
  - Understand Table Registration
  - Know What are WHO columns

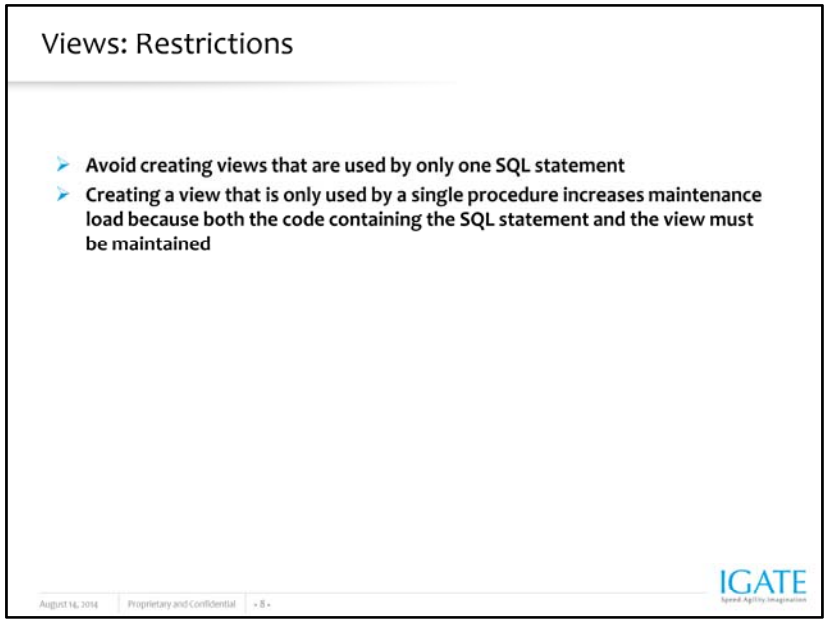

Data Types to be avoided
Avoid creating tables with the LONG, LONG RAW, or RAW datatypes
Within Oracle Forms, you cannot search using wildcards on any column of these types. Use VARCHAR2(2000) columns instead
August 14, 2014
Proprietary and Confidential
IGATE

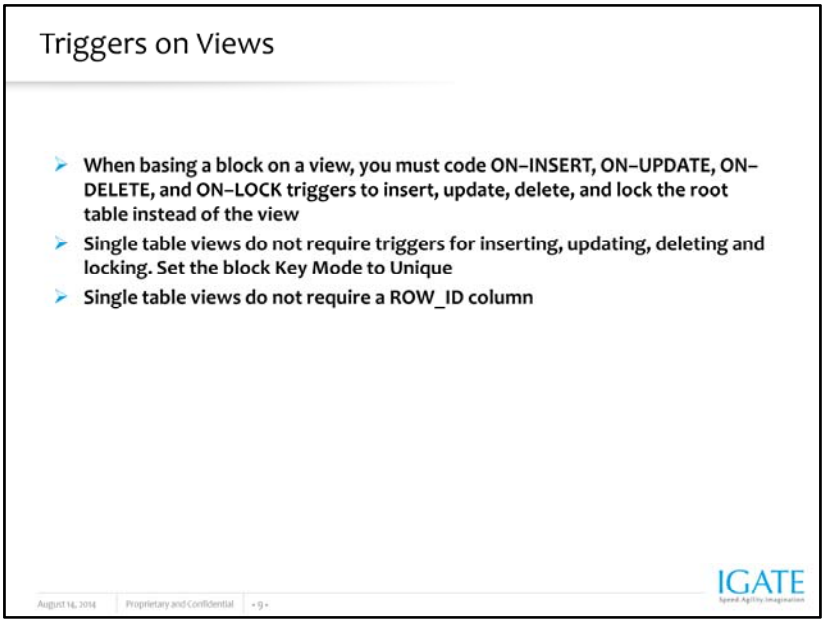

Examples of wrong data type usage
Setting a value 'Generator' to a variable/field whose field size is less than 9 would raise an exception
Wrong format specification in the type conversion functions like to_date or to_char would raise an exception
Storing a value 123.4567 in a field having two decimal precision and using it in the further calculations would result in unexpected result
August 14, 2014
Proprietary and Confidential
IGATE


