Automating Loader Testing

JCC Consulting, Inc.

Abstract

The JCC LogMiner Loader, developed at JCC, is used in wide range of mission critical applications and acts as the glue for diverse and complex architectures.

We have had very few times when bugs have escaped our labs. That is because we subject all changes to aggressive testing that tortures the product with randomized configurations and sudden failures. It also tests a wide range of environments including multiple versions of the operating system, hardware architectures, companion products etc.

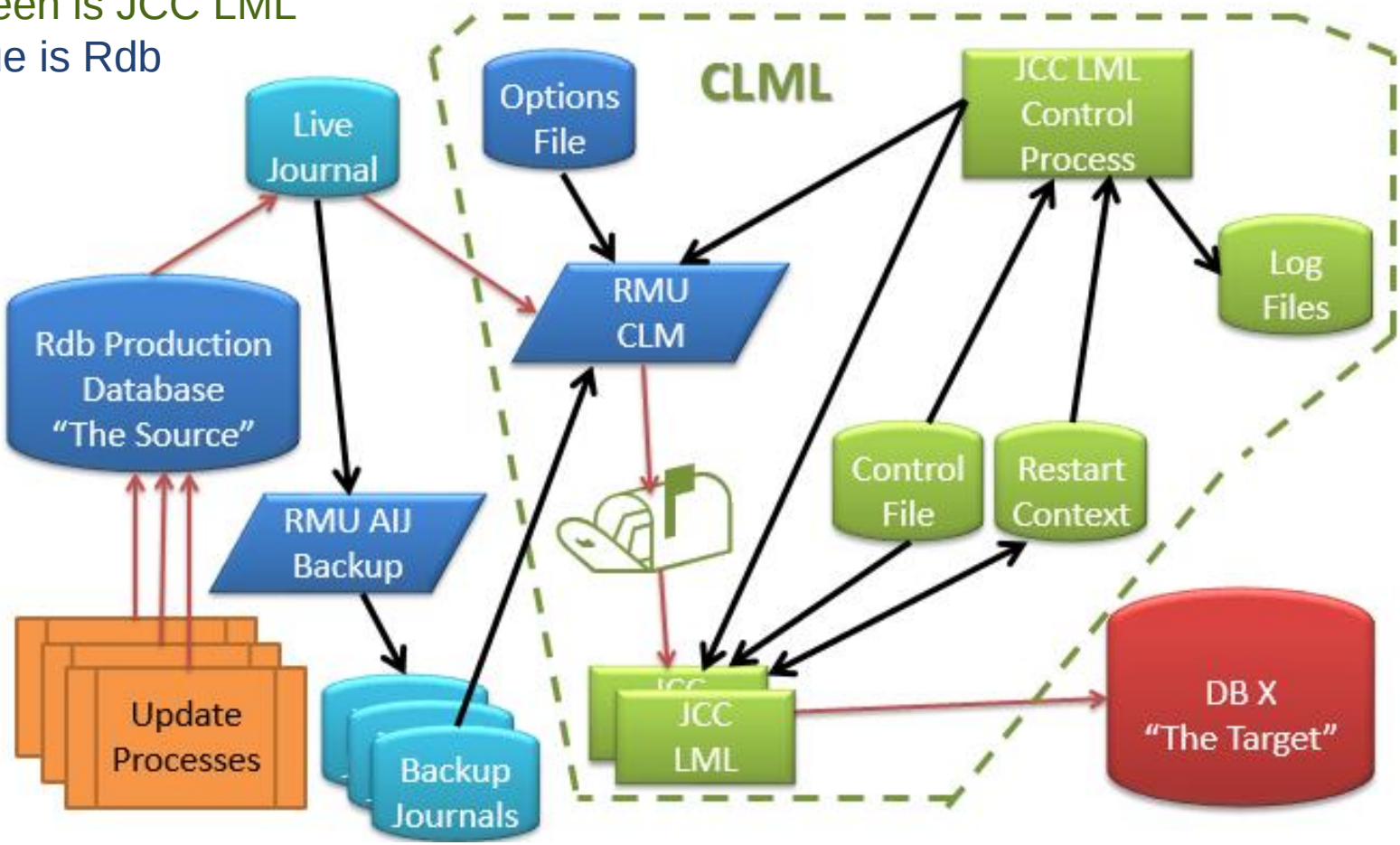
In this presentation, we will discuss how we have achieved a rigorous level of testing.

Who am I?

