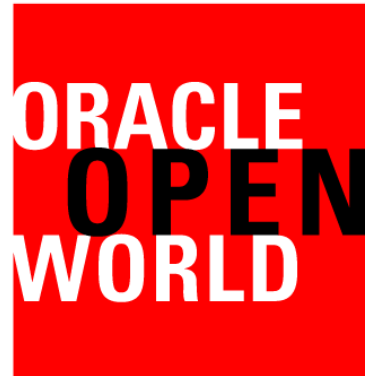


ORACLE®



Oracle OpenWorld

Latin America 2010

December 7–9, 2010



Oracle OpenWorld
Beijing 2010

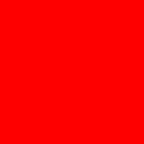
December 14–16, 2010



ORACLE®

Online Application Upgrade of Oracle's Bug DB with Edition-Based Redefinition

Louise Morin, Principal Member of Technical Staff
Bryn Llewellyn, Product Manager
Database Server Technologies Division, Oracle HQ



The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Program Agenda

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- Bugdb application characteristics
- Steps to EBR readiness
- EBR exercise
- Testing
- Summary
- References
- Questions and Answers





Larry
CEO, Oracle

“Eat your own dog food.”
“Drink your own champagne.”

Program Agenda

- Introduction
- **Challenge of online application upgrade**
- Brief EBR overview
- Bugdb application characteristics
- Steps to EBR readiness
- EBR exercise
- Testing
- Summary
- References
- Questions and Answers



Online application upgrade

the challenge

- The database holds data!
 - Upgrades change the representation of some of the data
 - Most of it is untouched
 - All of it must survive!
- There's code in the database
 - Synonyms, views, PL/SQL
 - You change this with *create or replace*
- Online users of the old app must experience no changes
 - Must make changes “in secret”
- Need old and new in concurrent use

Program Agenda

- Introduction
- Challenge of online application upgrade
- **Brief EBR overview**
- Bugdb application characteristics
- Steps to EBR readiness
- EBR exercise
- Testing
- Summary
- References
- Questions and Answers



Edition-based redefinition

the solution

- The *edition*
 - A database has one or many editions
 - Different editions can hold differently defined occurrences of the same object
- The *editioning view*
 - Tables hold the union of data needed by old and new tables are *not* editionable
 - An editioning view projects a table differently into each edition views, synonyms, and PL/SQL objects *are* editionable
- The *crossedition trigger*
 - A forward crossedition trigger synchronizes from old to new
 - A reverse crossedition trigger synchronizes from new to old

In place / Conflicts prevented

- You need only your production database
 - People have tried to buy “in secret” with a second database
- Conflicts are detected and prevented before they happen
 - Concurrent users of the old and new apps might attempt conflicting changes
 - A two database approach cannot synchronize transactionally (too slow) – so need human conflict resolution
- These benefits of EBR were stated *a priori* as *MUST* requirements

The NE on E prohibition

- There's no edition-extended syntax
 - One object refers to another, as ever, using *Owner.Name*
 - An editioned object can see only those other editioned objects that are in its own edition
 - An editioned object sees noneditioned objects ordinarily
- Therefore, a noneditioned object cannot depend on an editioned object
 - There's no way for a noneditioned object to say in which edition it should look for the editioned object it wants to depend on

The NE on E prohibition – *cont*

- A table can have a column defined by a UDT
 - The *Employees* table is not editioned
 - The *Phone_Numbers* column depends on the *Phone_Numbers_Type* (nested table)
 - A user-defined type (collection or ADT) is a PL/SQL object
 - Sounds like a gotcha
- Editions-enabling a user
 - An object whose type is noneditionable is never editioned
 - All objects whose type is editionable are editioned when their owner is editions-enabled
 - All objects whose type is editionable are not editioned when their owner is not editions-enabled

Readying for EBR

- Existing objects
 - Decide which objects should be editioned
 - Identify the *NE on E prohibition* breakers
 - “Move” these to owners who won’t be editions-enabled
 - Migrate the data they define
 - Now you can editions-enable the users you wanted to
- Cover existing tables with new editioning views
 - Rename tables to obscure names
 - Restore the former names with covering EVs
 - “Move” grants, triggers, VPD policies from table to EV
- Congratulations!
This was your last offline application upgrade

Case study

- The HR sample schema, that ships with Oracle Database, represents phone numbers in a single column:
 - Diana Lorentz 590.423.5567
 - John Russell 011.44.1344.429268
- Users now need to ring phone numbers from any country in the world
- So we want a uniform representation with two columns:
Country Code; and Number Within Country

Case study

ct 1_Show_20_Rows_Repeat_Pre_Upgrade

First Name	Last Name	Phone
Steven	King	011.32.242.647.4719
Neena	Kochhar	708.108.8233
Lex	De Haan	205.621.9819
Alexander	Hunold	011.38.209.317.1291
Bruce	Ernst	431.800.6569
David	Austin	
Valli	Pataba	
Diana	Lorent	
Nancy	Greenb	
Daniel	Faviet	
John	Chen	
Ismael	Sciarr	
Jose Manuel	Urman	
Luis	Popp	
Den	Raphae	
Alexander	Khoo	
Shelli	Baida	
Sigal	Tobias	
Guy	Himuro	
Karen	Colmen	