Views
Complex blocks are based on views while simple setup blocks are based on tables
You do not need to code any POST–QUERY logic to populate non–database fields
You do not need to code PRE–QUERY logic to implement query–by–example for non–database fields
This allows you to centralize and share LOV definitions. An LOV view is usually simpler than a block view, since it includes fewer denormalized columns, and contains only valid rows of data
August 14, 2014
Proprietary and Confidential
- 5 -
IGATE


Views (Contd…)
You should also base your Lists of Values (LOVs) on views
Whenever performance is an issue and your table has foreign keys, you should define a view to improve performance
Views allow a single SQL statement to process the foreign keys, reducing parses by the server, and reducing network traffic
August 14, 2014
Proprietary and Confidential
IGATE


Views: Advantages
Views are extremely desirable because:
They speed development, as developers can build on logic they already encapsulated
They modularize code, often meaning that a correction or enhancement can be made in a single location
They reduce network traffic
They are often useful for reporting or other activities
They can be easily and centrally patched at a customer site
August 14, 2014
Proprietary and Confidential
- 7 -
IGATE


Views: Restrictions
Avoid creating views that are used by only one SQL statement
Creating a view that is only used by a single procedure increases maintenance load because both the code containing the SQL statement and the view must be maintained
August 14, 2014
Proprietary and Confidential
- 8 -
IGATE



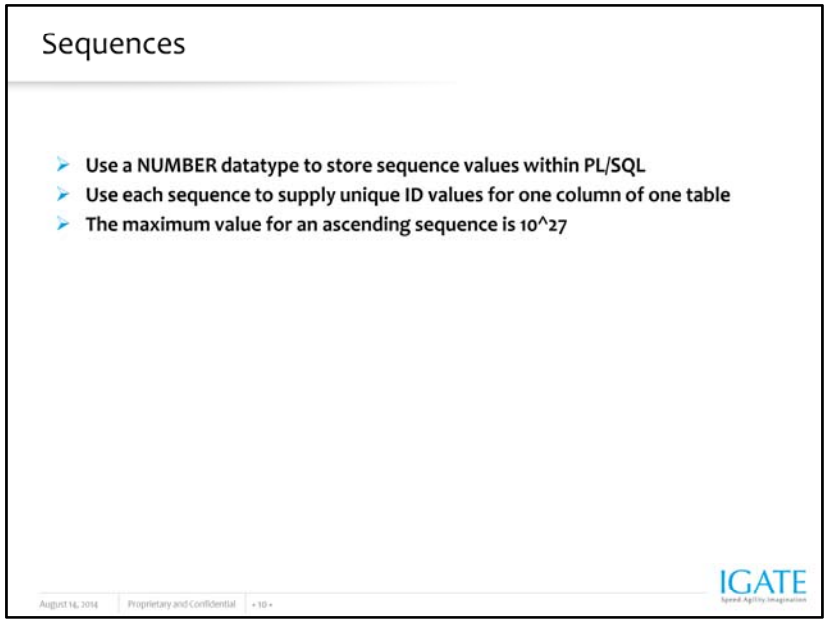
Your view should then include all of the columns in the root table, including the WHO columns, and denormalized foreign key information.

Suggestion: You only need to include the ROWID column if an Oracle Forms block is based on this view. The Oracle Forms field corresponding to the ROW_ID pseudo–column should use the ROW_ID property class.

Change Block Key Mode
In Oracle Forms, you need to change the block Key Mode property to Non–Updatable to turn off Oracle Forms default ROWID references for blocks based on views. Specify the primary keys for your view by setting the item level property Primary Key to True.

