OpenVMS Rdb Migration Options

Presented by Brett Cameron

Agenda

- **Introduction**
- Some basic considerations
- General approach
- Connectivity options
- How VSI can help
- CDD replacement
- VSI partner solutions
- Summary
- Questions

The Oracle Rdb announcement…

March, 2026

“We regret to inform you that Oracle has decided to discontinue their plans for Rdb on x86-64. While this is not the outcome any of us had hoped for, it does provide clarity and allows us to focus on defining the right path forward for you… “

We understand that the Oracle Rdb product family will remain fully supported for customers with Extended Support contracts through to December 31, 2027. Starting on January 1, 2028, the Oracle Rdb product family on both Itanium and Alpha will continue to be supported for customers with valid support contracts in the indefinite Sustaining Support phase. In this phase, customers (with valid support contracts) will have the rights to all existing patches, but no new patches will be produced (per Oracle Lifetime Support Policy).

The Oracle Rdb announcement

ORACLE SEALS PURCHASE OF Rdb FROM DEC FOR $108m

CBR Staff Writer September 5, 1994

Share this article

Oracle Corp finally agreed the acquisition of Digital Equipment Corp's Rdb relational database business on Friday, taking the assets of the Rdb business, DEC's CDD/Repository – only recently still regarded as one of the company's most strategic software products, and the DBA Workcenter suite of database administration tools. It pays $108m – around one year's annual revenue – in what is DEC's most pressing requirement, cash, and of course gets all the corresponding support businesses. Oracle says it is making every effort to retain the 250 engineers, management and support employees responsible for the development and maintenance of the Rdb database and repository businesses, and it is to create an Oracle New England Development Center, its first such database facility based outside California; it will also add a Colorado Springs Rdb Support Center to its worldwide customer support and services network. Employees abroad will be integrated with existing Oracle operations in those countries. The company did not make clear whether it plans any new releases of Rdb, but says it does intend to make significant investments in the Rdb technology set. Oracle will continue to enhance its capabilities and quality and will extend the existing gateway between Rdb and the Oracle7 parallel database to ensure interoperability between the two, but over time hopes to migrate everybody to a future release of Oracle. It will also complete and bring to market the implementations of Rdb for DEC OSF/1 and Windows NT for Alpha AXP. DEC retains responsibility for all existing maintenance contracts for the next 15 months; thereafter, those that have not expired will pass to Oracle. DEC will continue to offer consulting services for the Rdb product set, and will add consulting support for Oracle7 products. Oracle will also expand the range of products it offers for DEC hardware, in particular putting the Oracle Media Server up on DEC's video server hardware; it will implement Oracle7, Oracle Co-operative Development Environment and Co-operative Applications to DEC's Windows NT for Alpha AXP, to complement the existing OpenVMS VAX, OpenVMS AXP, and DEC OSF/1 AXP versions. Oracle Workgroup Server will go up under the DEC Windows NT for iAPX-86 and Oracle also agreed to resell DEC's transaction processing and data integration lines.

https://www.techmonitor.ai/hardware/oracle_seals_purchase_of_rdb_from_dec_for_108m?cf-view

The Oracle Rdb announcement – some history

- VAX Rdb/VMS first released in 1984 as part of the VAX Information Architecture
    - A relational DBMS designed for high-performance transaction processing on VMS
    - Interoperability with Datatrieve and other tools of the day
    - Proprietary Relational Data Operator (RDO) interface (subsequent support for ANSI SQL)
    - BLR (Binary Language Representation)

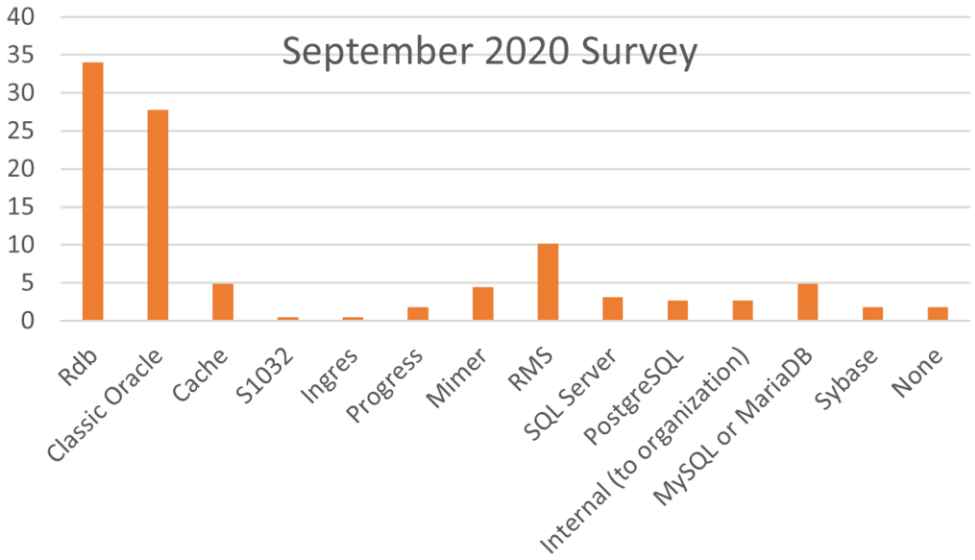
- Included one of the first cost-based optimizers
- Arguably ahead of its time
    - Like quite a few other DEC products of the day