ct 2_Show_20_Rows_Repeat_Post_Upgrade

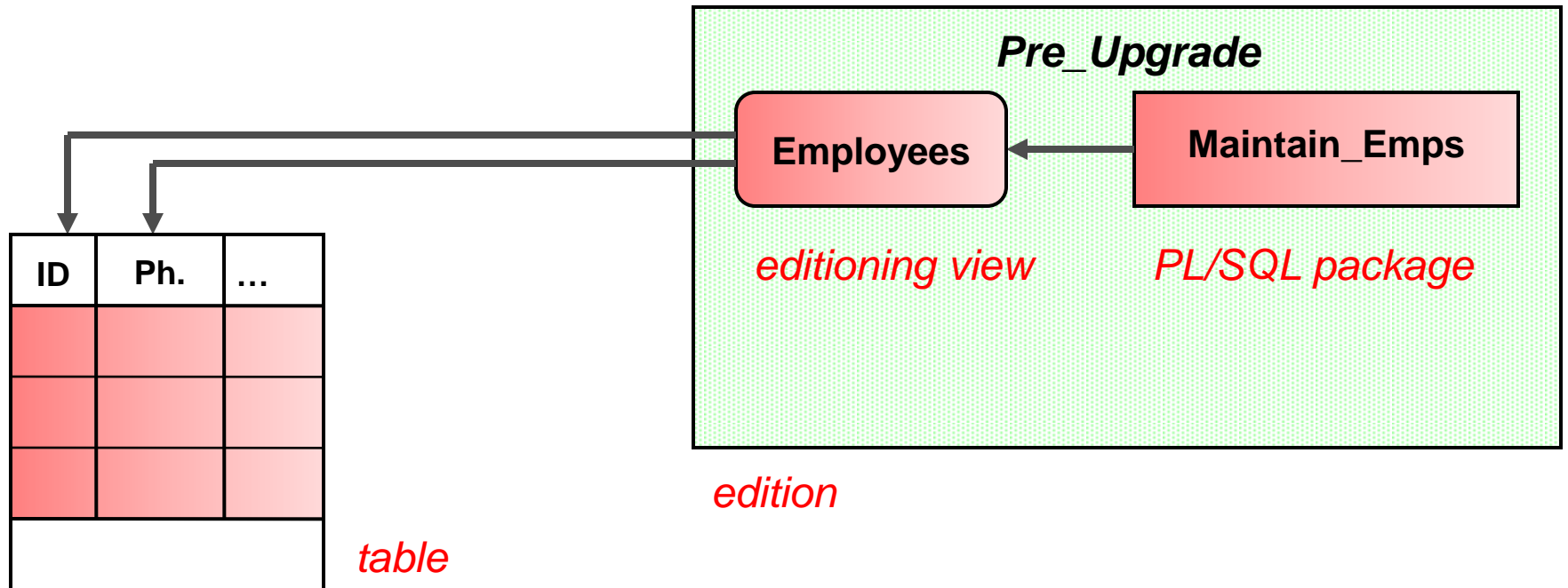
First Name	Last Name	Cntry #
Steven	King	+32 242-647-4719
Neena	Kochhar	+1 708-108-8233
Lex	De Haan	+1 205-621-9819
Alexander	Hunold	+38 209-317-1291
Bruce	Ernst	+1 431-800-6569
David	Austin	+1 207-718-4492
Valli	Pataballa	+45 662-851-6340
Diana	Lorentz	+47 951-260-7204
Nancy	Greenberg	+1 434-531-4024
Daniel	Faviet	+1 665-985-7057
John	Chen	+1 189-665-7741
Ismael	Sciarra	+1 235-670-4647
Jose Manuel	Urman	+1 782-701-7504
Luis	Popp	+40 224-526-8013
Den	Raphaely	+35 385-817-3247
Alexander	Khoo	+1 754-359-1977
Shelli	Baida	+36 949-238-6312
Sigal	Tobias	+40 440-870-9983
Guy	Himuro	+1 550-189-7906
Karen	Colmenares	+1 411-573-4711

The edition-based redefinition exercise proper

The app has already been EBR-readied



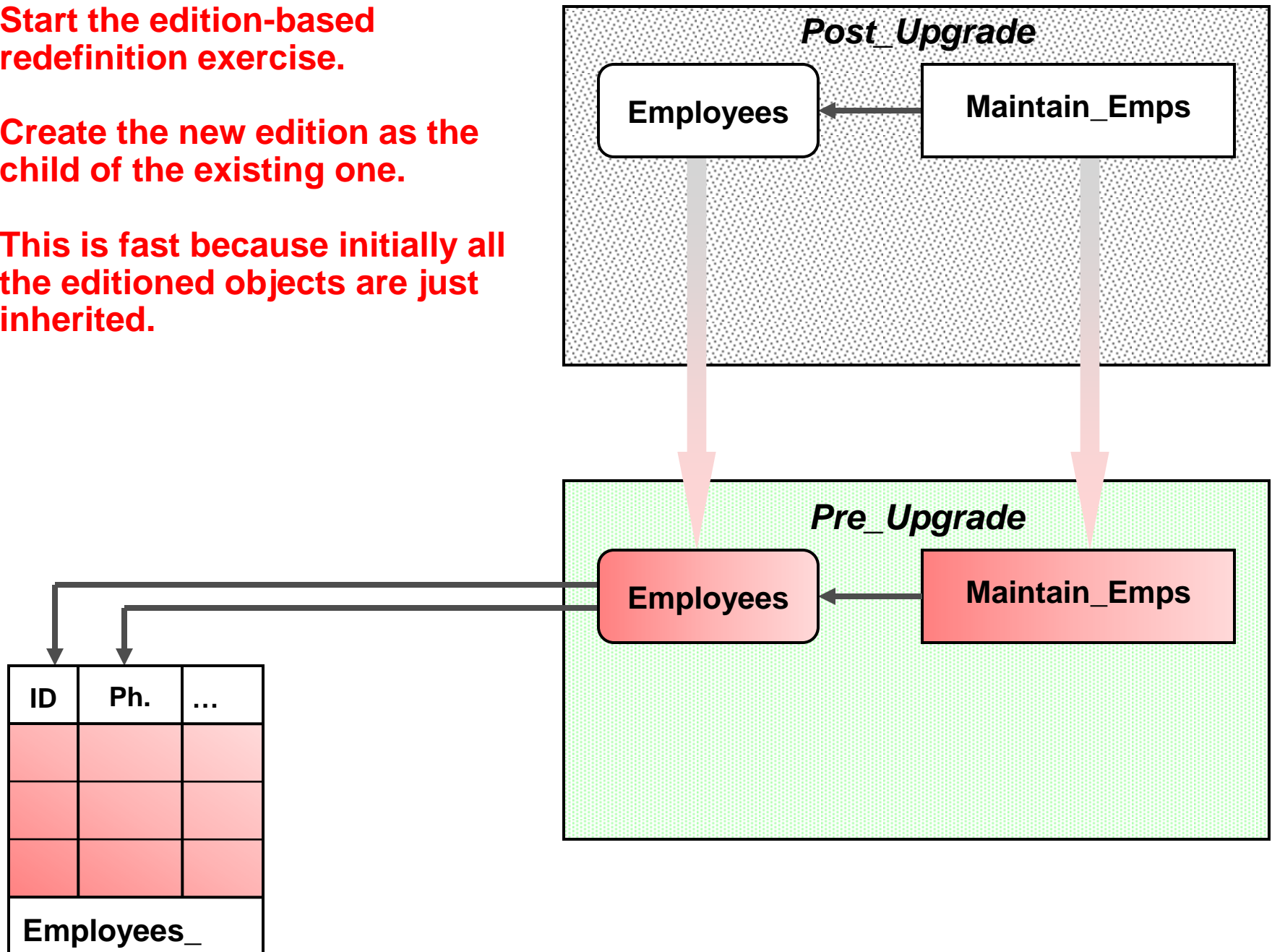
Starting point.
Pre-upgrade app in normal use.



Start the edition-based redefinition exercise.

Create the new edition as the child of the existing one.

This is fast because initially all the editioned objects are just inherited.