For example, a view based on the EMP table has the columns ROW_ID, EMPNO, ENAME, DEPTNO, and DNAME. Set the Key Mode property of block EMP_V to Non–Updatable, and set the Primary Key property of EMPNO to True.

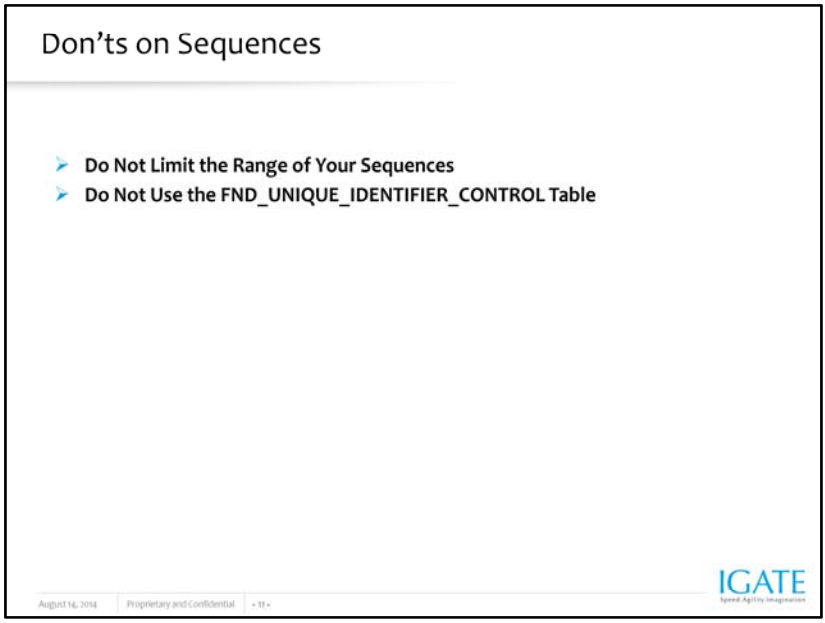
If your block is based on a table, the block Key Mode should be Unique.



Sequences can be created through front end of Apps or at the database level. Sequences created using the fron end can be assigned to a particular field on the form.
Seeded sequences are also available.

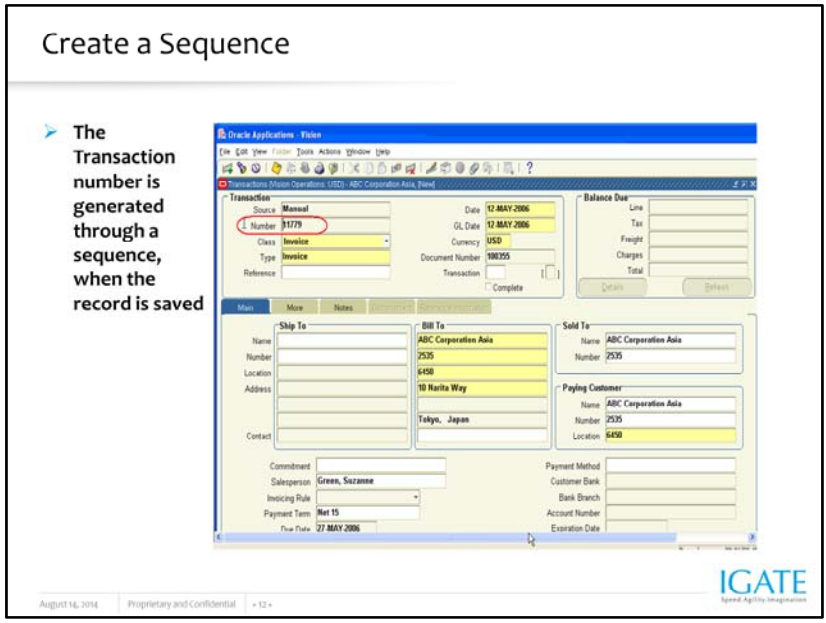
The purpose of backend sequences would be to provide unique ids to certain fields, if they are not part of the source data.