- Key database architect Jim Starkey
    - Relational database pioneer
    - Also created Datatrieve, DEC Standard Relational Interface, and Rdb/ELN (for VAXELN)
    - Invented triggers, BLOBs, multi-version concurrency control, ...
    - Contributed extensively to the architecture of Rdb
    - Subsequently involved with various other databases and is still innovating today

- In a way, any relational database you might use to replace Rdb has been influenced by this pioneering work
- Rdb stood the test of time

The Oracle Rdb announcement – some numbers

- Neither VSI nor Oracle have particularly accurate figures regarding Rdb usage
- Some number of users are likely running unsupported versions
- Based on the results of a 2020 VSI survey it is possible to assume approximately 34% of OpenVMS users use Oracle Rdb



Very likely an upper limit and some are probably purely CDD-related with no use of Rdb for data storage.

- By inference CDD and Trace are also impacted
- And the CODASYL DBMS
- For many users, doing nothing as a consequence of this decision is probably not going to be a viable option…

My background with Rdb

- My first experience with Rdb?
  - Early 1992
  - Just a few weeks into my first job (DEC)
  - No training
  - Thrown straight in
  - Had never used or even seen SQL before (actually, I'd never seen a relational database before)
  - It all seemed strangely intuitive
  - Have used Rdb on and off pretty much ever since

- So, a bit over 34 years ago!

- Favorite command?
  - `$ rmu/alter`
  - Has got me out of a few holes over the years (some of which I dug myself)

So where to from here?

- Regardless of the Oracle Rdb news, database migrations can be a critical part of modernizing your infrastructure…
  - Improving performance and scalability
    - Better handling of growing data volumes or query demands
  - Reducing costs
    - Reducing licensing fees or moving to more economical cloud-managed services
  - Replacement of legacy and/or unsupported/poorly supported products
    - Access to modern/newer capabilities like advanced analytics, high availability, AI integrations, …
    - Moving away from legacy systems nearing EOL
    - Larger pool of skilled resources
  - Aligning with strategic goals like cloud adoption
    - Cloud-first policies
    - Compliance requirements
    - Vendor consolidation
- But database migrations can be complex, and Rdb has some unique features

Agenda

- Introduction
- **Some basic considerations**
- General approach
- Connectivity options
- How VSI can help
- CDD replacement
- VSI partner solutions
- Summary
- Questions

Database selection

- It pretty much all begins here…
  - And it's often not just a technical choice - it's a strategic one that impacts your long-term success and objectives

- Start with a solid target selection process, potentially blending technical and business factors
  - A thorough evaluation of the options helps balance these factors
  - May want/need to include a proof-of-concept

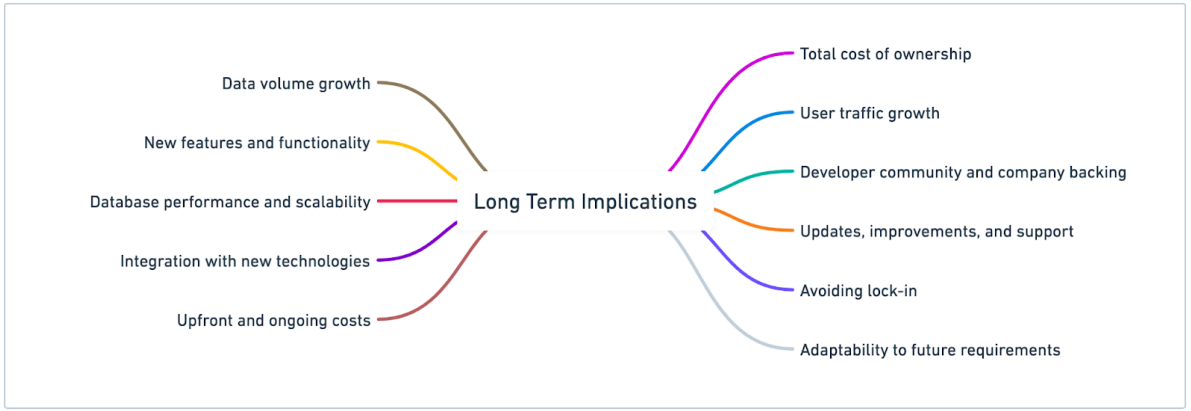
- Follow a structured, phased migration approach
  - Prioritize planning, testing, and risk management

Some technical considerations

- Feature compatibility
  - SQL dialect (it's a fluffy standard)
  - Data types
  - Stored procedures
  - Triggers
  - Indexing capabilities
  - Concurrency model(s)
  - Isolation levels
  - …
- Performance
  - Will it handle the workload (now and in the future)?
  - OLTP versus OLAP
  - Read-heavy versus write-heavy
  - Local access verses network-connected
  - …
- Migration effort/approach
  - Availability of tools to reduce manual rework
- Scalability and high availability
  - Horizontal/vertical scaling options
  - Replication
  - Failover mechanisms
  - Numbers of concurrent connections
  - …
- Security and compliance
  - Built-in encryption
  - Auditing capabilities/requirements
  - Certifications such as GDPR and HIPAA
  - …
- Integration ecosystem
  - Compatibility with existing tools and application stack
  - …