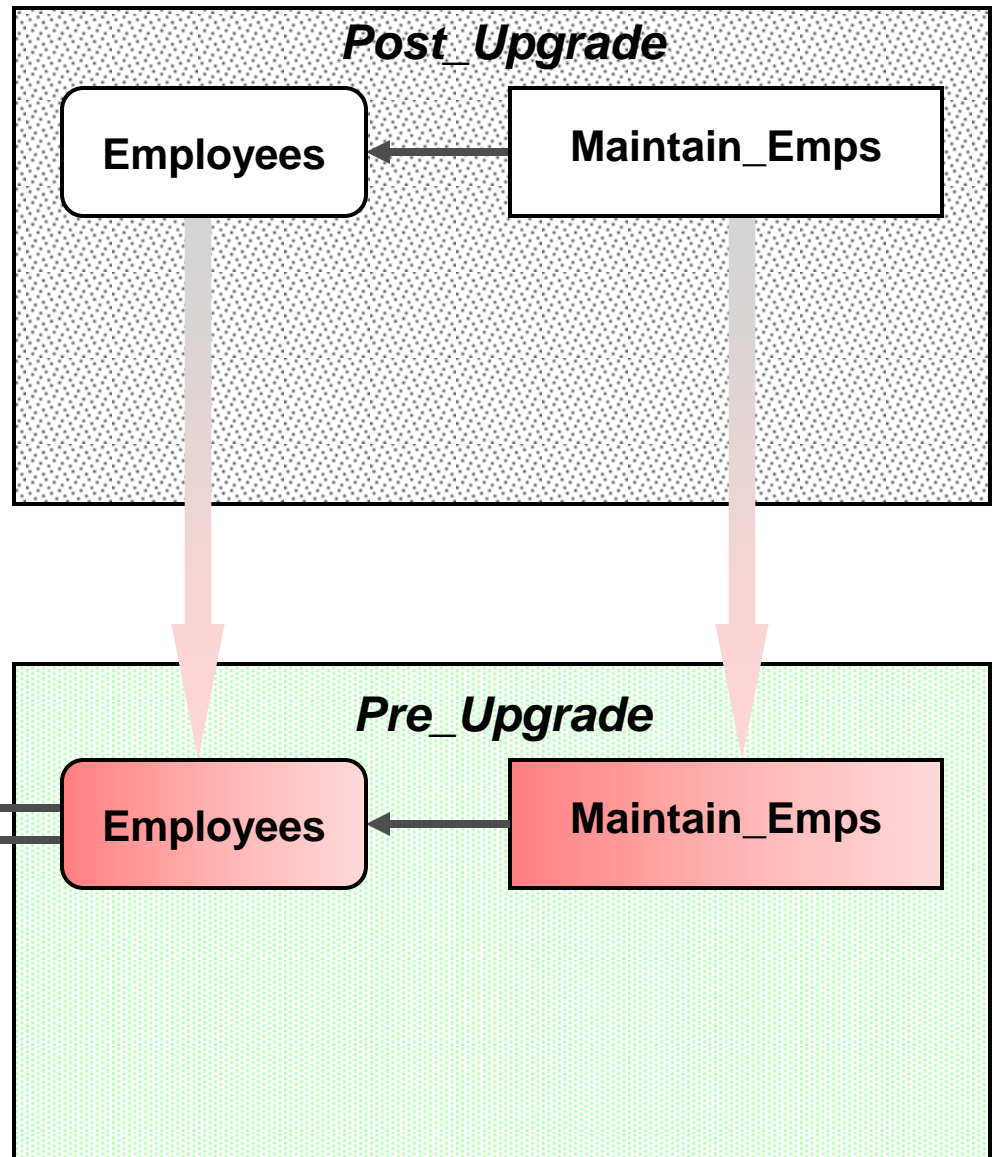


Create the replacement columns in the underlying table.

The editioning view shields the app from this change.

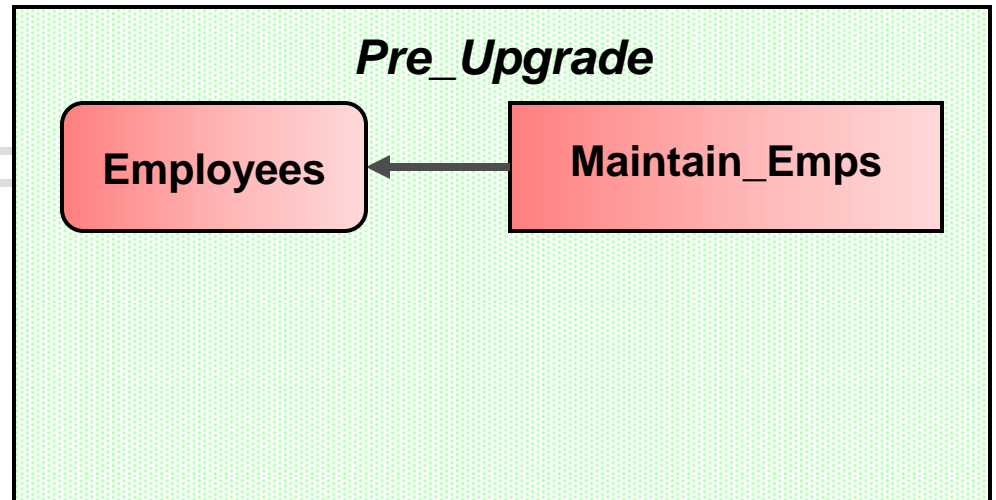
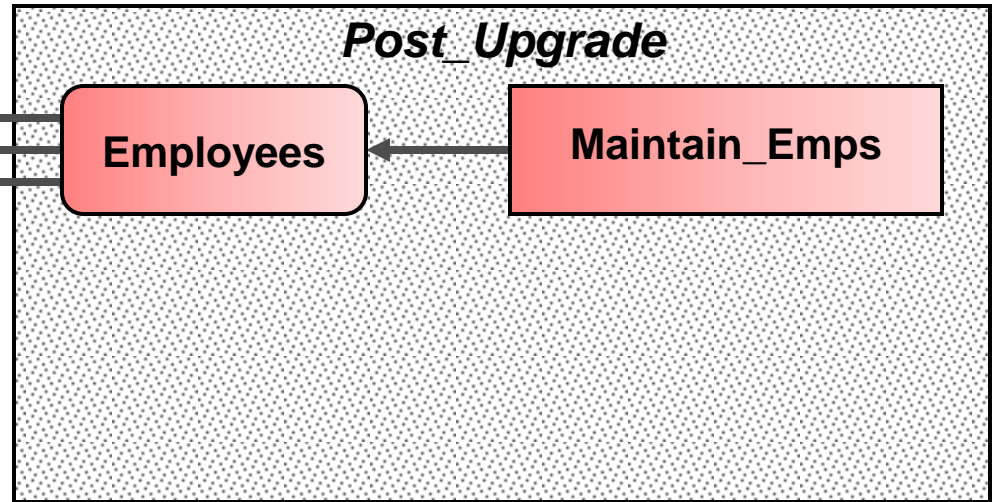
ID	Ph.	...	Cntry	#

Employees_



Change *Employees* to select the new columns.