- Founder JCC Consulting, Inc., 1984
  - Consulting, training, testing, architectures, futures planning, systems health check, performance tuning, problem resolution
- Outreach
  - DECUS sessions and seminars, US & Europe
  - DECUS VMS SIG, Symposia Committee Chair, Board
  - Oracle Open World
  - Jamaica Computer Society
  - Rdb Technical Forums, beginning - present
- Education
  - Fairfield University, BS
  - Virginia Polytechnic University, PhD
    Theoretical Nuclear Physics

What Are We Testing? Middleware…

Green is JCC LML
Blue is Rdb



Challenges of Testing Middle Ware

It's a major combinatorics issue:

- Targets
- Operating Systems
- Transports
- Sources
- Versions
- Hardware Platforms
- Emulators

Target Databases

- Rdb
  - SQL (DSRI)
  - JDBC
- Oracle
  - OCI
  - JDBC
- API
- XML to a file
- Tuxedo
- JDBC
  - Many flavors of targets
  - Multiple JDBC drivers

JDBC Targets

JDBC makes it possible to have, as an end target, any data store that has a JDBC (class 4 or later) driver. We have active regression cycles for all but the DB2 target (which is planned.)

- Rdb
- Oracle
- SQL Server
- MariaDB
- MySQL
- Postgres
- Teradata
- Big Data
- DB2
- …

Operating Systems

- OpenVMS Versions for Source database
  - Alpha and Itanium
  - V7.1
  - V7.3-2
  - V8.4
  - V8.4-2
- Targets databases on
  - OpenVMS (Rdb and Oracle)
  - Linux (Oracle, MariaDB, Postgres, Teradata, etc.)
  - Windows Server (SQL Server of various versions)
  - Alpha Emulators (Rdb)

Transports & Networks

- Use the transport required by the target.
  - DECnet
  - TCPIP
    - Have used SSH to run the Loader across an encrypted link
      - Not part of the Loader kit, establish separately
- Can emulate the effects of a slow TCPIP network
  - Beneficial in ad-hoc testing

Sources

- The JCC LogMiner Loader is dependent on the LogMiner which is built into Rdb. Therefore, Rdb is the only source.
- But
  - Rdb versions from 7.0-9 to whatever is available for field test.
  - Rdb running on all levels of Itanium architecture, old to new.
  - Rdb running on Alphas (we use Alpha emulators these days)
  - We have even developed a two-stage architecture to make it possible to use an Rdb source on VAX.
    - Cluster a VAX and an Alpha
    - Application runs on the VAX
    - JCC LogMiner Loader runs on the Alpha

Rdb Source Versions

- We currently actively test with Rdb versions:
  - V7.0-9
  - V7.1-501
  - V7.2-510
  - V7.2-541
  - V7.3-130
  - V7.3-200
- Use the latest available update or patch
- Have tested on every Rdb version released since the LogMiner was introduced

Versions, Hardware Platforms, Transports, Emulators

- All variants of each of these for recent releases or for something that is critical to a customer.
- Major Loader features
- New Loader features
- Random other Loader features

Combinatorics Review

- The combinatorics are large:
  - Host architecture – 2
  - Hardware or emulators – 3
    - Also test on virtual VMware instance.
  - OpenVMS versions – 5
  - Transports
    - DSRI versions – 1
    - OCI versions – 5
    - JDBC drivers and JAVA versions – many
      - See the JCC BLOG regarding tested JAVA versions
    - Tuxedo – 2
  - Compiler Optimizations - 4
  - Target databases – many
  - Loader parameters and controls – dozens

Methodology

- A regression test starts with building or refreshing one or more databases, source and target(s)
  - Either RMU restore or SQL insert into/select from used to initialize source tables
  - OCI SQL Services used for most Oracle targets to do insert into/select from
  - ODBC SQL Services used by early SQL Server target
  - Home grown JAVA application used for all other targets
    - Allows copy of tables across any two databases reachable by JDBC
    - Allows execution of a series of SQL statements in a single database
  - Views on the source database used when Dbkey is the primary key of a table.
- Execute a random workload against source database
  - Audit triggers so all inserts, deletes and updates are captured

Methodology

- 32 Regression cycles run one or more Loader families.
  - One regression cycle runs 7 different scenarios simultaneously
  - Many cycles do two or three or more scenarios
  - Loader parameters are randomized
  - Improve the chances of uncovering surprises/bugs
  - Replication
  - Auditing
- Detect when Loader session(s) become idle, shut down and continue with the comparisons
- Special cycles to test particular features
  - Skip time interval
  - Transformation of values (version 3.5 feature)
  - Filtering
  - Mode selection (Copy, Static, Continuous)
  - Others

