

ORACLE®

# Transforming one table to another: SQL or PL/SQL?

Bryn Llewellyn  
Distinguished Product Manager  
Database Division  
Oracle HQ  
*twitter: @BrynLite*

November 2015

# Safe Harbor Statement

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.

# DOAG 2015

You've heard the mantra, "If you can do it in SQL, do so; only if you can't, do it in PL/SQL." Really? What if you want to compute nonoverlapping RowID ranges that completely span a table? An exact solution is trivial, but it's too slow for a table of typical size. Approximate solutions, using only SQL, are posted on the internet. But they impose limitations, and leave generalization to the reader. This session examines alternative general solutions—some that use only SQL and others that combine SQL and PL/SQL. All pass the same correctness tests and all have comparable performance. Choosing one, therefore, requires some rather subtle and subjective considerations. Your choice might be different, and the speaker will welcome the debate.

# Whitepaper and code kit download

- My presentation slides are no use without me presenting them
- I have written a whitepaper that gives a careful and detailed account of what my presentation addresses
- I have also provided a self-contained code kit
  - go to the PL/SQL page on OTN: **`oracle.com/plsql`**
  - notice the link to the Oracle Database PL/SQL and EBR blog
  - notice the blog that announces this whitepaper and code kit
  - notice my Twitter handle *@BrynLite*.  
I'll tweet news on updates to the paper or the code kit

# **Hardware and Software Engineered to Work Together**