[Simple way to create truly Dynamic Breadcrumbs in Oracle Apex](http://thedaya.blogspot.com/2015/08/simple-way-to-create-truly-dynamic.html)

So, when wanting to create a Dynamic Breadcrumb for an application I was working on, Google didn't really give me an adequate solution - Their solutions work, but were not really effective for pages that can be called from many different "Parents".

I spent a little time and came up with a simple solution that once implemented, should remove the need to ever worry about parenting breadcrumb regions, or even adding breadcrumbs to newly created pages.

Method behind this is simply:

\* As opposed to using the standard "static" breadcrumb, use an apex collection to manage the tree of pages visited.

\* Write robust procedures to manage the inserting of data into this collection (which includes figuring out when to clear the collection, for instance, when the Home page is visited), and functions to retrieve a html marked string that translates into hyperlinks, which can be displayed in the page title - in effect, becoming your breadcrumb :)

Lets start with the database procedures needed.

one package, 2 public procedures.

**Spec**

create or replace PACKAGE "BREADCRUMB\_PKG" IS

function get\_breadcrumb(p\_collection\_name varchar2, p\_app\_id number, p\_session\_id number) return varchar2;

procedure add\_entry(p\_collection\_name varchar2,p\_app\_id number, p\_page\_id number, p\_reset\_breadcrumb varchar2 default 'N');

END breadcrumb\_pkg;

/

**Body**

create or replace PACKAGE BODY "BREADCRUMB\_PKG"

AS

FUNCTION get\_breadcrumb(p\_collection\_name VARCHAR2,p\_app\_id number, p\_session\_id number ) RETURN VARCHAR2 IS

--return html breadcrumb

w\_link varchar2(2000);

w\_link2 varchar2(2000);

CURSOR c1 IS

select seq\_id,c001, c002

 from apex\_collections

 WHERE collection\_name = p\_collection\_name

 order by seq\_id asc;

begin

FOR r1 IN c1 loop --loop through entries in collection, produce html

w\_link2:=w\_link ||r1.c002||'';

w\_link := w\_link || '<a href="f?p='||p\_app\_id||':'||r1.c001||':'|| p\_session\_id||'::NO:::" class="t-Breadcrumb-label">'||r1.c002||' > </a>';

end loop;

return w\_link2;

END get\_breadcrumb;

 procedure add\_member (p\_collection\_name varchar2, p\_app\_id number, p\_page\_id number) is

 --get page title and add to collection

 w\_page\_name varchar2(40);

 begin

 select page\_title

   into w\_page\_name

   FROM APEX\_APPLICATION\_PAGES

  where page\_id = p\_page\_id and APPLICATION\_ID = p\_app\_id and page\_mode = 'Normal';   
--note, coded to specifically exclude modals. you may remove if necessary

APEX\_COLLECTION.ADD\_MEMBER(

        p\_collection\_name => p\_collection\_name,

        p\_c001            => p\_page\_id,

        p\_c002            => w\_page\_name);

  exception

    WHEN no\_data\_found THEN

            NULL;

end add\_member;

procedure add\_entry (p\_collection\_name varchar2,p\_app\_id number, p\_page\_id number,  p\_reset\_breadcrumb varchar2 default 'N') is

-- add page entry to breadcrumb

cursor c1 (p\_curr\_seq\_id number)is

select seq\_id

 from apex\_collections

 where collection\_name = p\_collection\_name

 and seq\_id > p\_curr\_seq\_id;

begin

IF( (NOT APEX\_COLLECTION.COLLECTION\_EXISTS (p\_collection\_name))OR p\_reset\_breadcrumb = 'Y') THEN

if (apex\_collection.collection\_exists(p\_collection\_name)) then

     APEX\_COLLECTION.TRUNCATE\_COLLECTION (p\_collection\_name);

else

    APEX\_COLLECTION.CREATE\_COLLECTION(p\_collection\_name);

end if;

add\_member(p\_collection\_name, p\_app\_id , p\_page\_id );

else

declare

  w\_curr\_seq\_id number;

    begin

--first check if page exists in collection

  select seq\_id

    into w\_curr\_seq\_id

    from apex\_collections

    where collection\_name = p\_collection\_name

     and c001 = p\_page\_id;

for r1 in c1 (w\_curr\_seq\_id) loop --if it does , delete entries that follow (caters for navigating back to a page)

    APEX\_COLLECTION.DELETE\_MEMBER(

        p\_collection\_name => p\_collection\_name,

        p\_seq => r1.seq\_id);

end loop;

exception

WHEN no\_data\_found THEN --if entry does not exist, insert

 add\_member(p\_collection\_name,p\_app\_id,p\_page\_id );

end;

END IF;

end add\_entry;

END breadcrumb\_pkg;

/

Next, we need to utilize this from the apex front end.