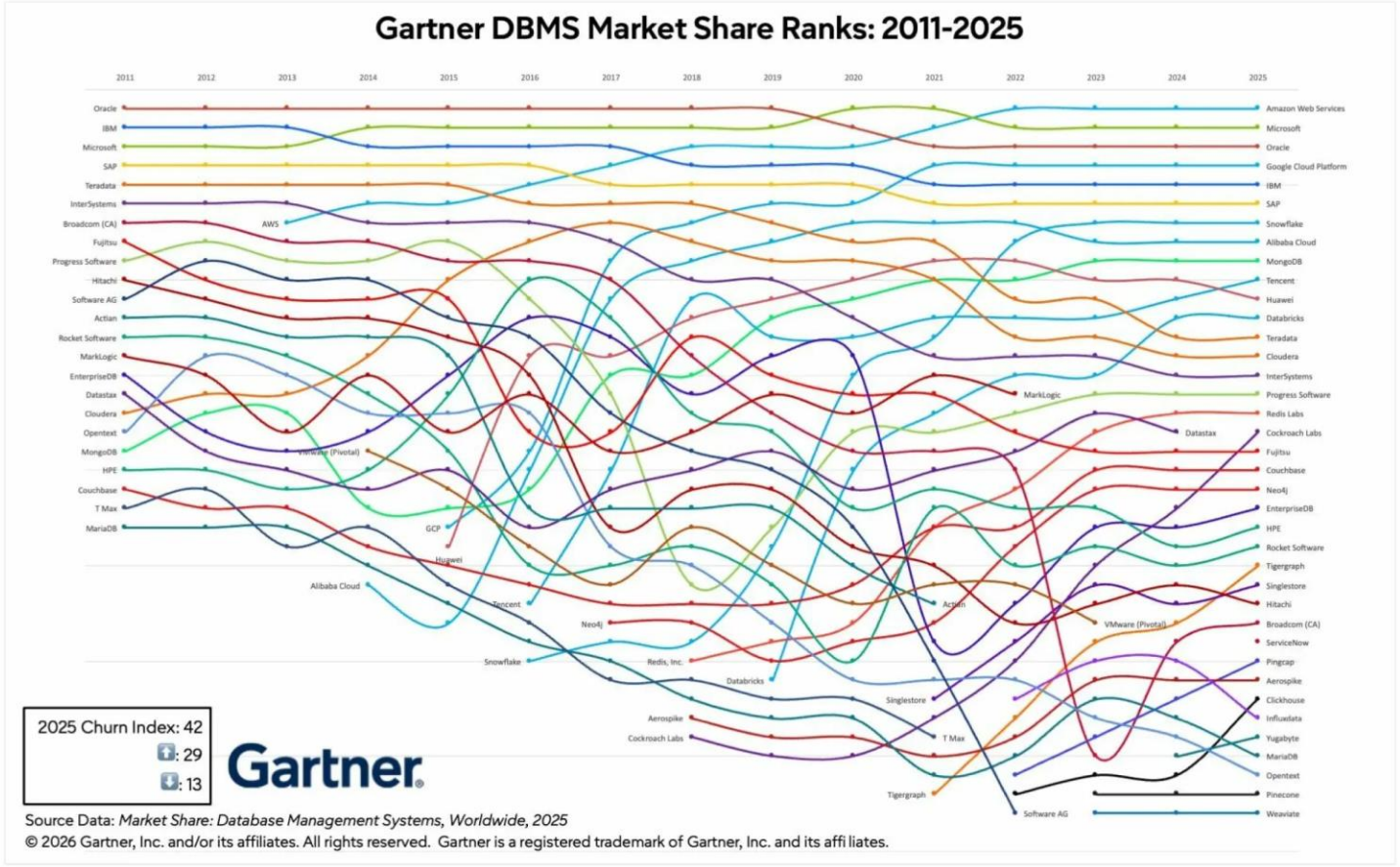
Non-technical considerations

- Corporate policies
  - Preferred vendors
  - Cloud provider alliances or corporate mandates
- Team expertise
  - Do you have existing skills in-house?
  - Are skills readily available?
- Cost structures
  - Licensing model, TCO, …
- Vendor lock-in (or not) and support
  - Ecosystem maturity
  - Community versus enterprise support
  - SLAs



https://blog.bytebytego.com/p/key-steps-in-the-database-selection

What are some options?



The database market is a competitive space ☺

What are some genuine options?

Criteria	SQL Server	PostgreSQL	MySQL/MariaDB	Mimer	SharkSQL	Oracle RDBMS
Supportability	High	High	High	High	Low	High
Maintenance	Moderate	Moderate	Moderate	Low	Low	Moderate
Skills availability	High	High	High	Low	Low	High
Runs on OpenVMS	No	No	Yes (MariaDB*)	Yes	Yes	No
Open source	No	Yes	Yes	No	No	No
Compatibility features	No	No	No	Yes	Yes	Yes

*Currently only runs on OpenVMS Integrity

Two main paths: keeping the database on OpenVMS or moving the database to an alternative platform (Windows or Linux), while keeping applications on OpenVMS where possible.

