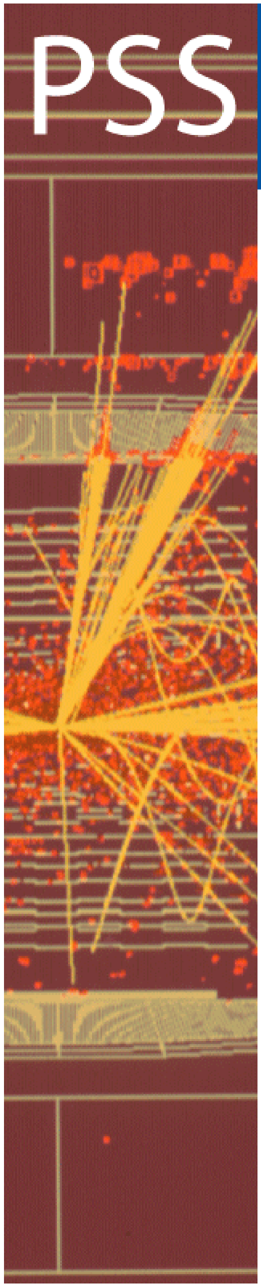
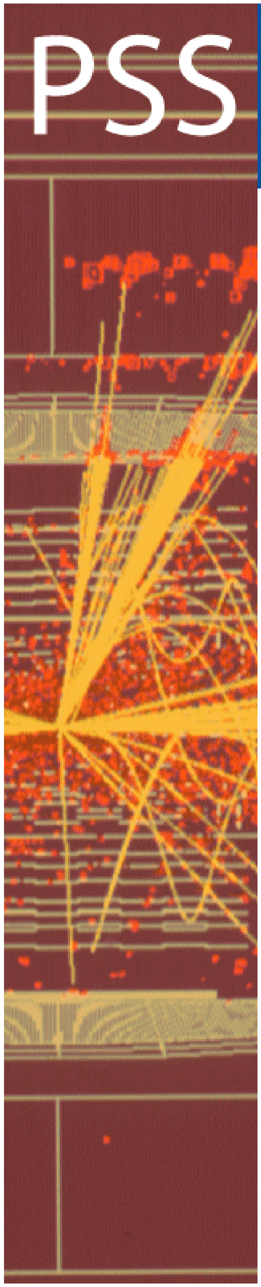
PSS Physics Services Support

CERN IT Department



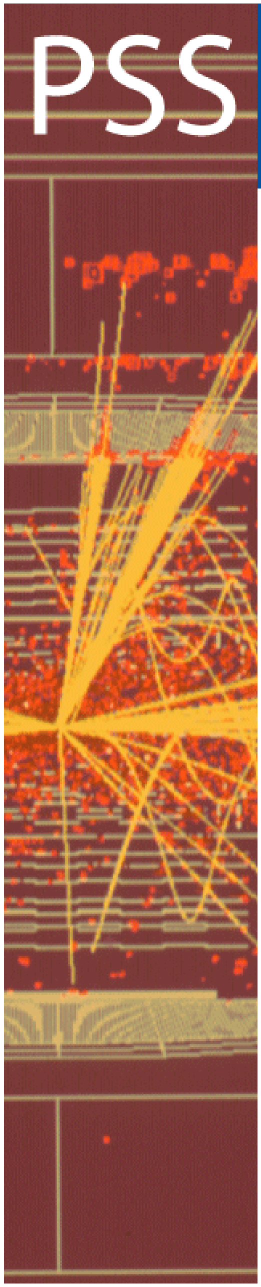
Transportable Tablespaces for Scalable Re-Instantiation

Eva Dafonte Pérez



Outline

- Overview
- Transportable Tablespaces
- Pulling Tablespaces
- Pros and Cons
- Streams Re-synchronization procedure using Transportable Tablespaces
  - Idea
  - Example and Steps