For more details, refer to "Sequences" (See Page 3-10) from Oracle Applications Developer's Guide.



Do not create sequences that wrap using the CYCLE option or that have a specified MAXVALUE. The total range of sequences is so great that the upper limits realistically are never encountered. In general, do not design sequences that wrap or have limited ranges.

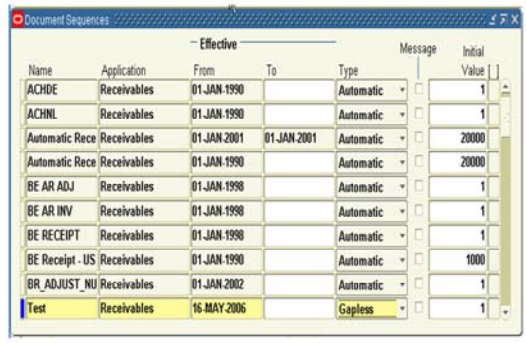
Do not rely on the FND_UNIQUE_IDENTIFIER_CONTROL table to supply sequential values. Use a sequence or the sequential numbering package instead. The FND_UNIQUE_IDENTIFIER_CONTROL table is obsolete and should not have any rows for objects in your product. Additionally, do not create application–specific versions of the FND table to replace the FND_UNIQUE_IDENTIFIER_CONTROL table.



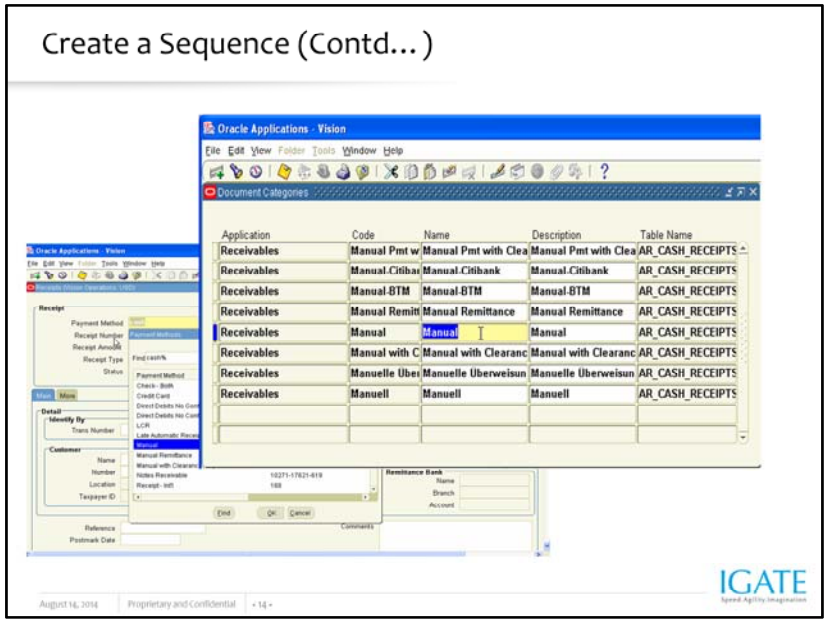
To navigate to the form shown above, select Receivables, Vision Operations (USA) -> Transactions -> Transactions. The Transaction number is generated through a sequence.

Create a Sequence (Contd...)

- Shown in the figure are existing sequences.
- You can also create a new sequence
- Navigate to System Administrator -> Document-> Define