Agenda

- Introduction
- Some basic considerations
- **General approach**
- Connectivity options
- How VSI can help
- CDD replacement
- VSI partner solutions
- Summary
- Questions

High-level project phases

1. Assessment and planning
   - Inventory of schema(s), impacted code, reports, …
   - Data volume(s)
   - Dependencies
   - Risks
   - Proof-of-concept?
2. Schema conversion
   - Translate DDL, constraints, and other database entities to target platform
   - Typically, can be at least somewhat automated
3. Data migration
   - Cleanse, extract, transform, and load data
   - Big bang, phased/trickle, …
4. Application changes
   - Update queries, connections, and code for compatibility
5. Testing
   - Functional
   - Non-functional (performance, scalability, security, failover, …)
6. Cutover and go-live
   - Final data synchronization
   - Switch traffic to new database platform
   - Be sure to have a rollback plan!
7. Post-migration
   - Monitoring
   - Tuning
   - Decommission and/or archive old platform
   - Relax

Typical strategies

Database migration between relational systems involves moving schemas and data from the source database to the target database while ensuring consistency and minimal application disruption…

Big bang migration:

- The entire dataset is moved in one single event during a scheduled maintenance window
- Simple but requires (potentially significant) system downtime

Trickle/continuous migration:

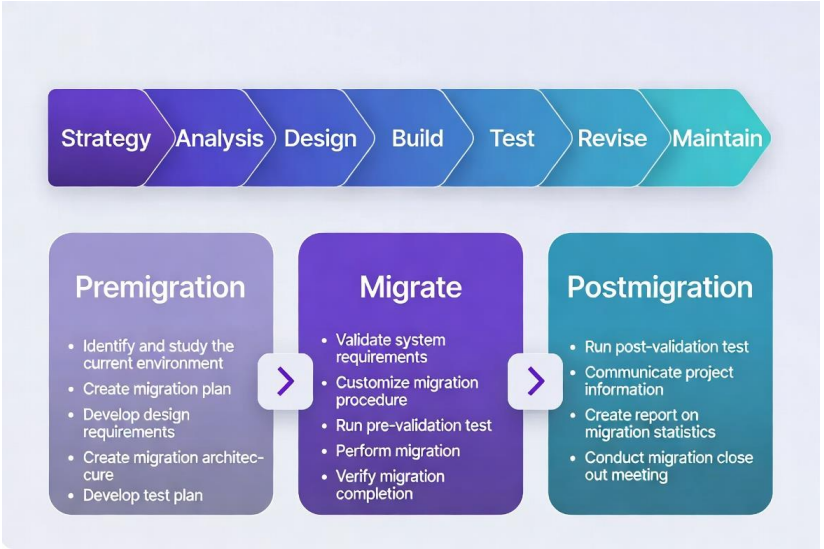
- Data is migrated in phases while the source remains live
- Use change data capture (CDC) to keep systems in sync until final cutover

Zero-downtime approaches are also possible, but this gets complicated, requiring use of replication and redirection techniques to copy data to the new location and to switch applications over with effectively no interruption to service (magic).

Risks and mitigation

Common pitfalls/issues:

- Underestimating schema incompatibilities
- Failure to fully understand fundamentally different database characteristics
- Data loss or corruption
- Prolonged downtime impacting business



Mitigations:

- Use proven tools, proven processes, and proven people!
- Thorough planning and testing
- Appropriate executive sponsorship, strong project management, sufficient budget
- Parallel runs during cutover
- Common sense ☺

Agenda

- Introduction
- Some basic considerations
- General approach
- **Connectivity options**
- How VSI can help
- CDD replacement
- VSI partner solutions
- Summary
- Questions

Connectivity options

In the context of the database options presented earlier, the following table summarizes the currently available connectivity options that can be used on OpenVMS to interact with these databases.

Database	Interface mechanism
Mimer SQL	Native
SharkSQL	Native
SQL Server	FreeTDS API
PostgreSQL	PostgreSQL client API
MariaDB/MySQL	MariaDB client API
Oracle RDBMS	SQL Relay

- It should be noted that Oracle have not stated that they will not port the Oracle RDBMS client to VSI OpenVMS x86-64
- Available JDBC drivers can also be used on VSI OpenVMS to interact with these and other databases

Client APIs

PostgreSQL client for VSI OpenVMS

- Well-proven and current port of the PostgreSQL client for VSI OpenVMS
- Can be used as part of any Oracle Rdb migration to PostgreSQL on Linux (or on Microsoft Windows)
- Client kit includes an embedded SQL pre-processor and API for C
- Would be necessary to develop new or adapt existing pre-compilers to support other programming languages
    - Which could be readily done

MariaDB client

- Similarly well-proven on OpenVMS, but could probably do with an update
- Includes a language-agnostic API that can be used to develop applications in other languages other than C/C++
- Currently no embedded SQL pre-processors
    - Would be possible to develop new or adapt existing pre-compilers from other products

Client APIs - FreeTDS

- https://www.freetds.org/
- An open-source implementation of the TDS (Tabular Data Stream) protocol used by the Sybase and Microsoft SQL Server databases
  - Arguably better support for SQL Server than Sybase in most recent versions
  - Recently added SSL/TLS support for Sybase
  - Working with the maintainers
- Provides libraries that can be used by application programs to interact with these databases
- Includes various utilities
  - Interactive database query tools
  - Data load and unload tools
  - …
- OpenVMS port includes all functionality
- VSI kit also includes embedded SQL pre-processors and associated APIs for COBOL and C/C++ (more on these later)
  - Can be used in place of the Sybase pre-processors
  - With some minor caveats
  - Well-proven (have now been used for multiple projects)

Can be used in place of the Sybase pre-processors, or you can write new embedded SQL code in C/C++ and COBOL on OpenVMS today that can interact with Microsoft SQL Server.