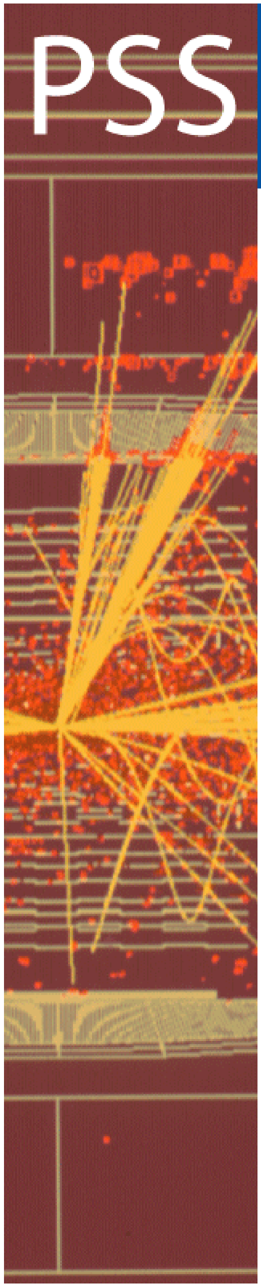


Overview

- Target site out of the Streams recovery window
- Complete transfer of data (schemas and tables) using Oracle Data Pump utility to destination database may take days
  - Example ATLAS Conditions data

➔Find **fastest** way to move data

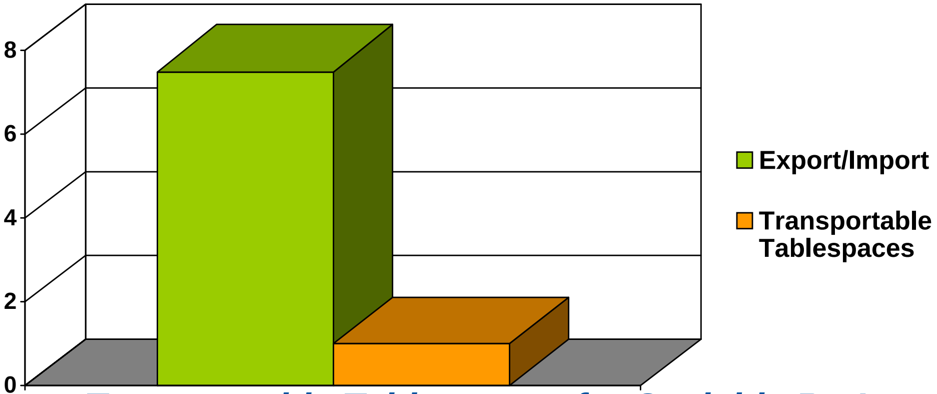
- Options:
  - Transportable Tablespaces
  - Pulling Tablespaces (new in Oracle 10g)

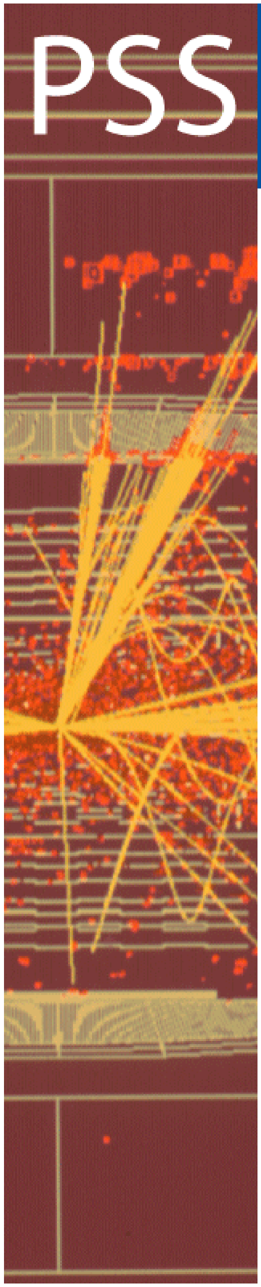


Transportable Tablespaces

- Move a set of tablespaces from one Oracle database to another
- Export metadata of tablespace instead of data in tablespace
  - Datafiles containing all data copied to destination
  - Plug the tablespace by importing metadata
- Moving data using transportable tablespaces is much faster than export/import