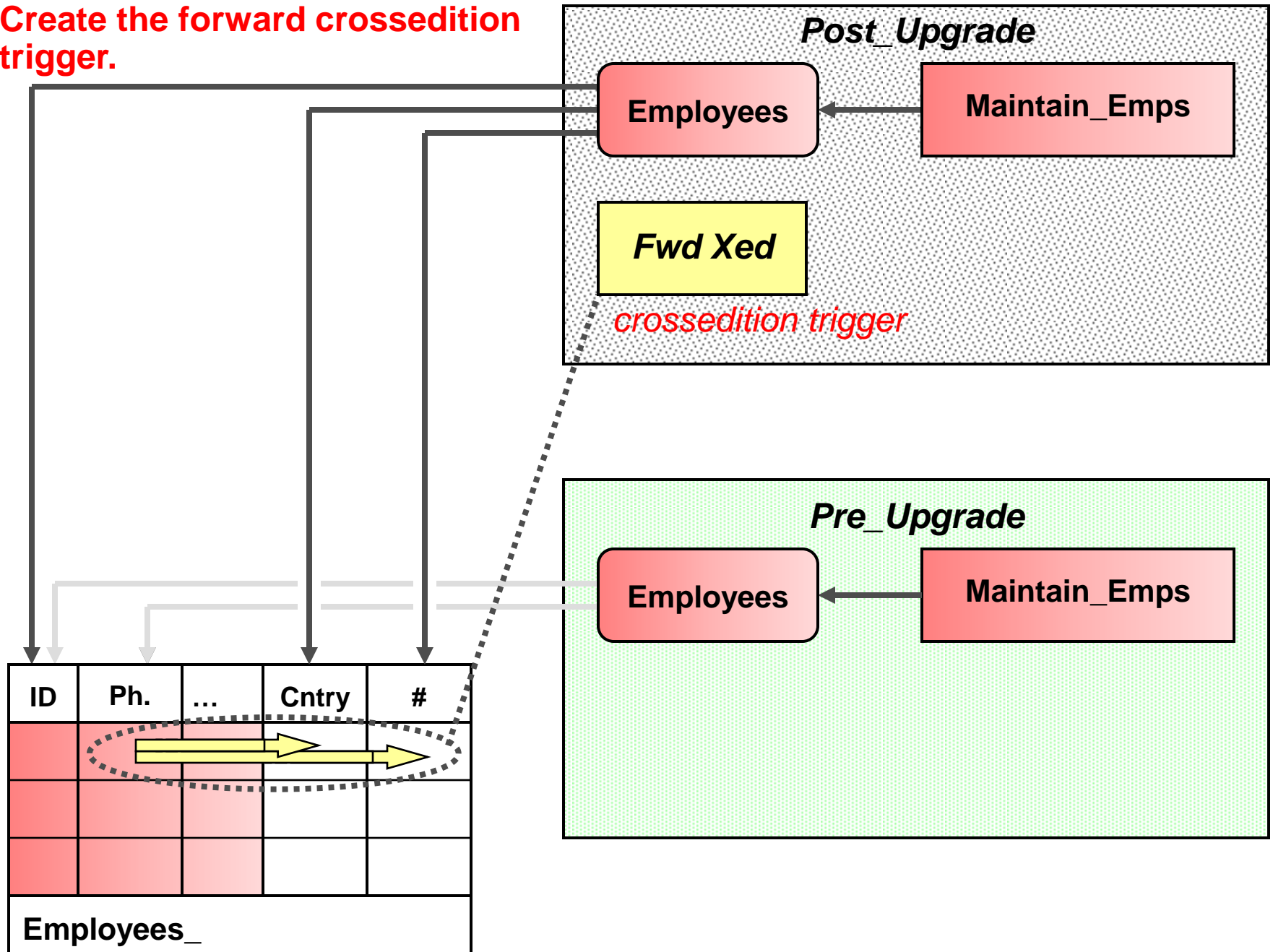
Change *Show_Employees* to implement the new behavior.