SQL Relay

- Powerful open-source database proxy and connection management solution (http://sqlrelay.sourceforge.net/)
- Sits between applications and databases to eliminate/reduce connection overhead
  - Provides persistent connection pooling
  - Eliminates slow connect/disconnect cycles

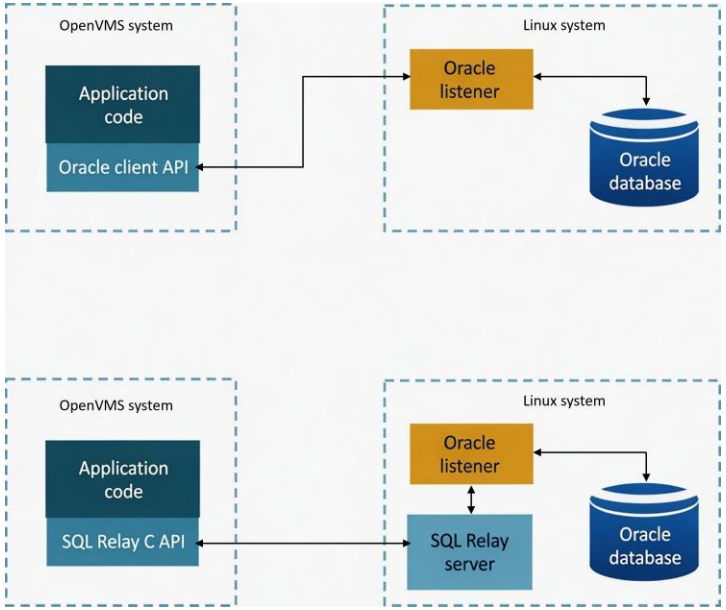
- Can significantly improve performance and scalability of some applications (especially high-concurrency systems)
- Facilitates database access to unsupported platforms

- Includes enterprise-grade features
  - Load balancing and failover
  - Query and session routing
  - Throttling
  - Enhanced security (TLS, Kerberos, IP filtering, query filtering)

- Supports Oracle, SQL Server, PostgreSQL, MySQL, DB2, Sybase, Firebird, SQLite, ODBC, ...
- Can be an ideal technology for scaling legacy or mixed-database environments without the need for major application changes

SQL Relay

- To date we have largely focused on using SQL Relay to provide a replacement solution for the Oracle client on OpenVMS
  - Ported the SQL Relay client C/C++ API to VSI OpenVMS
  - Implemented an OpenVMS-friendly wrapper API
  - Developed embedded SQL pre-processors that can be used in place of Oracle Pro*C and Pro*COBOL
  - Working on a Pro*Fortran pre-processor
  - ...



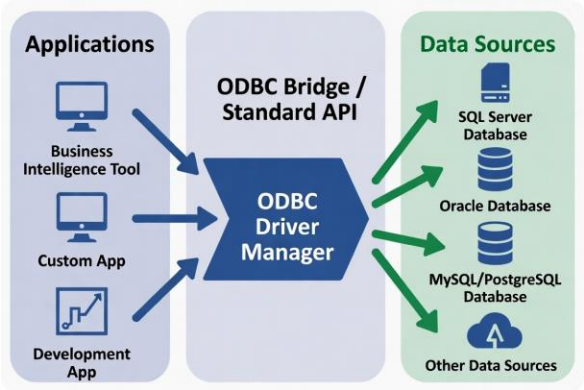
Run the Oracle database on another platform (maybe it's already on another platform) and access it from your OpenVMS applications with minimal code changes (that's the theory).

The database agnostic SQL Relay client API has been ported to OpenVMS (Alpha, Integrity, x86-64). The API communicates with the SQL Relay gateway process (Linux or Windows), which can be configured to support multiple databases, including Oracle RDBMS.

Readily possible to create alternative implementations of the pre-compiler to work in conjunction with other target databases and SQL dialects.

UnixODBC

- http://www.unixodbc.org/
- An open-source implementation of the ODBC API for database connectivity
- The OpenVMS port of UnixODBC includes all functionality provided by the open-source release



- Drivers for several relational databases
  - PostgreSQL
  - MariaDB/MySQL
  - Sybase and Microsoft SQL Server (arguably works better with SQL Server)
  - Oracle RDBMS
  - Mimer
- Also includes interactive SQL and ODBC driver configuration utilities, sample programs, and sample configuration files
- Can easily be used with languages other than C/C++
- Working on a language-agnostic wrapper API
- Can be used in conjunction with tools such as the VSI-developed SQLMOD pre-compiler
- Straightforward to install and configure

In general, UnixODBC incurs somewhat less overhead than SQL Relay and would likely be the preferred approach for situations where there's a choice between the two, however there may be situations where some of the advanced features of the SQL Relay server could be used to good effect to achieve higher levels of scalability.