Methodology – Compare Results

- Old Style
  - If both targets are Rdb, use RMU UNLOAD and do diffs (slow)
  - Otherwise, copy final version of tables from one database to the other
    - Use ODBC or OCI for the transport if differences done in target
    - SQL procedure to handle diffs because floating point values can drift in the least significant bit(s) across database platforms
  - Some targets (API, Tuxedo) have special programs to load results back into the source database
  - May require coordinated jobs on two hosts.
- New Style
  - Copy the rows in the target tables to the source Rdb database
    - Use new JDBC code constructed for JDBC reachable targets
  - Use the SQL UNION operator to create a third copy of a table.
    - Eliminates duplicate rows
  - Or use SQL procedures when floating point bits don’t map perfectly.
    - Check within the precision of the datatype less a couple of bits.

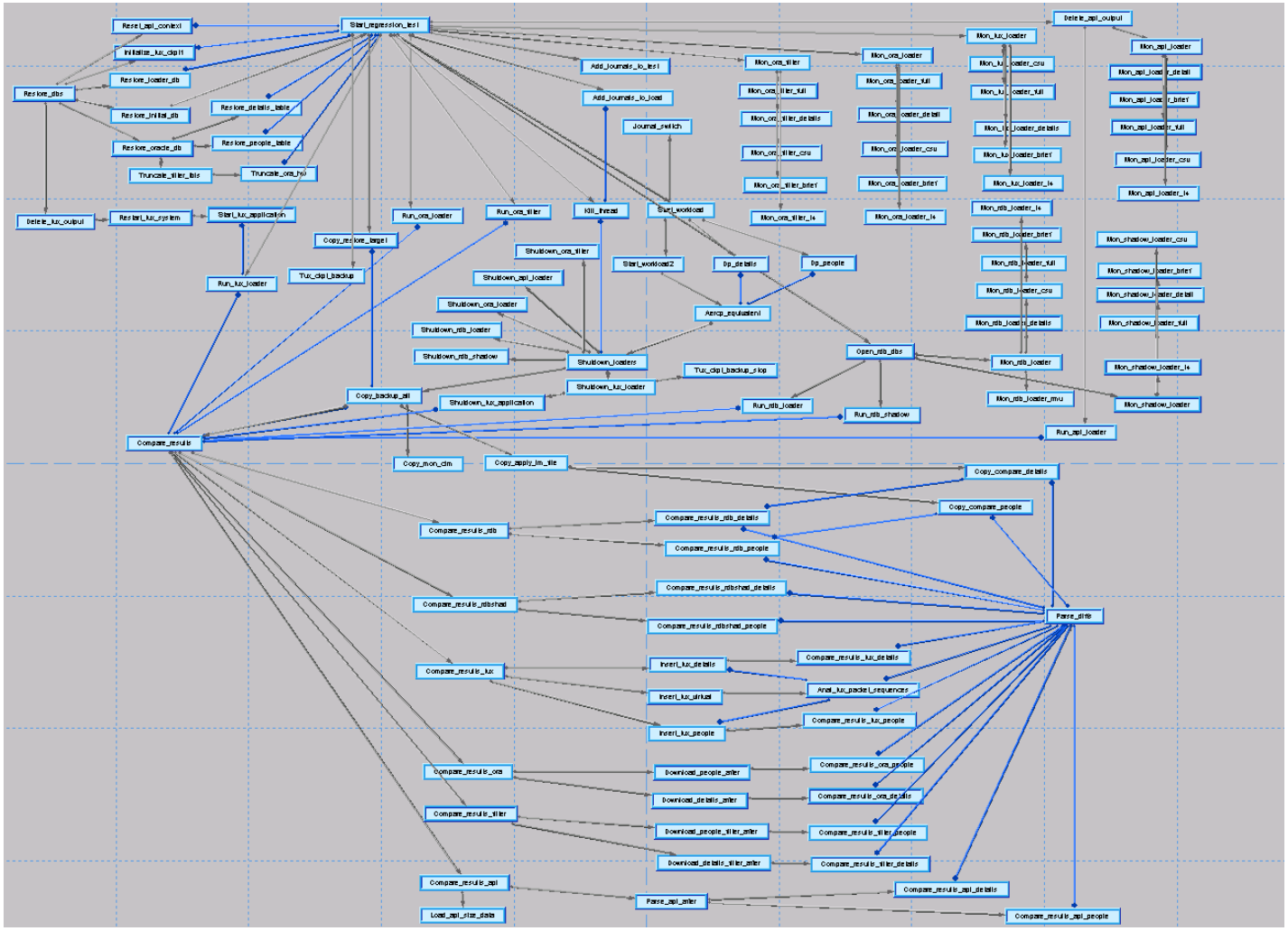
Methodology – Compare Results

- If row counts are the same, or SQL procedure validates compares, then test succeeds
- Stop, if there are differences
  - Sends Email notification so a human can interpret
- If there are no differences
  - Track success in a central Rdb database
    - Columns include all the various parameters assigned to the test
  - Restart the regression cycle