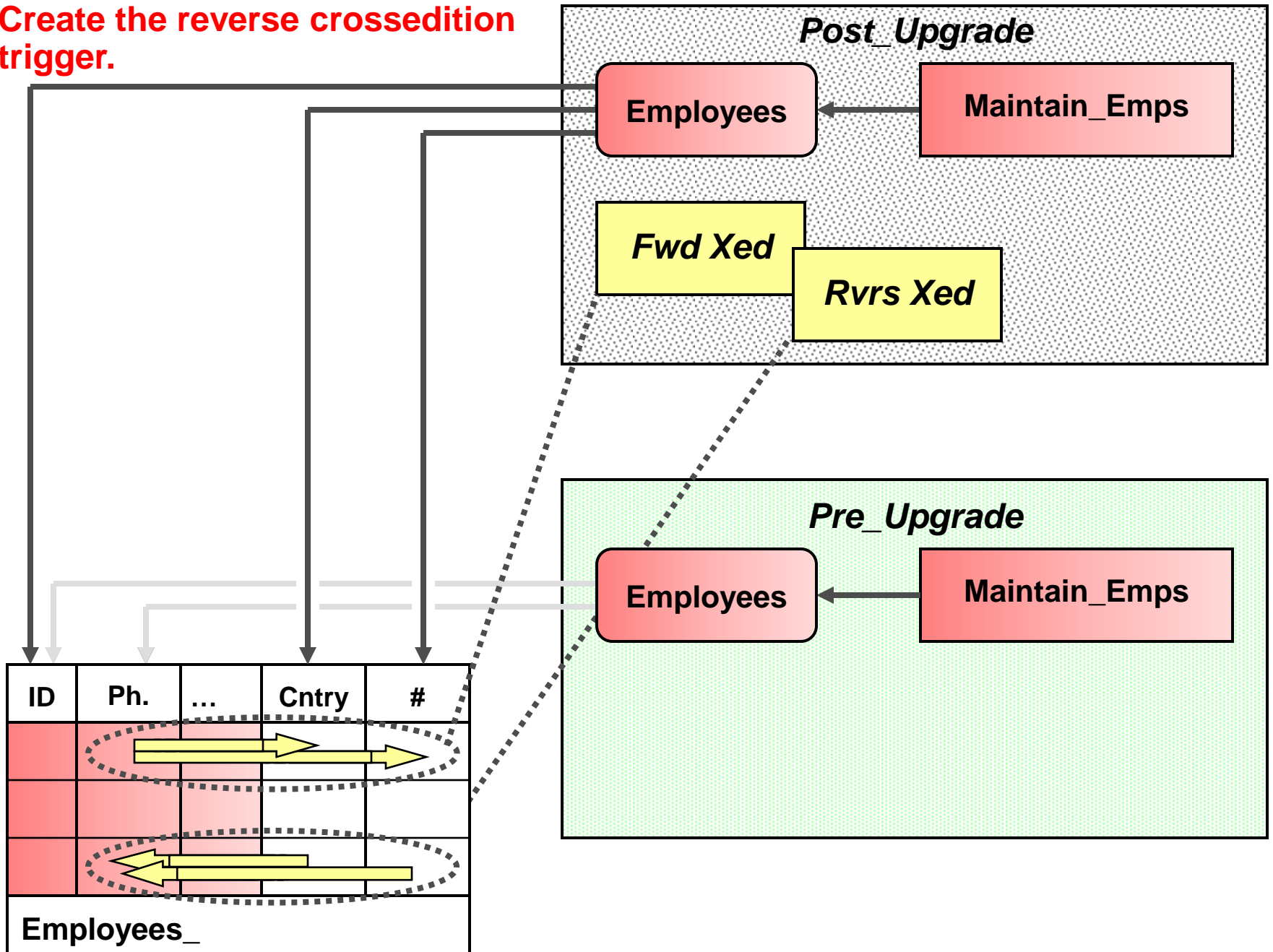
ID	Ph.	...	Cntry	#

Employees_

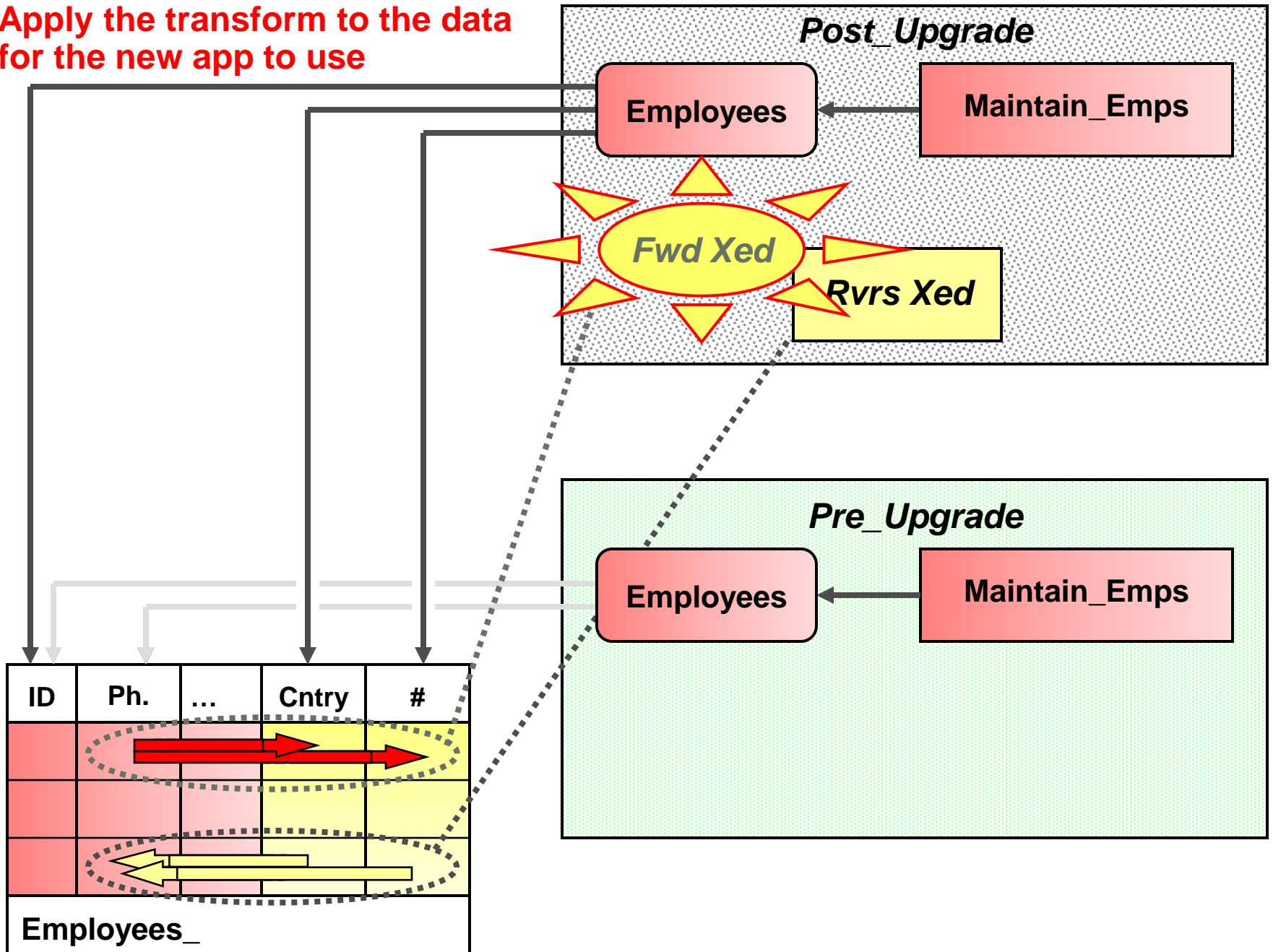
Create the forward crossedition trigger.



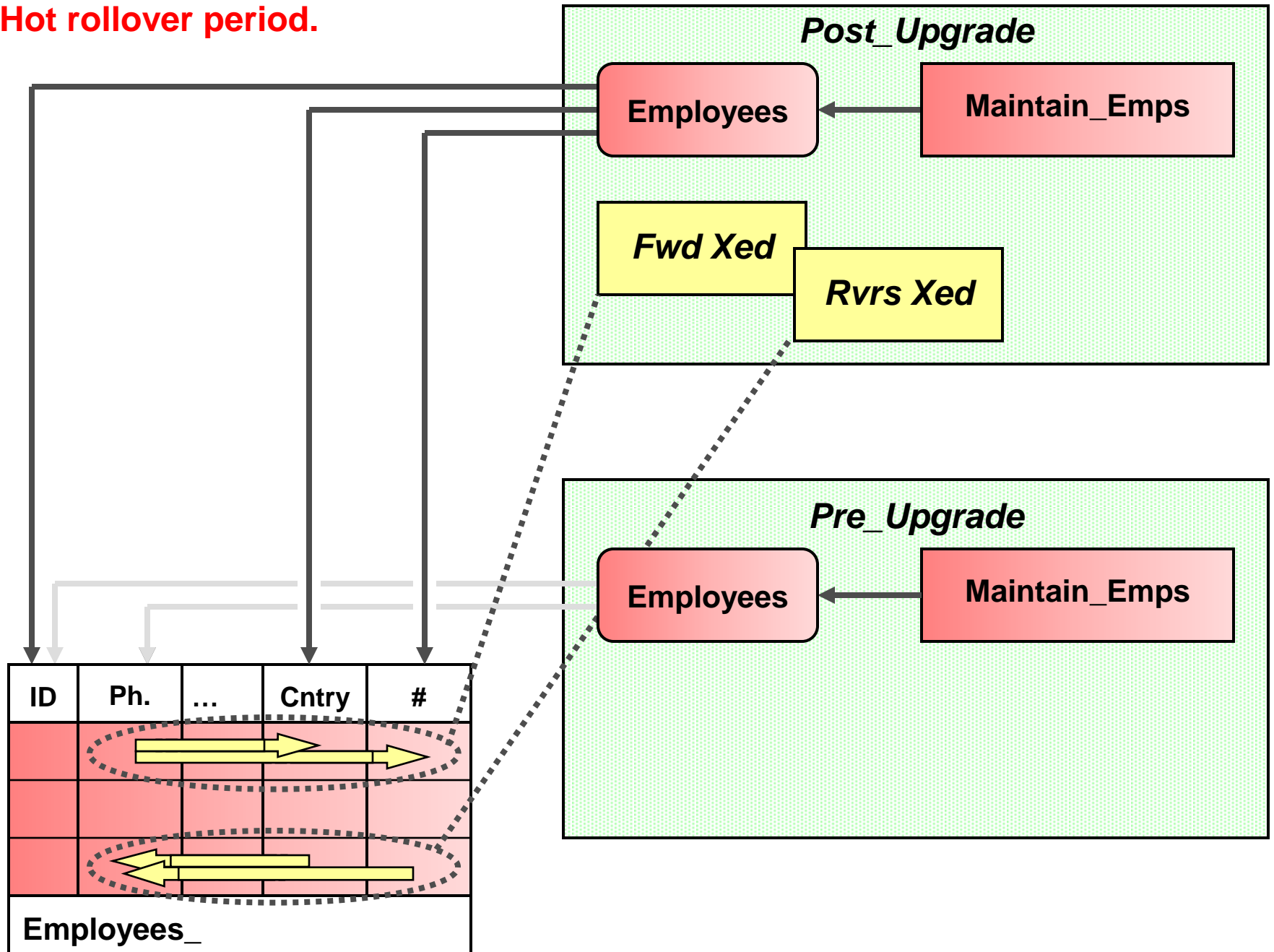
Create the reverse crossedition trigger.