Agenda

- Introduction
- Some basic considerations
- General approach
- Connectivity options
- **How VSI can help**
- CDD replacement
- VSI partner solutions
- Summary
- Questions

How we can help

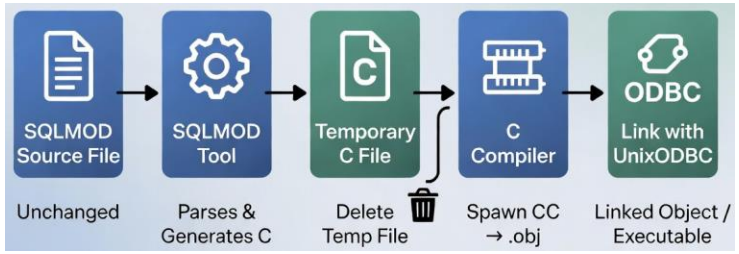
- VSI has a number of tools that are potentially useful to Oracle Rdb migration projects
  - Code conversion tools
  - Development tools
  - Client APIs
  - Gateways
  - Data migration/synchronization
  - ...
- Proven, flexible, adaptable, extensible
- ...
- In some cases, these tools may require modification
  - Every project is different

- And we can provide services to help with migration projects
  - Experienced team
  - Excellent skills with Oracle Rdb and with most target database environments
  - Excellent group of partners

VSI SQLMOD tool

- Compiles Oracle Rdb SQLMOD files into objects that can be linked with COBOL applications
- Resultant programs interact with Microsoft SQL Server via UnixODBC using the FreeTDS ODBC driver
- Successfully parses more than 99% of the 1000+ SQLMOD files tested to date
    - Some minor issues
- Straightforward to enhance to support other languages and database targets
    - Largely generic solution for scenarios involving Oracle Rdb SQLMOD
    - Most modifications likely to be related to variances in SQL dialects and datatype conversions
- Fully integrates with VDD (the VSI Data Dictionary)

- Includes a useful unit-test tool
    - Create test scripts for individual SQLMOD procedure or sequences calls
    - Facilities to compare results obtained for Rdb with those obtained for the target database
    - Performance tests
    - Functionality to generate sample test data for all procedures in each SQLMOD file
    - Integrates with VDD
    - Includes a powerful scripting capability



Database conversion tool

- Tool to assist with replication/migration of data and metadata from Oracle Rdb to another environment
  - Originally written almost 20 years ago, resurrected and enhanced for a recent project
  - Designed to operate in the source OpenVMS environment
  - Simple OpenVMS CLI that supports both interactive and batch modes of operation
- Can be used to replicate database entities such as tables and indexes in a target database based on Rdb metadata
- Can automate the transfer data from the source Rdb database into the target datastore
- Supported targets include…
  - Mimer
  - SQLite (limited applicability)
  - Microsoft SQL Server and Sybase
  - MariaDB/MySQL
  - PostgreSQL
  - RabbitMQ-compliant queues and streams
  - Kafka-compliant streams
- Not a silver bullet (has limitations)…
  - Some Rdb features are intrinsically linked to OpenVMS
  - Some Rdb features do not map all that well to other databases
  - Some functionality not yet implemented!

While we are working on optimizing the performance of direct database-database transfers, it can often be more efficient to pump data into intermediary queues or streams and use consumers on the target platform to populate the target database from those queues or streams. Using streams also makes it possible to replay all or part of the load in the event of any problems.

Capabilities

- Data transfer operations
- Table creation
- Index creation
- Creation of views, constraints, sequences, and triggers
- Substitutions and inclusion of additional data
- Parallel processing capabilities

```
$ run dka100:[cameron.rto]rto.exe
set debug
set target database=sqlserver
attach source [-.db]shop.rdb
attach target "DRIVER=FreeTDS;SERVER=brc-db-1.database.windows.net;PORT=1433;UID=brett;PWD=password;DATABASE=BRC-DB-1"
drop tables
create tables/check=nullable
copy tables/truncate/commit=128 *
exit
```

Embedded SQL pre-processors

- Embedded SQL pre-processors for COBOL and C/C++
- Well-proven
  - Used for many years with several mission-critical applications
  - Most recently used for an x86-64 migration
- Currently somewhat Sybase-centric
  - Can be readily adapted to use with Oracle Rdb

- Work would be required to adapt them for use in an Oracle Rdb context
  - Differences in SQL grammar between Sybase and Rdb
  - Additional datatypes
  - VDD support (see subsequent slides)
  - …
  - This is all readily feasible

```
$ mcr []esqlc.EXE
Usage: EXEC14$DKA200:[SYS0.SYSCOMMON.FREETDS.][BIN]esqlc.EXE;1 [options] input-file

Options:
        -O <output-file>     name for output file
        -D <database>        database to use for statement validation
        -P <password>        password to use when attaching to database
        -U <username>        username to use when attaching to database
        -S <server>          name of server on which database resides
        -z                   output parsed SQL statements and exit
        -w                   stop on warnings
        -v                   verbose mode (output additional information)
        -q                   quiet mode
        -h <function-name>   hook function
        -L <object-library>  object library for compiled SQL statements
        -T <text-library>    text library for COBOL definitions
        -W                   display all warnings
        -r                   relaxed mode

        VSI Embedded SQL Preprocessor ESQL/Cobol Precompiler V0.1 (Beta)
$
```