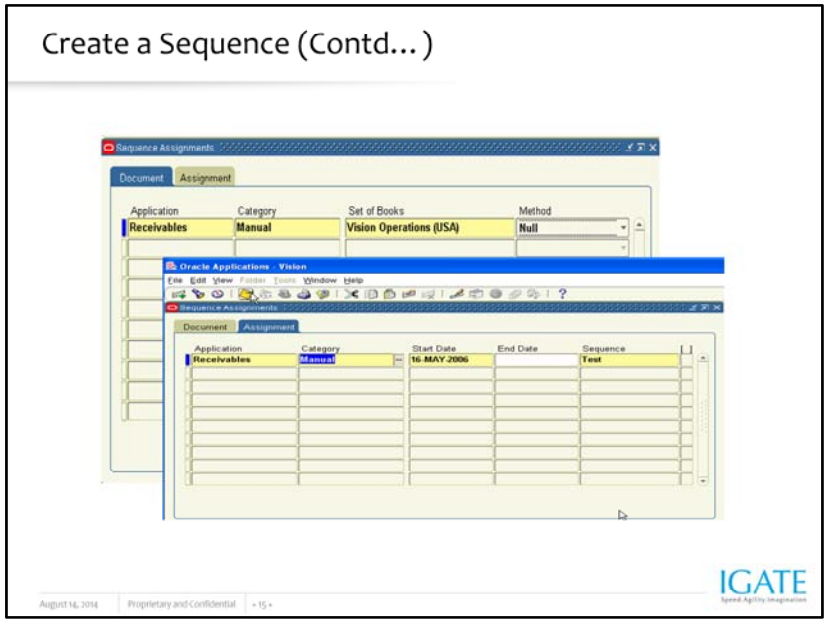
A new sequence Test is created and its type is set to Gapless, i.e. which indicates that there would be no gaps in the sequence numbers.



The window on the top can be accessed by selecting System Administrator -> Document-> Categories. This window displays the categories. Manual payment is one such category, associated with Application : Receivables and Table name : AR_CASH_RECEIPTS_ALL

Various payment Methods (Categories) are as shown in the Receipts form on the left. The Receipts form can be accessed by selecting : Receivables, Vision Operations (USA) -> Receipts.

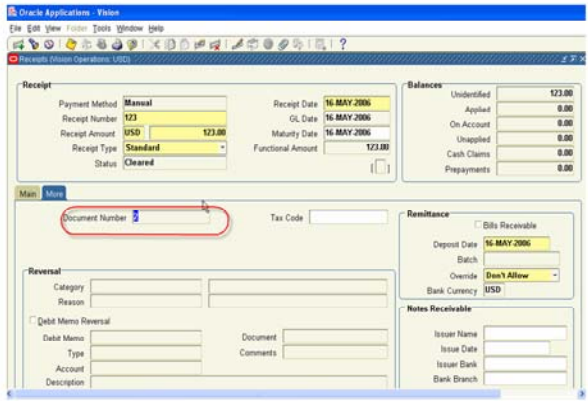
The sequence created is created for manual type payments.



The sequence is assigned to the Manual category in the Sequence Assignments form. This can be accessed by selecting System Administrator -> Document-> Sequence Assignments.

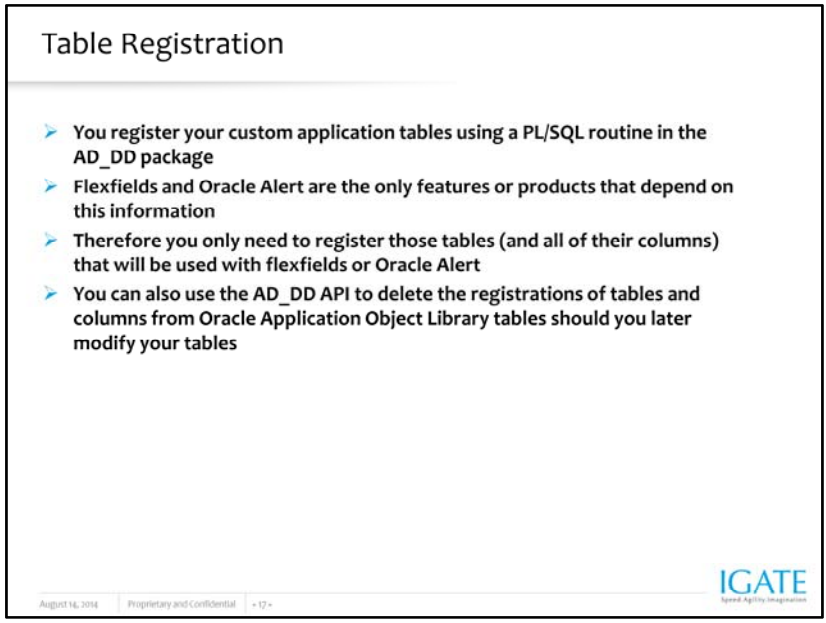
Note that initially the value under Method is set to Null, indicating that no sequence is yet assigned. To assign a sequence, click on the Assignment