First, create a hidden item on page 0 called P0\_BREADCRUMB.

         \* Set source to PL/SQL expression, and put this as the source.

breadcrumb\_pkg.get\_breadcrumb('BREADCRUMB',:APP\_ID,:A\_SESSION\_ID);

We also need an APPLICATION ITEM  - A\_RESET\_BREADCRUMB. This can help us control when to completely reset the breadcrumb, for instance, if an item from the navigation menu in clicked, it makes sense to clear our “tree”.

Now lets create a region to show the breadcrumb.

      1)    On page 0, create  a region of type PL/SQL Dynamic contact, and set region position to Breadcrumb Bar.

      2)    Now, add the code below to the PL/SQL block area .

begin

htp.print('<BR> &nbsp;&nbsp;  '||:P0\_BREADCRUMB||'<BR>');

end;

We now need a way to populate the collection, so that when breadcrumb\_pkg.get\_breadcrumb is called, it can loop through it and return an html breadcrumb.

We do this by creating an APPLICATION PROCESS,on load (before header).

Add the following code to that.

declare

begin

breadcrumb\_pkg.add\_entry('BREADCRUMB',:APP\_ID,:APP\_PAGE\_ID,:A\_RESET\_BREADCRUMB);

if(:A\_RESET\_BREADCRUMB = 'Y') then

:A\_RESET\_BREADCRUMB:='N';

end if;

end;

And……that’s it, you now have a completely dynamic breadcrumb 

When wanting to reset the breadcrumb (for items on the navigation menu for instance), simply set the application item A\_RESET\_BREADCRUMB to ‘Y’ as part of the parameters.

Pop a comment if you have any difficulties using this. Cheers

Posted 14th August 2015 by Unknown

# [Barry's Oracle Development Blog](http://barrybrierley.blogspot.com/)

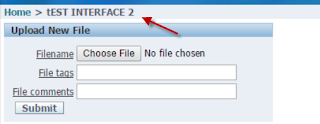
I hate reading manuals. I find most of my answers on forum posts and blogs. But there's a lot of incorrect and outdated information out there. This is what worked for me.

## Wednesday, 14 October 2015

### Dynamically changing a breadcrumb name

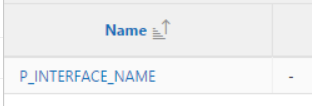
#### Problem

You want your breadcrumb to have a dynamic name.

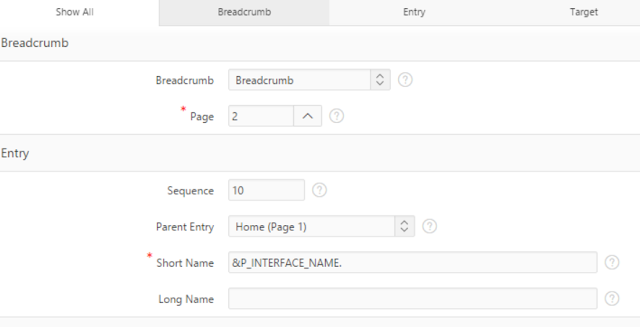
[](http://4.bp.blogspot.com/-mLhHl3zejcY/Vh4Tp6NusqI/AAAAAAAAB-s/t32wngoN12Q/s1600/14-10-2015%2B09-34-09.png)

#### Solution

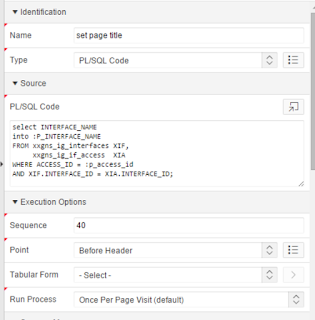
Create a Shared Component > Application Item .

[](http://4.bp.blogspot.com/-BRyTDdOBtkM/Vh4USEC9unI/AAAAAAAAB-0/kwStbjxelCk/s1600/14-10-2015%2B09-36-53.png)

 Edit your Breadcrumb entry for the Page that is Create / Edit.  
 Change the Short Name to &(Application\_item).  
NOTE: The fullstop at the end of it.

[](http://4.bp.blogspot.com/-4Q4vljCla1g/Vh4Uw4DzJmI/AAAAAAAAB-8/UL8L8Jx-UNs/s1600/14-10-2015%2B09-38-17.png)

 Create a process to populate the variable. In my it was a before header process on the page.

[](http://2.bp.blogspot.com/-RacGqiLkRhg/Vh4Vpb42FqI/AAAAAAAAB_E/29YEU7wi1UM/s1600/Capture.PNG)