Apply the transform to the data for the new app to use

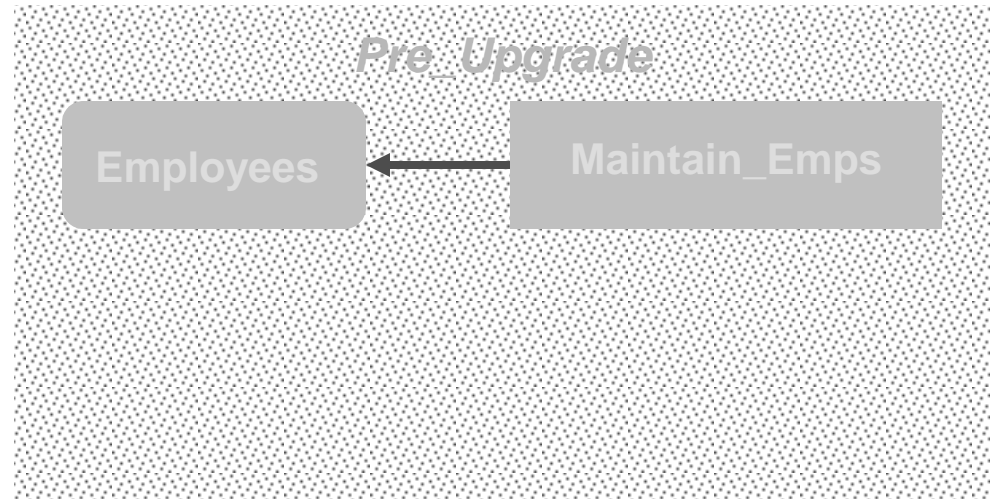
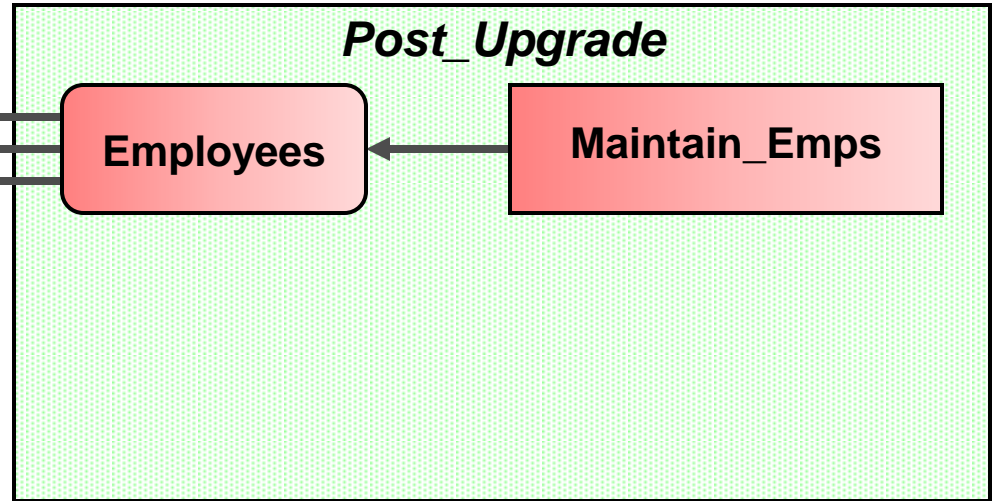


Hot rollover period.



The *Pre_Upgrade* edition is retired.

The edition-based redefinition exercise is complete.



ID	Ph.	...	Cntry	#

Employees_

Nota bene

- Online application upgrade is a high availability subgoal
- Traditionally, HA goals are met by features that the administrator can choose to use at the site of the deployed application
 - independently of the design of the application
 - without the knowledge of the application “vendor”
- The features for online application upgrade are used by the application “vendor”
 - when preparing the application for EBR
 - when implementing an EBR exercise
- Site administrators, of course, will need to understand the features

Program **Agenda**

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- **Bugdb application characteristics**
- Steps to EBR readiness
- EBR exercise
- Testing
- Summary
- References
- Questions and Answers



Oracle Bugdb Application Characteristics

- Defects and enhancement requests tracking system
- Mission critical
- 24/7 High availability requirement
 - 4 nodes RAC linux-X64 RDBMS 11g Release 2, ~ 500 gb , ias 10.1.2
 - Small custom mod/plsql application
 - Uses SES (context index), VPD, AQ
 - avg ~ 400 concurrent users in development and support LOBs
 - OLTP Web interface, many critical interfaces and hundreds of sqlnet clients (sqlplus and db links access).

EBR Project Goal

- Run online application upgrades
 - without performance impact
 - without wrong results
 - following EBR readiness best practices
- Proof of concept approach
 - Users selected based on high availability benefits
 - Edition notion introduction

Oracle IT Organization – Get All Teams Involved

- Architecture/security – process change, new technology, security review
- Operation – refresh test env, maintain mid tiers, EMGC monitoring, perform backups ...
- Database administrators – db upgrade, patches, tuning...
- Release mgt – collect change requirements, coordinate user acceptance testing w/ LOBs rep., approve application releases and patches
- App development – code, test, and automate patching
- Migration development – integrate acquisitions
- Application user base – Line of Business representatives

EBR Implementation Project Phases

- Readyng the application for EBR
 - Define the scope
 - Editions-enabling the users
 - EV layer Creation
- EBR exercise – change packages and triggers
- EBR exercise – include schema changes

Oracle Change Control Management Process

- Code in a dev db -> users test in a UAT db -> stage patch application -> production
- Data migration in a clone db -> test -> Prod

Program Agenda

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- Bugdb application characteristics
- **Steps to EBR readiness**
- EBR exercise
- Testing
- Summary
- References
- Questions and Answers