Implementation of Test

- For simple regression cycles we use a DCL procedure which submits batch jobs for parallelism
  - Synchronizes on job completion
  - Captures job completion status as a first level detection of failure
  - Waits on an AERCP check to detect final equilibrium
- For more complex cycles we use a job scheduler

An Early Regression Cycle – 7 Loader Families at Once



Loader Response to Failure Scenarios

- The Loader is supposed to transparently restart from last committed transaction without loss of data
  - Recover from interruptions
  - Cannot recover from lost journals
- What happens if things fail – does restart work properly?
  - Randomly stop CLM and LML processes via FORCEX with a failure exception code and restart
- Do not hesitate to reboot target machines or kill processes.
  - Have done all sorts of dumb stuff to induce failures, some by accident

Tracking What's Done

- Originally used Email to track success
- Summer Intern modified procedures to save regression parameters in an Rdb database
  - Created an internal web page for testing data
  - Allows analysis of coverage of parameters
- Can mine to generate summary reports via email

Some Internal Testing Statistics

LOADER	RDB	TARGET	EARLIEST	MOST_RECENT	TOTAL
T03.05.00	V7.1-501	(null)	2016-07-21 09:30:27	2016-07-21 09:30:27	2
T03.05.00	V7.1-501	rdb	2016-07-20 23:44:54	2016-09-08 05:43:29	317
T03.05.00	V7.2-510	(null)	2016-09-06 16:03:02	2016-09-08 11:38:43	32
T03.05.00	V7.2-510	oracle	2016-07-20 23:51:57	2016-09-08 03:48:14	180
T03.05.00	V7.2-510	rdb	2016-09-06 16:03:02	2016-09-08 11:38:43	32
T03.05.00	V7.2-541	jdbc	2016-07-21 09:30:27	2016-07-21 09:30:27	1
T03.05.00	V7.3-130	oracle	2016-07-20 16:59:10	2016-09-08 10:51:04	1009
T03.05.00ev6	V7.3-200	api	2016-09-05 13:49:18	2016-09-08 11:43:26	64
T03.05.00ev6	V7.3-200	rdb	2016-09-05 13:49:17	2016-09-08 11:43:25	267
T03.05.00i2	V7.3-200	rdb	2016-08-18 13:41:46	2016-09-08 11:14:48	1026
T03.05.00st	V7.0-9	rdb	2016-07-20 16:32:49	2016-09-08 11:15:08	2339
V03.04.04ev6	V7.3-200	api	2016-07-20 16:54:46	2016-09-01 23:30:39	1645

Release Strategy

- We begin regression testing as soon as we can get new capabilities built into a new kit.
- These cycles continue to run while code is completed
- As we discover issues, those fixes are factored into code revisions
- When we have the final code complete and it has been cycling for multiple thousand times we will build a final kit
  - Kits are packaged as ZIP archives or self-extracting ZIP executables.
- This is installed everywhere.
  - All four kits are tested
- Final release only after 30,000 or 40,000 successful cycles

Sharing What We've Learned

- Testing has to be ad-hoc to the software being validated
- Random activities are effective
  - Goes where nobody would ever think of going
  - Discovers bugs that are once a decade events
- Middleware is a complicated beast. We have had to learn all sorts of things about different products
  - Limited lock space for targets, our tables have a million rows each
  - Missing SQL verbs, e.g. TRUNCATE
  - Learning to set up SSH pipes between systems so we can test
- Combinatorics lead to lots of different regression cycles
  - Perforce limited in coverage

Learning More

- The product that we have been testing is The JCC LogMiner Loader.
  - Read about it at http://www.jcc.com/products/jcc-logminer-loader-and-data-pump
- Contact us at Info@JCC.com
- Contact me at Jeff@JCC.com

Join the Conversation

Join the worldwide Rdb community. Send mail to OracleRdb-request@JCC.com.

Include “SUBSCRIBE” in the body of the message.