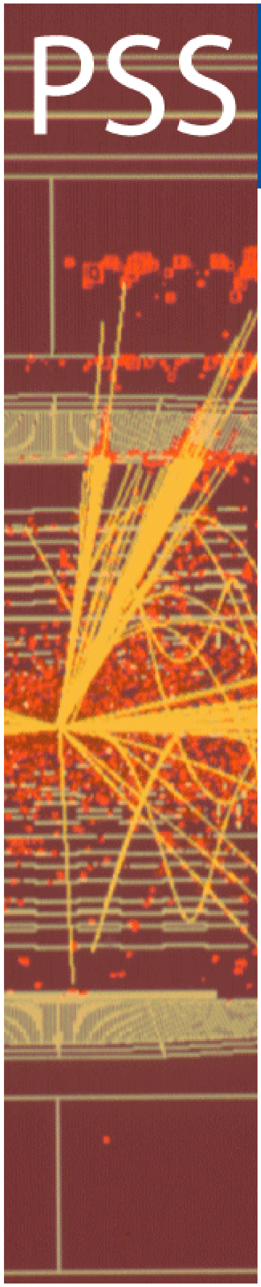
Example: ATLAS COOL test data - 67 G





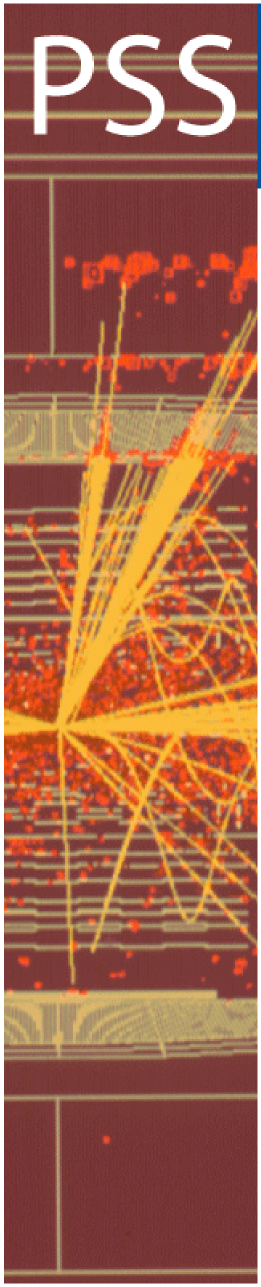
Transportable Tablespaces

- Restrictions:
  - Database block size and character set must be the same at source and target
  - The Oracle release and the compatibility must be the same or higher at target
  - Tablespace must be self contained
  - User/s must exist at target database
- Cross-Platform since Oracle 10g
  - Oracle provides a byte-order-conversion solution that uses Oracle Recovery Manager (RMAN)



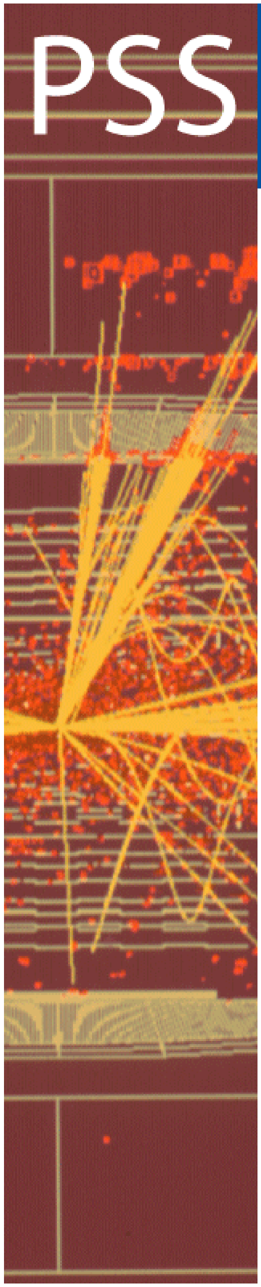
Pulling Tablespaces

- Single packaged procedure
  - DBMS_STREAMS_TABLESPACE_ADM
    - PULL_SIMPLE_TABLESPACE
    - PULL_TABLESPACES
  - Uses Data Pump to transport tablespaces
  - Transfers the data files to the target system's format
    - Using DBMS_FILE_TRANSFER
  - Performs any required endian conversion automatically
  - Creates a log file
- All tasks are encapsulated