Readying for EBR

Database Administrators Tasks

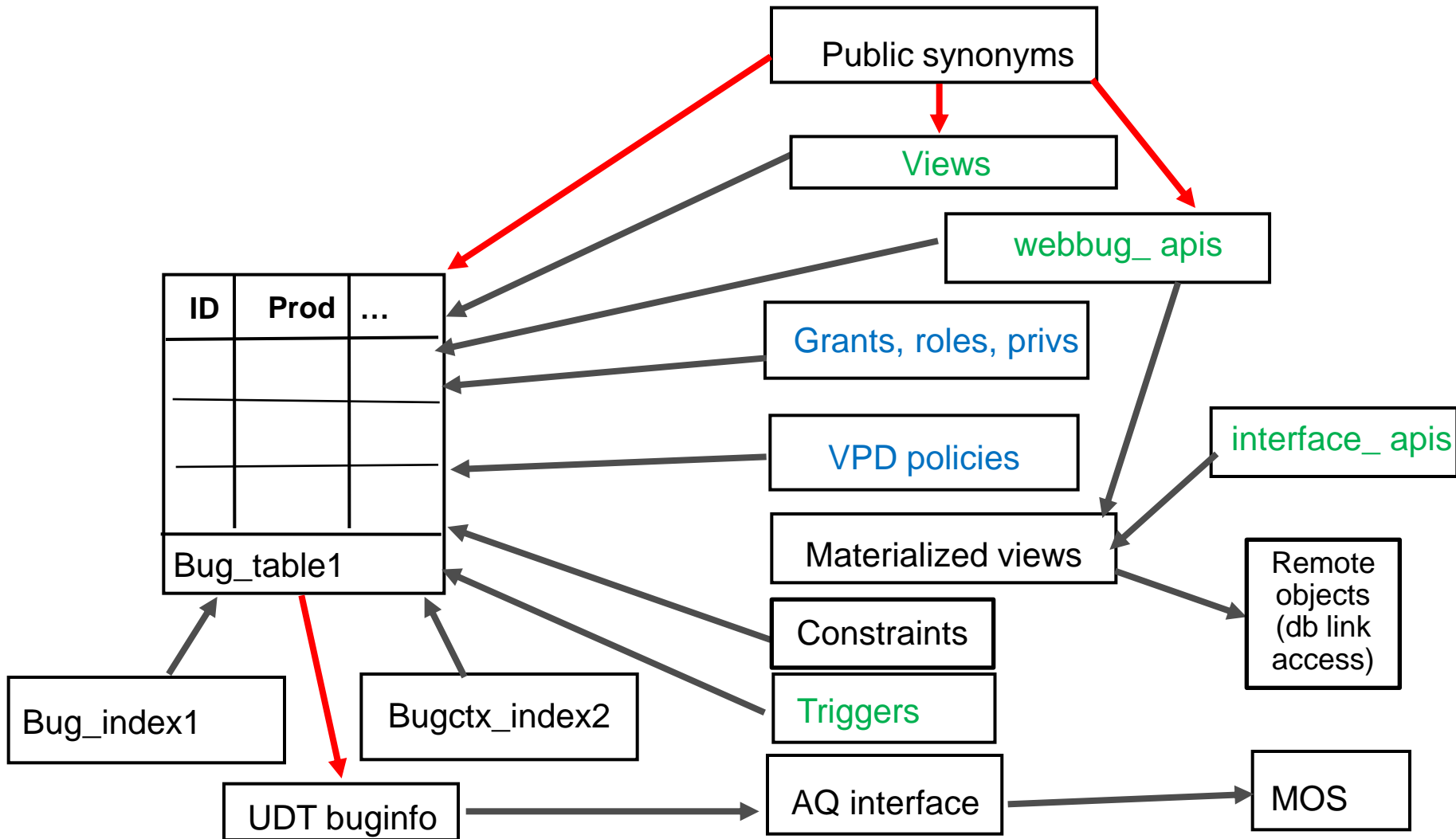
- Upgrade db to 11.2
- Set compatible => 11.2.0
- Increase SGA, tune memory parameters ahead
- Capture sql tuning set to run RAT (Real Application Testing) performance comparison before and after each change. Run AWR diff reports
- Grant developer role previously limited to dba role
 - Select_catalog_role, Select any dictionary
 - Create a ebr_admin package

Readying for EBR

Define the scope

- Categorize non system schemas with objects into 2 buckets based on their HA req. and functionality
 - HA req. : online transactions user interfaces and queries
 - No HA benefits: AQ interfaces and interfaces using MVs
- Check the conditions for a successful editions enabling
- Change the application to resolve the NE on E prohibition breakers
 - UDT (user defined type)
 - Public synonyms

Schema design before EBR readiness



Readying for EBR

Rules for Enable Editions to Succeed

There's no non-editioned object depending on editioned object type of this user, or if FORCE is specified

- To check for this condition :

Identify object pairs dependencies in target schemas

- Review pair of occurring objects discovered and move the rule breakers to a schema which will be not be edition enabled

NE on E prohibition breakers will fail w/ ora-38820, or become invalid, if –force is used when you enable editions

Readying for EBR

Rules for Enable Editions to Succeed

There's no evolved object type

- To check for this condition :

```
select type_name,version#  
from user_type_versions  
where version# > 1;
```

- You need to reset the object type with command :

```
alter type ... Reset;
```

Rule breakers will get ora-38820 error "user has evolved object type" or ORA-22374: cannot reset the version of a type with table dependents. In such case, you would have to drop the table, reset the type and reimport the table.

Readying for EBR

Development Team Tasks

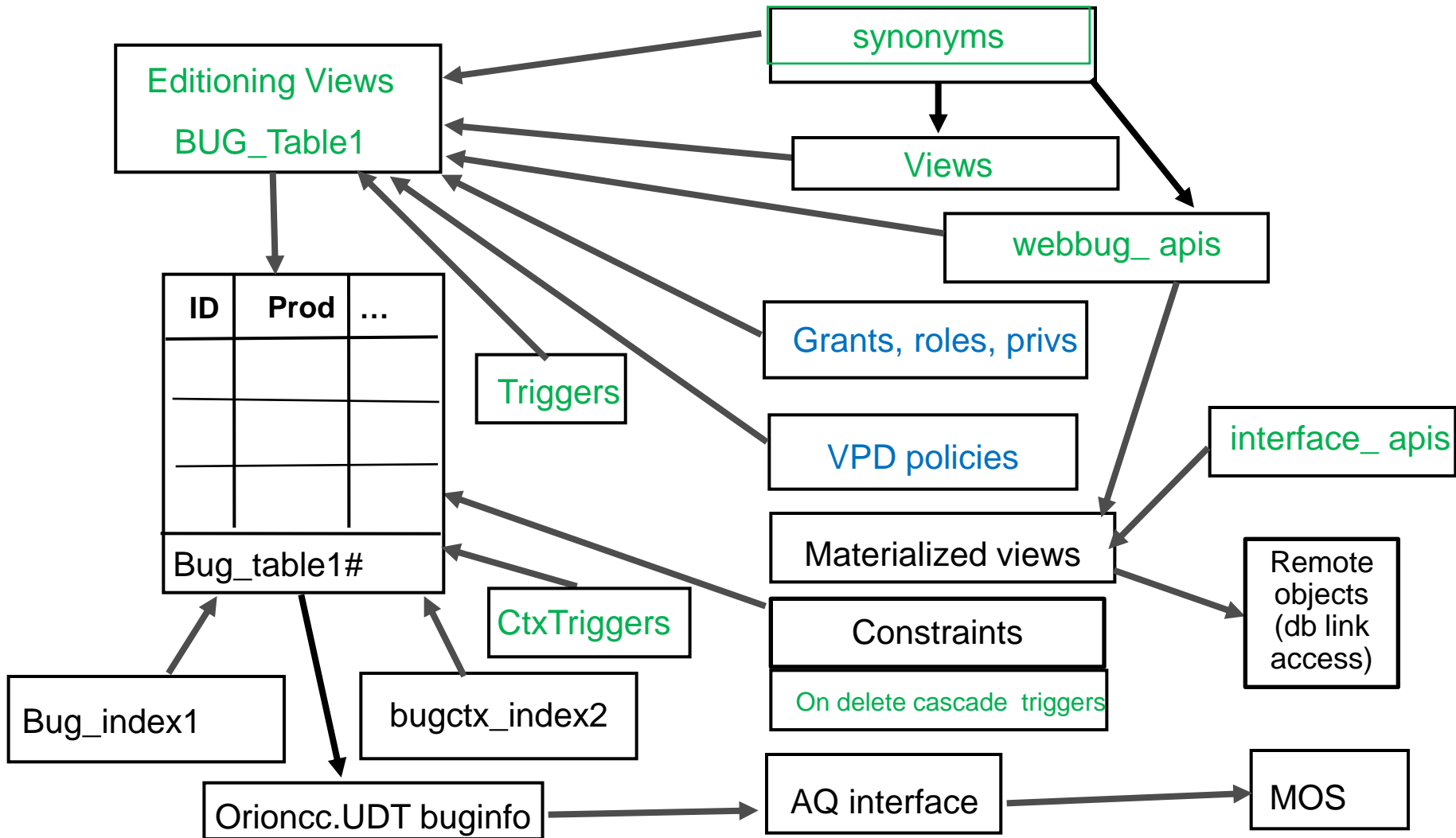
- Make application changes to remove the NE on E
 - For UDT, was used for AQ, move to another schema
 - Used `dbms_aq` apis to dequeue, stop the queue, and recreate in the non editioned schema
 - For public synonyms :
 - Drop the public synonyms for the 2 schemas
 - Create an on login trigger that set `current_schema` to main app schema and create a function to identify app users (excludes `sys`, `system`, `dbstmp` users)
 - Create private synonyms between the 2 targeted schemas so packages will compile

