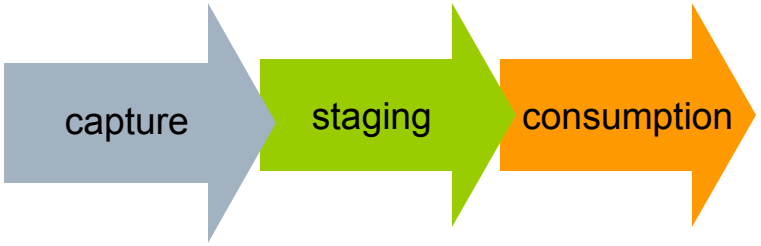
Replication using Streams at RLS

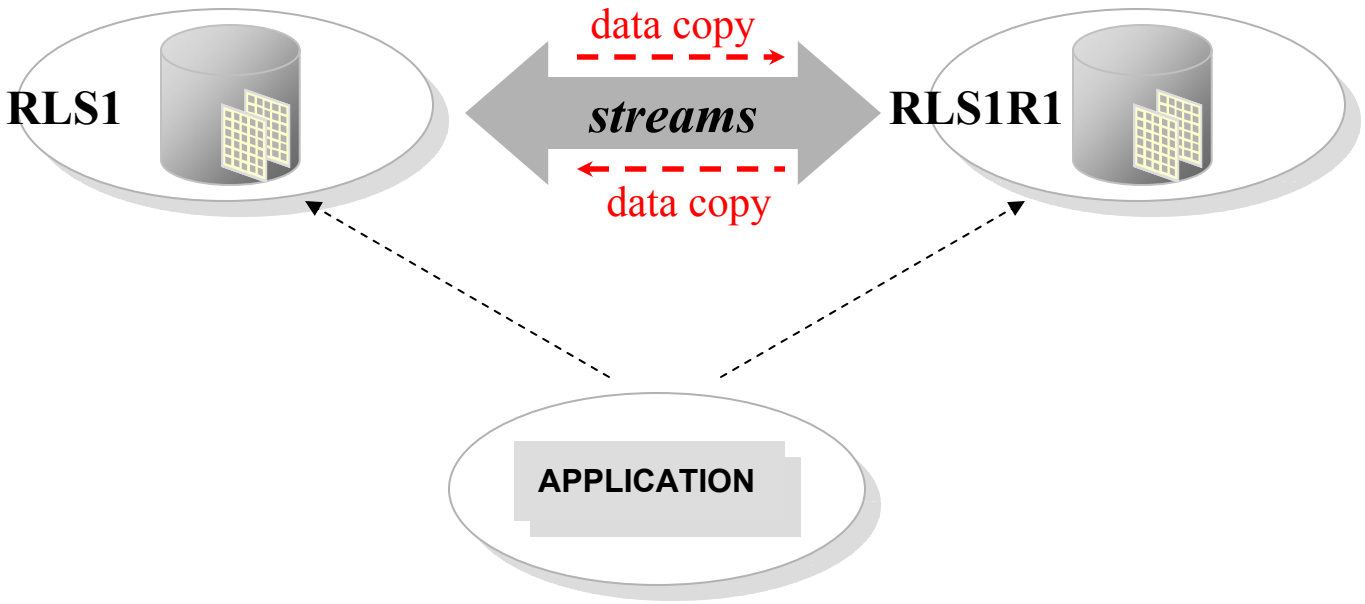
- Oracle Streams
  - new way to share information between databases
  - Data replication:
    - captures DML and DDL changes (events - LRC)
    - propagates events
    - applies changes at destination databases
  - requirements:
    - DB version (>=9.2.0.4)
    - DB must run in archive log mode



- RLS database
  - OS: Linux
  - initial DB version: 9.2.0.5
  - upgrade to 10g (10.1.0.2.0)

Replication using Streams at RLS

- Streams configuration
  - two machines located at CERN
  - streams setup in both directions
  - schema level replication: two schemas



Replication using Streams at RLS

- RLS stress test
  - written in Python; multi-thread application; producer-consumer pattern
  - uses full **RLS application** stack of POOL FileCatalog

FileCatalog

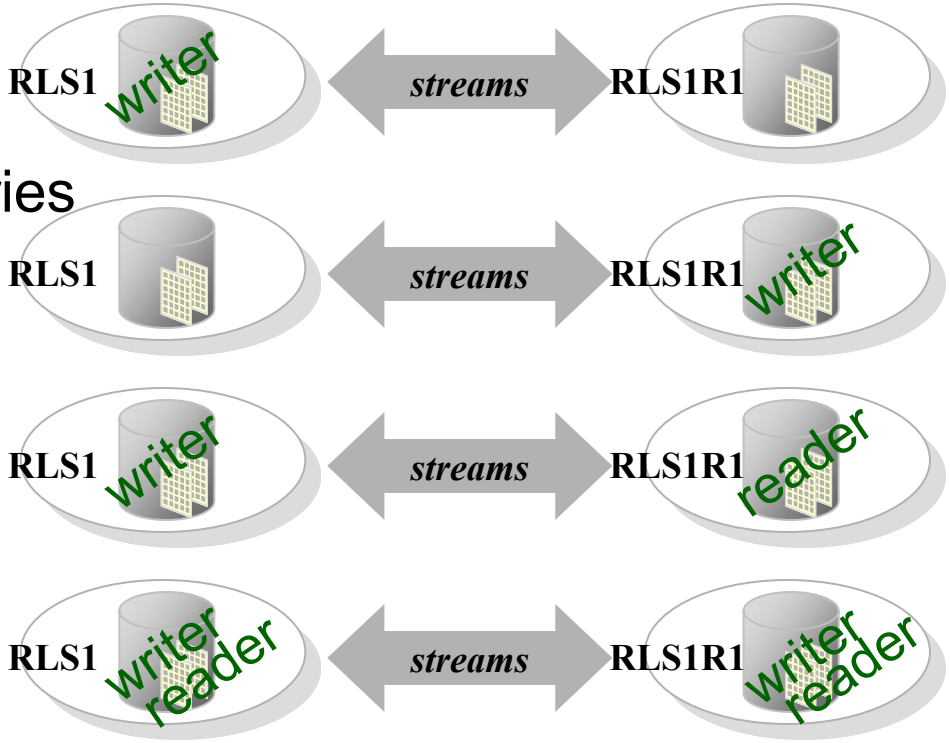
maintains consistent lists of accessible files (physical and logical names) together with their unique identifier (FileID)

resolves a logical file reference (FileID) to a physical file

  - allows to setup **multiple writers and readers** using **one or multiple RLS end-points** which write and/or read RLS data via RLS end-points
  - log file: can be analyzed to produce some statistics
  - designed to **avoid conflicts** – not allow conflict resolution studies

Replication using Streams at RLS

- Stress test run configurations
  - one end-point
  - two end-points
  - about 50.000 entries
  - during 3 weeks



- success results
- conflicts can occur => necessary look at handling conflicts

Replication using Streams at RLS

- Next tasks:
  - conflict handling
  - monitoring
  - add external replication sites