Pros and Cons

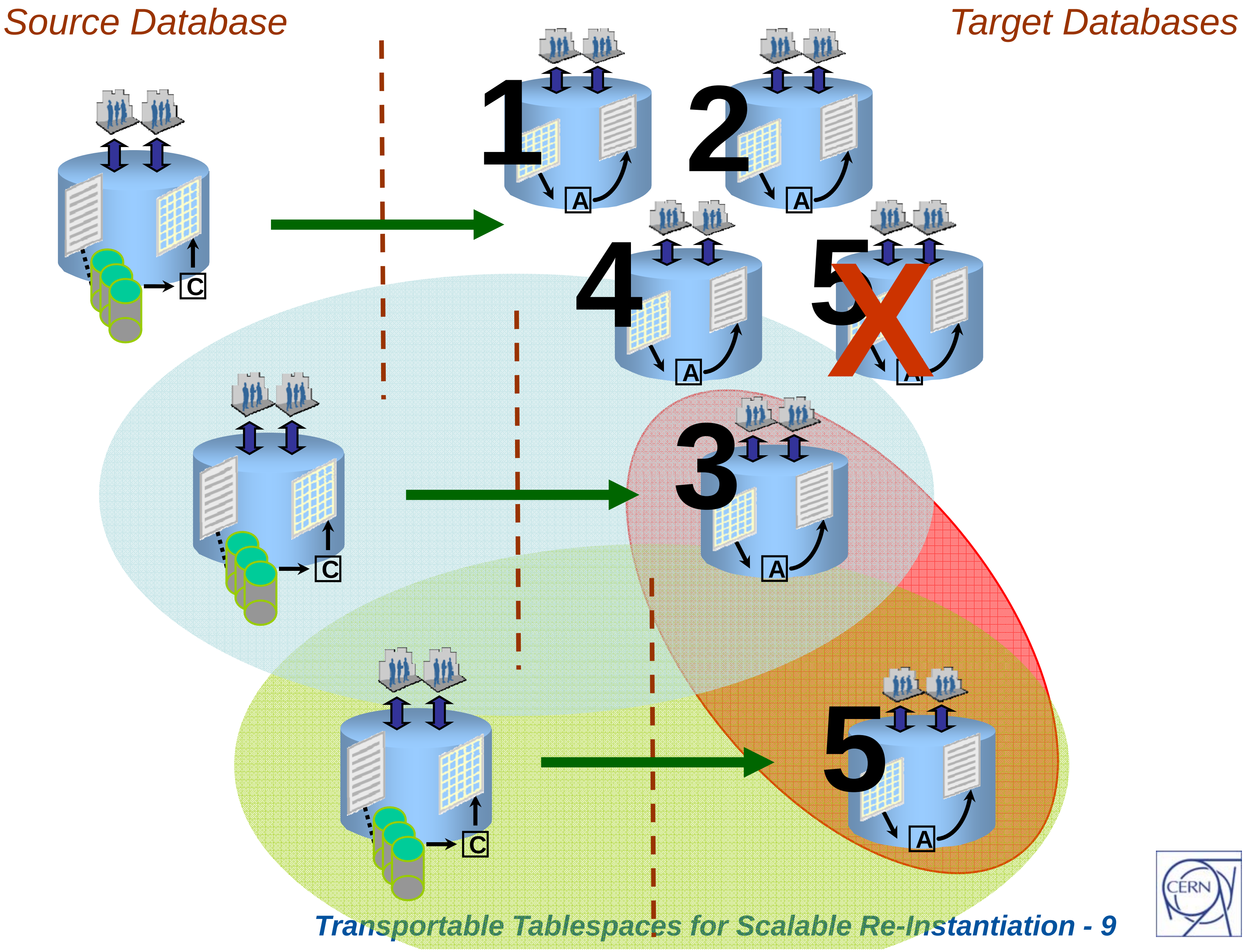
- Using Transportable Tablespaces:
  - Enables a full set of tablespaces to be moved
  - Fastest approach
  - Tablespaces need to be set to read-only while the files are copied
- Using Pulling Tablespaces:
  - Combines all steps in the transportable tablespaces approach into a single step
  - Less flexibility - multiple files transferred sequentially
  - Tablespaces need to be set to read-only