Readying For EBR

Editions-Enabling the Users

- Alter user BUG enable editions **force**;
- Alter user PRODCOMP enable editions **force**;
- Recompile
 - Procedure for developers (dbms_utility.validate)
 - Procedure for EBR admin
(sys.utl_recomp.recomp_parallel;)
- Check for any new invalids
 - `select * from dba_invalid_objects;`

Schema EBR ready



Readying for EBR

EV (Editioning Views) Layer Creation

- Rename tables to `substr(table_name,1,29) || '#'`
- Create EVs with same table names and same column names in same order (using `dbms_metadata` api)
- Drop grants on tables and recreate on the EVs
- Recreate VPD policies on EVs

- Generate new files for EVs definition for source control check-in
- Change all table definitions in source control to match the new table name # on the db

Program Agenda

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- Bugdb application characteristics
- Steps to EBR readiness
- **EBR exercise**
- Testing
- Summary
- References
- Questions and Answers



EBR Exercise

Application change

```
SYS > CREATE EDITION MR12 AS CHILD OF ORA$BASE;
```

```
SYS > GRANT USE ON EDITION MR12 TO BUG;
```

```
BUG > EXEC DBMS_SESSION.SET_EDITION_DEFERRED('MR12');
```

```
BUG> @ InstallMR12code.sql
```

```
SYS > exec sys.utl_recomp.recomp_parallel;
```

```
SYS> select * from dba_invalid_objects;
```

```
SYS> ALTER DATABASE DEFAULT EDITION = MR12;
```

EBR Exercise

Monitoring the hot rollover

- Monitor session_edition_id on v\$session

```
SELECT SYS_CONTEXT('Userenv', 'Current_Edition_Name') FROM DUAL;
```

```
select object_name, session_edition_id, type, inst_id, v.status, count(*)  
from gv$session v, dba_objects  
where session_edition_id=object_id  
group by object_name, session_edition_id,type, inst_id, v.status  
order by object_name, session_edition_id,type, inst_id, v.status
```

EBR Exercise

Monitoring the hot rollover – cont.

- Observe background (MMON) and users
 1. Web interface users
 2. Sqlplus, db link access
 3. Critical interfaces (connection pooling, AQ interface)
 4. Other agents
(ex. Emgc dbsnmp, CRS oraagent.bin)
- Terminate remaining sessions using previous edition prior to retiring the edition

Program Agenda

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- Bugdb application characteristics
- Steps to EBR readiness
- EBR exercise
- **Testing**
- Summary
- References
- Questions and Answers



Testing

- Functional testing
 - Special attention to triggers firing and VPD policies
- User Interface testing – use 11.2.0.2 service

```
SET ORA_EDITION=e1  
sqlplus usr/p@11202  
sql> SHOW EDITION
```

- Concurrent edition usage test (during OAU)
- Scheduler jobs
- Sql loader – control file “into table”=<evname>
- Cursors stats

Performance Testing

- Use Real Application Testing
- Mass update the parsing_schema_name to match the schema defined in the on login trigger (BUG)
 - select * from dba_sqlset where name like 'EBR%';
 - Get the ID ex. 25
 - update wri\$_sqlset_statements set parsing_schema_name='BUG' where sqlset_id=25;
 - update wri\$_sqlset_plans p set parsing_schema_name='BUG' where stmt_id in (select id from wri\$_sqlset_statements where sqlset_id=25);

Program **Agenda**

- Introduction
- Challenge of online application upgrade
- Brief EBR overview
- Bugdb application characteristics
- Steps to EBR readiness
- EBR exercise
- Testing
- **Summary**
- References
- Questions and Answers



Summary

- Oracle Database 11g Release 2 brings a revolutionary new capability that supports **online application upgrade**
 - This means preserving most of the data, changing some of it, and changing code – all while it's in continuous use
 - It's called **edition-based redefinition**
 - It works *in place* (no need for a second db).
Old and new representations of the same data are synchronized transactionally.
No need for human conflict resolution.

Summary

- A mission critical Oracle Corp internal application has gone live with EBR
 - the application has been readied (11.2 platform; users are editions-enabled)
 - an EBR exercise has been conducted with zero downtime
 - users didn't even notice the hot rollover

Summary

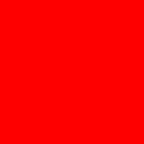
- The mission critical internal application is BugDB
 - Half of all Oracle employees, world wide, are authorized to use it and might need to at a moment's notice
 - There are typically hundreds of concurrent end-user sessions
 - There are also very many mechanical clients

References

- Read the edition-based redefinition chapter in the Oracle Database Advanced Application Developers' Guide, 11.2
- Download Bryn's OpenWorld 2010 session S318090
- Read Bryn's whitepaper: published on the High Availability subpage under the Database page on OTN
- Listen to Bryn's 30-minute recorded presentation
download.oracle.com/technology/products/database/files/Edition_Based_Redefinition_Overview_ReadMe.pdf
download.oracle.com/technology/products/database/files/Edition_Based_Redefinition_Overview.zip
- Internet search for *edition-based redefinition*

Q&A





The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

ORACLE®

SOFTWARE. HARDWARE. COMPLETE.