Create a Sequence (Contd...)



IGATE
Speed. Agility. Imagination

In the Receipts form, when a record is created for Manual type of payment, the Document Number is automatically generated through the sequence.



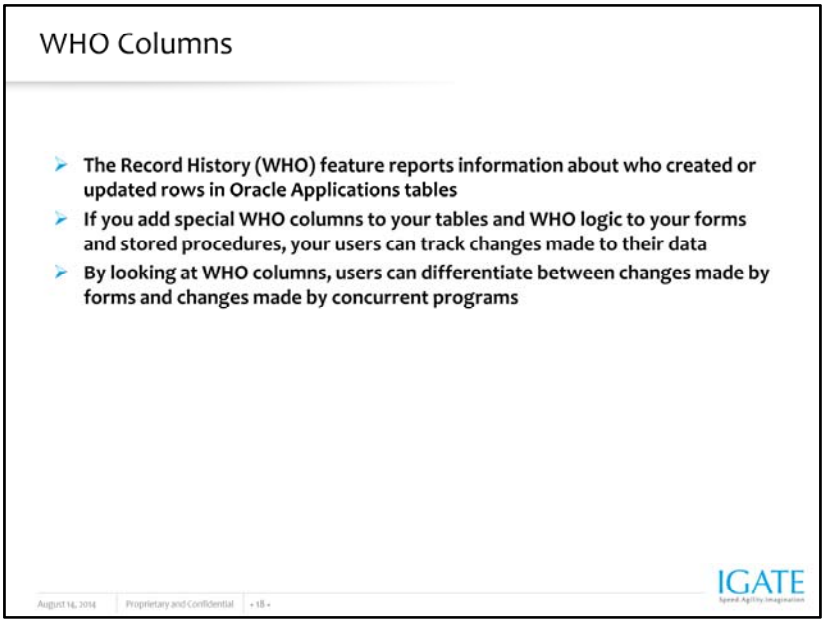
Sample Procedures in the Package

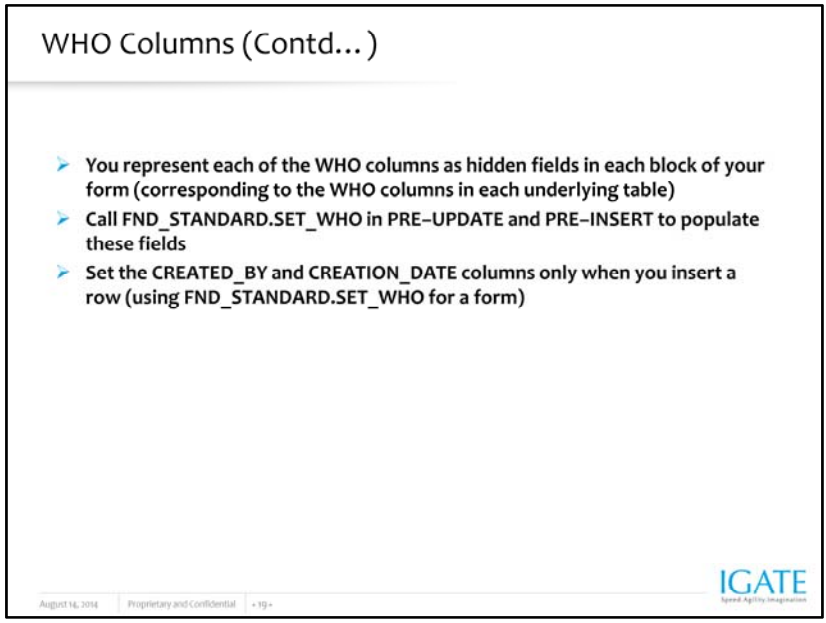
```
procedure register_table (p_appl_short_name in varchar2,
p_tab_name in varchar2,
p_tab_type in varchar2,
p_next_extent in number default 512,
p_pct_free in number default 10,
p_pct_used in number default 70);
```