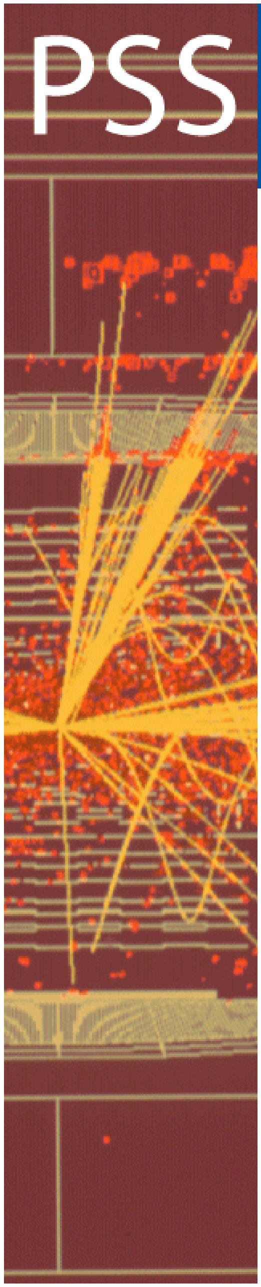


Streams Re-Synchronization

- Idea:
  - Use transportable tablespaces to move data faster
  - Complete re-instantiation using Streams
- Transport tablespaces from Tier0 is **not** possible
  - Cannot be read-only
  - Contains more data than Tier1's tablespaces
- BUT, we can transport tablespaces from Tier1
  - Can be read-only
  - Contains exactly the data that must be transferred

Streams Re-Synchronization
Example and Steps





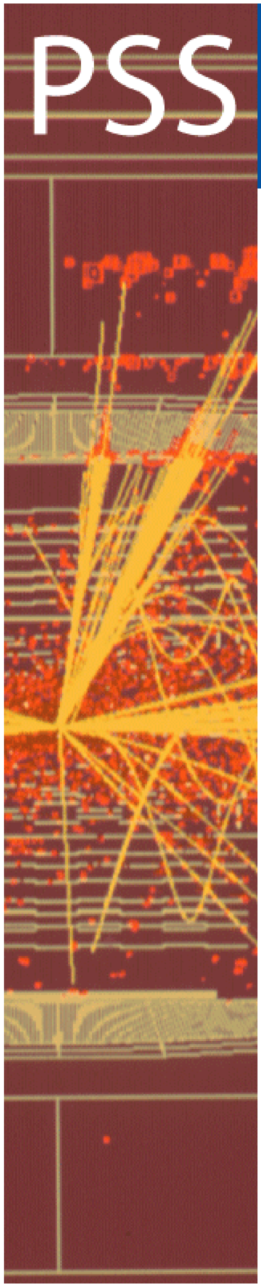
Streams Re-Synchronization
Example and Steps

- Example:
  - Original streaming to sites 1, 2 , 4 and 5
  - Separate streaming for site 3
  - Site 3 is out of sync
  - Would like to use site 5 to synchronize site 3
  - Have to move tablespaces “TS1” and “TS2”