Vole and Rdb CDC

- Simple extension of the RMS CDC functionality implemented by the VSI Vole log mining solution
  - Same transformation and load engine as RMS CDC
  - Straightforward to bolt on a prototype Oracle R it a spin

This is kind of neat… the database conversion tool can be used to create a shareable image that includes code to extract data from Rdb tables in a fairly optimal manner (on a par with `RMU/ULOAD`).


```
$ run sys$system:rto.exe
attach source sql$database
create image demo/keep
set image "demo.exe"
show counts

salary                          2844047
title                           443308
dept_emp                        331603
employee                        300024
dept_manager                    24
department                      9

set target rabbitmq
attach target "amqp://zaphod:zaphod@10.10.108.242:5672"
publish/prefix="demo"/format=cdc
salary                          2844047         1.630577e+02    1.744196e+04
title                           443308          2.599591e+01    1.705299e+04
dept_emp                        331603          2.005871e+01    1.653162e+04
employee                        300024          1.816464e+01    1.651692e+04
dept_manager                    24              2.001000e-03    1.199400e+04
department                      9               1.000000e-03    9.000000e+03
exit
$
```

Page 1 of 1 - Filter: demo Regex ?

Overview					Messages			Message rates			+/-
Virtual host	Name	Type	Features	State	Ready	Unacked	Total	incoming	deliver / get	ack	
/	demo.department	classic	D Args	idle	9	0	9	0.00/s			
/	demo.dept_emp	classic	D Args	idle	331,603	0	331,603	0.00/s			
/	demo.dept_manager	classic	D Args	idle	24	0	24	0.00/s			
/	demo.employee	classic	D Args	idle	300,024	0	300,024	0.00/s			
/	demo.salary	classic	D Args	idle	2,844,047	0	2,844,047	0.00/s			
/	demo.title	classic	D Args	idle	443,308	0	443,308	0.00/s			

Vole and Rdb CDC

- Perform initial database load utilizing RTO database migration tool (previous slide)
- Intercept subsequent changes using Vole

```
$ run vole$root:[bin]vole.exe
vole> monitor database/changes sys$sysdevice:[cameron.employee-sample-database.db]employee.rdb -
/nowait -
/tables=(department,dept_emp,dept_manager,employee,salary,title) -
/broker="amqp://guest:guest@10.10.108.242:5672"

[11/06/26 22:14:03] successfully connected to broker at amqp://guest:guest@10.10.108.242:5672
[11/06/26 22:14:03] changes will be published to rdb.cdc.topic with routing key prefix "rdb.cdc"
[11/06/26 22:14:03] initialization complete

vole>
```

- Routinely achieve publish rates of > 20,000 records per second using an old HP BL860c i4
- A typical SQL Server or similar modern database implementation can easily handle such numbers
- Net result is a simple and efficient off-platform database synchronization solution
- Code reuse ☺

RDML conversion

- Prototype tool for converting Fortran, COBOL, and BASIC code with embedded RDML to COBOL and embedded SQL and/or direct API calls
    - Showing very considerable promise

```
* Modify the address fields for the specified EMPLOYEES record.
*
&RDB&         FOR E IN EMPLOYEES WITH E.EMPLOYEE_ID = employee_id
&RDB&            MODIFY E USING
&RDB&               ON ERROR
                       MOVE "N" TO success_flag
                       CALL "Error_handler" USING RDB$STATUS, retry_count,
                            success_flag,  lock_error_flag
&RDB&               END_ERROR
&RDB&                  E.ADDRESS_DATA_1 = address_data_1;
&RDB&                  E.ADDRESS_DATA_2 = address_data_2;
&RDB&                  E.CITY           = city;
&RDB&                  E.STATE          = state;
&RDB&                  E.POSTAL_CODE    = postal_code;
&RDB&            END_MODIFY
&RDB&         END_FOR
```

```
* Modify the address fields for the specified EMPLOYEES record.
*
       EXEC-SQL
          update EMPLOYEES E set
                       E.ADDRESS_DATA_1 = :address_data_1,
                       E.ADDRESS_DATA_2 = :address_data_2,
                       E.CITY           = :city,
                       E.STATE          = :state,
                       E.POSTAL_CODE    = :postal_code
          where E.EMPLOYEE_ID = :employee_id
       END-EXEC.

       IF NOT SQLCODE = 0 THEN
                    MOVE "N" TO success_flag
                    CALL "Error_handler" USING RDB$STATUS, retry_count,
                         success_flag,  lock_error_flag
       END-IF.
```

- May not achieve 100% automated conversion
- Expect to be able to do better than 90%
- Various complications associated with use of DB keys
- Some case-by-case considerations
- …

Agenda

- Introduction
- Some basic considerations
- General approach
- Connectivity options
- How VSI can help
- **CDD replacement**
- VSI partner solutions
- Summary
- Questions

CDD replacement

From: Brett Cameron <brett.cameron@vmssoftware.com>
Sent: Tuesday, June 2, 2020 12:39 AM
To: [redacted]
Cc: Brett Cameron <brett.cameron@vmssoftware.com>;[redacted]
Subject: CDD preplacement project

Tatyana,

Below are some initial comments on the "CDD replacement project". Could you please take a read through this and let me know if anything is not clear or needs more detail before going to the team?