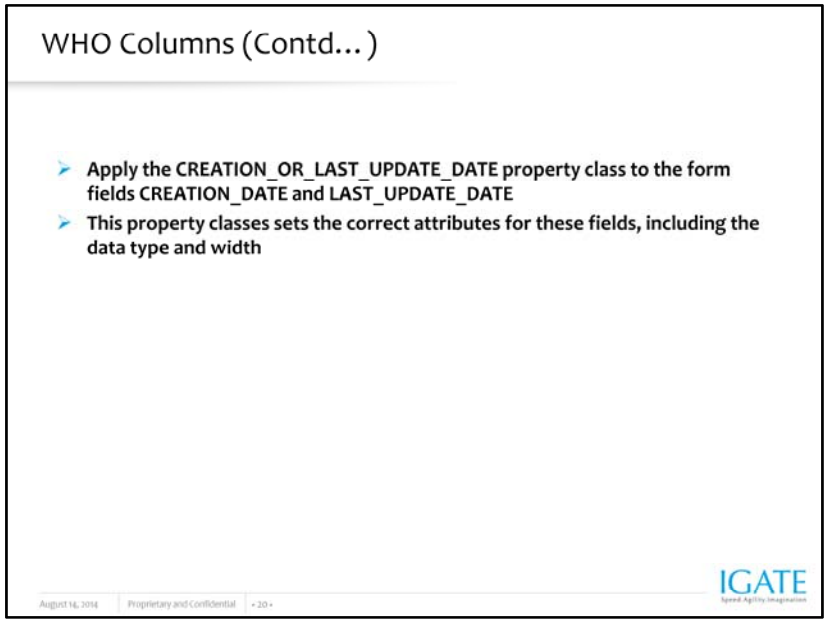
```
procedure register_column (p_appl_short_name in
varchar2,
p_tab_name in varchar2,
p_col_name in varchar2,
p_col_seq in number,
p_col_type in varchar2,
p_col_width in number,
p_nullable in varchar2,
p_translate in varchar2);
```

```
EXECUTE ad_dd.register_table('FND',
                             'CUST_FLEX_TEST', 'T', 8, 10, 90);
```

Here is an example of using the AD_DD package to register a flexfield table and its columns:

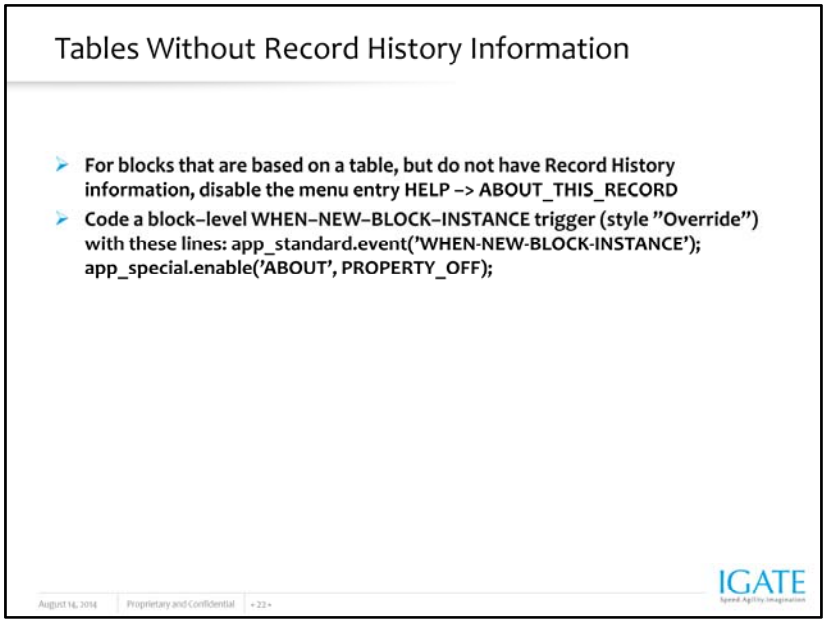

WHO Columns
The Record History (WHO) feature reports information about who created or updated rows in Oracle Applications tables
If you add special WHO columns to your tables and WHO logic to your forms and stored procedures, your users can track changes made to their data
By looking at WHO columns, users can differentiate between changes made by forms and changes made by concurrent programs
August 14, 2014
Proprietary and Confidential
IGATE


WHO Columns (Contd...)
You represent each of the WHO columns as hidden fields in each block of your form (corresponding to the WHO columns in each underlying table)
Call FND_STANDARD.SET_WHO in PRE-UPDATE and PRE-INSERT to populate these fields
Set the CREATED_BY and CREATION_DATE columns only when you insert a row (using FND_STANDARD.SET_WHO for a form)
August 14, 2014
Proprietary and Confidential
IGATE



For more details, refer to "Tracking Data Changes with Record History (WHO)" (See Page 3-2) from Oracle Applications Developer's Guide.

Record History



Use the APP_SPECIAL package to enable and customize menu entries and buttons on the toolbar. You can use the APP_SPECIAL.ENABLE procedure to dynamically control menu items, if the behavior you need is not provided automatically. First, determine if the default menu control handles the menu item in question, and ensure that there really is a need to override the default behaviors.
If the menu item is not controlled by the default menu control, use any appropriate trigger (typically PRE–BLOCK or WHEN–NEW–BLOCK–INSTANCE), adding the code:

Turn the menu item back on when you leave (typically POST–BLOCK) by calling:

```
app_special.enable('the menu item', PROPERTY_OFF|ON);
```

Include the full name of the menu item in this call, for example:

```
app_special.enable('the menu item', PROPERTY_ON|OFF);
```

You can determine the full names of the menu items by copying FNDMENU from the AU_TOP/resource/<language> area and opening the copy to examine the menu items.

```
app_special.enable('CLEAR.FIELD', PROPERTY_OFF);
```

If the menu item is controlled by the default menu control and you want to modify its behavior (enable or disable it), create the field– or block–level trigger listed (either WHEN–NEW–BLOCK–INSTANCE, WHEN–NEW–RECORD–INSTANCE, or WHEN–NEW–ITEM–INSTANCE). Set the trigger Execution Hierarchy to ”Override” and add the following code:

```
app_standard.event('TRIGGER_NAME');
app_special.enable('Menu_item',
PROPERTY_OFF|ON);
```

The item will be correctly reset in other blocks by the default menu control, so it is not necessary to reset it when leaving the block, record, or item.

Summary

- **In this session we covered:**
  - The Correct Usage of Data Types
  - Oracle Applications Views
  - Sequences
  - Table Registration Process
  - What are WHO columns

Review Question

- **Question 3: Whenever SQL*Loader encounters a database error while trying to load a record, it writes that record to a file known as the _____**
  - Option 1: Control file
  - Option 2: Flat file
  - Option 3: Bad file
  - Option 4: Discard file

- **Question 4: Records that do not meet the specified criteria are not loaded and are instead written to a file known as _____**
  - Option 1: Control file
  - Option 2: Flat file
  - Option 3: Bad file
  - Option 4: Discard file

IGATE
Speed. Agility. Imagination