- Steps:
  - Check tablespaces set is self-contained

```
EXEC SYS.DBMS_TTS.TRANSPORT_SET_CHECK ('TS1','TS2');
SELECT * FROM TRANSPORT_SET_VIOLATIONS; ✓
```

  - Split streaming for site 5



Streams Re-Synchronization
Example and Steps

- Steps:
  - Check sites 3 and 5 are able to connect to each other
  - Create database links between databases
  - Create directories pointing to datafiles and grant access to streams administrator on both sites
  - Stop replication to site 5
  - Ensure tablespaces are read-only

```
SELECT SELECT STATUS FROM DBA_TABLESPACES
WHERE TABLESPACE_NAME IN ('TS1','TS2');

ALTER TABLESPACE TS1 READ ONLY;

ALTER TABLESPACE TS2 READ ONLY;
```

Streams Re-Synchronization Example and Steps

CERN IT Department

- Steps:
  - Transfer the data files of each tablespace to the remote system
  - Import tablespaces metadata in the target

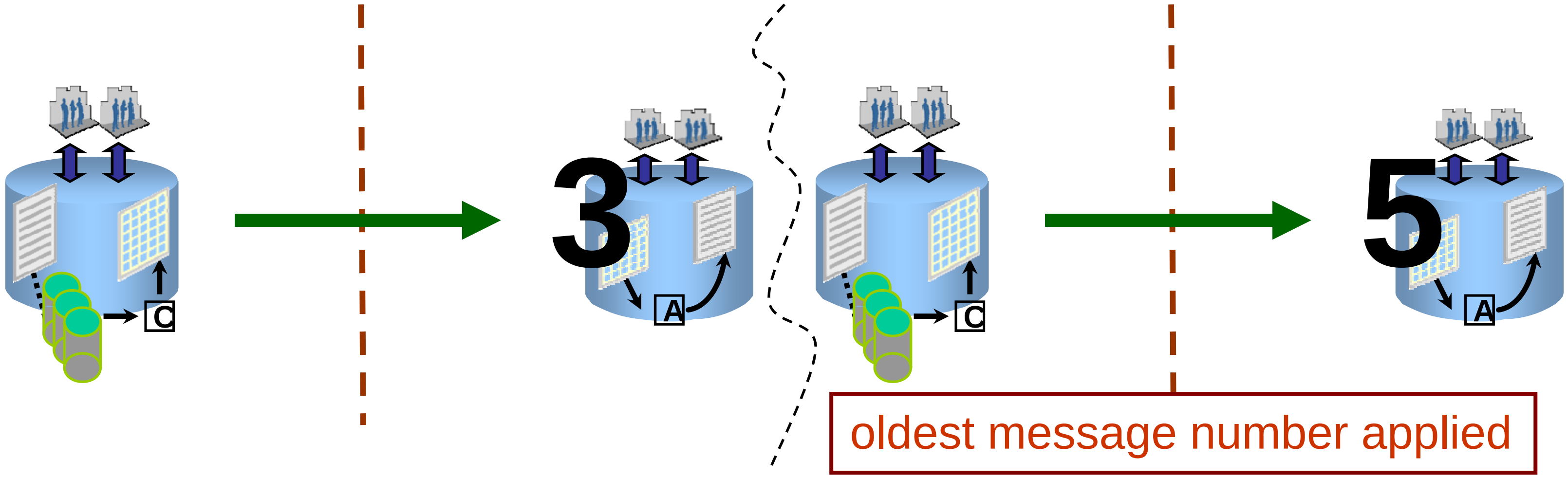
```
impdp user/pwd
TRANSPORT_DATAFILES="'/oradata/ts1_1.dbf','/oradata/ts2_1.dbf'"
NETWORK_LINK='srcdb'
TRANSPORT_TABLESPACES=(TS1,TS2)
NOLOGFILE=Y
```

  - Make tablespaces read-write

```
ALTER TABLESPACE TS1 READ WRITE;
ALTER TABLESPACE TS2 READ WRITE;
```

Streams Re-Synchronization
Example and Steps

- We have a consistent copy of the data between sites 3 and 5
- And we need to re-configure replication to site 3

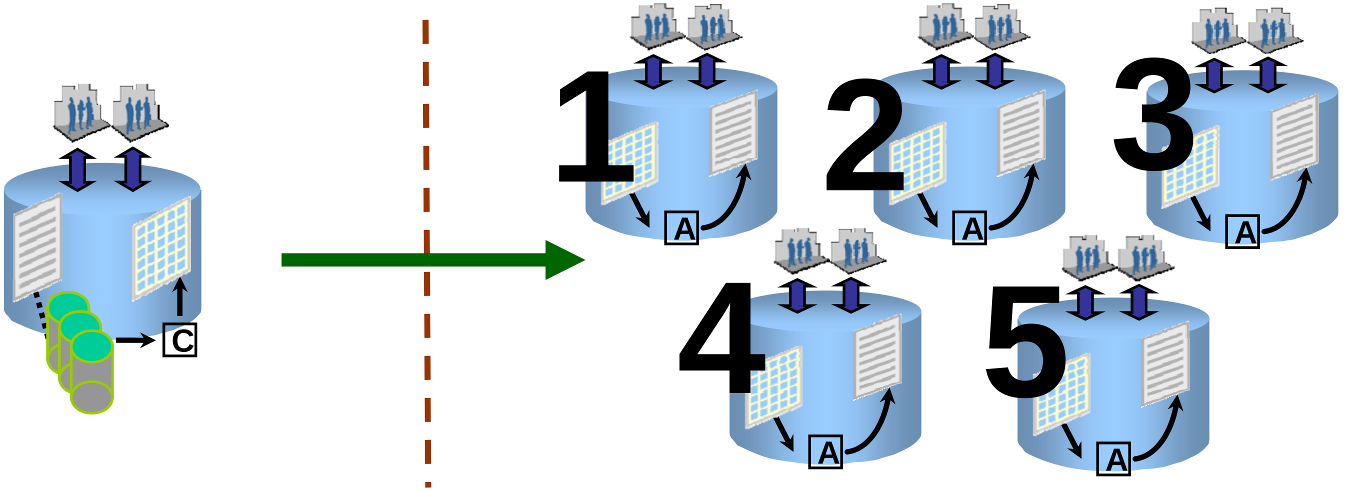


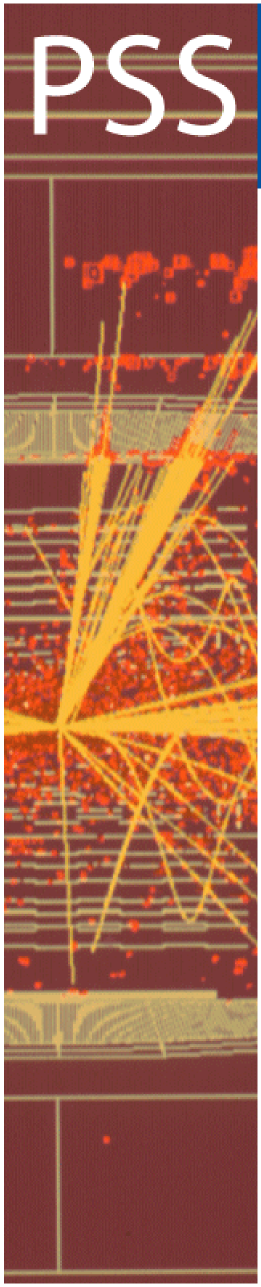
oldest message number applied

➔ first_scn = previous dictionary build
➔ start_scn =

Streams Re-Synchronization
Example and Steps

- Enable replication to site 5
- Enable replication to site 3
- Last transaction might be reapplied!!!!
  - Just ignore the error
- Wait
- Merge all the streaming





Summary

- Transportable tablespaces is a fastest way to move data between databases
- Can be used between Tier1 sites when complete synchronization is needed
- Successfully tested last week during CNAF re-synchronization for ATLAS conditions data