Susan, if you have some time could you please do likewise? That will teach you for being excited about the project ☺

Regards,
Brett

"

The CDD replacement project [... your mission, should you choose to accept it]

"

CDD replacement

The aims of the project

We need to create a partial replacement for the Oracle CDD product that addresses most of the requirements of VSI products that depend on CDD, which includes the languages compilers, ACMS, DECforms (development), DATATRIEVE, and others. Ideally there should need to be minimal change to any of these VSI products and the replacement solution should require minimal change from a customer perspective for them to use it with their existing application code.

Develop a replacement for CDD REPOSITORY OPERATOR that is able to parse, store, update, delete, and export data definitions.

The tool should replicate exactly the existing syntax for defining data types and it should implement as closely as possible a yet to be determined subset of CDO commands.

Develop a replacement for CDDSHR.EXE that provides an equivalent API to the existing Oracle CDDSHR.EXE.

- Decouples VSI compilers, ACMS, TDMS, DECforms, and Datatrieve from 3rd party dependency
- Doesn't necessarily do (or need to do) everything that CDD does
- Focusses on most common user requirements

CDD replacement

- Still some work remaining, but looking good
  - Being used/tested by a selected group of customers moving to OpenVMS x86-64
  - Being used by VSI Application Services on several projects
- Largely complete support for CDO language syntax
  - Some reported syntactical and dictionary navigation issues
- Working with COBOL, C, and BASIC compilers
  - Some relatively minor issues with Pascal, Fortran, and DECforms
  - Working with ACMS on Integrity and x86-64
  - Working with TDMS
- Currently focusing on bug-fixes for customer-reported issues
- Includes CDD-to-VDD conversion utilities
- Some DMU support
- No support for RMS file creation (basically this feature just uses FDL, so could be easily hooked in if required)
- …

VSI and field test customers have migrated tens of thousands of CDD field and record definitions into VDD and used those definitions to build applications on OpenVMS x86-64.

Agenda

- Introduction
- Some basic considerations
- General approach
- Connectivity options
- How VSI can help
- CDD replacement
- **VSI partner solutions**
- Summary
- Questions

Some partner solutions

Mimer

- https://www.mimer.com/
- Well-proven high-quality database solution for VSI OpenVMS
- First commercial database to be ported and supported on VSI OpenVMS x86-64
- Includes various Oracle Rdb compatibility features
- Support for Rdb-specific datatypes and syntactical constructs
- Support for modular SQL compilation
- Tools to help users migrate from Oracle Rdb
- ...

Mimer offers direct on-platform replacement with dedicated Rdb migration support, including scripts to export schema and data from Rdb, semi-automatic schema conversion tools to adjust COBOL and Fortran programs, and supports phased migration to x86-64.

Mimer

Sector7

- Powerful solution to move Oracle Rdb users using SQLMOD or embedded SQL from Oracle Rdb to PostgreSQL or Oracle RDBMS
  - Other database targets also possible (Sqlite, ODBC, MySQL/MariaDB)
  - Handle multiple languages
  - Recently added support for RDML conversion and have enhanced CDD support
- Well-proven proven
- Some excellent features, including performance optimizations for cursor operations
- See https://sector7.com/toolset-bundles/rdb-to-oracle-or-postgresql

Some partner solutions

SharkSQL

- A new and innovative, distributed relational database management solution for VSI OpenVMS and Microsoft Windows Server
  - Planned support for Linux

- Impressive set of features
  - Full SQL-99 compliance with partial SQL-2013 support
  - ACID-compliant transactions ensuring data consistency and lost update protection
  - Advanced security measures such as per-table encryption and packet-level security
  - Built-in two-phase commit implementation
  - Intuitive CLI for management tasks such as backups, restores, and configurations
  - MySQL/MariaDB wire protocol
  - …

- Includes various Oracle Rdb compatibility features similar to those provided by Mimer, but arguably more comprehensive

WDB Advanced DataBase Technology BURGER

Some partner solutions & services

JCC Consulting

- Top-notch Rdb skills and experience
- Tools to assist with database migration
  - See https://www.jcc.com/index.php/products/jcc-logminer-loader-and-data-pump

- JCC LogMiner Loader
  - Publishes changes made to data in an Rdb database to specified target in near-real-time
  - Highly configurable, feature-rich
  - Can be used in conjunction with all databases previously mentioned

- JCC Data Pump
  - Companion product to the JCC LogMiner Loader
  - Can be used for initial population of the target, data conversion, or to repair the data after difficulties with downstream processing

Some partner solutions & services

SCI

- Unquestionable Rdb skills
  - Not too shabby with various other databases either ☺

- SCI TranSSend
  - https://sciinc.com/transsend/
  - Near real-time CODASYL DBMS to Relational database replication
  - Changes to CODASYL DBMS captured in journal files and applied to relational target

"... [JCC] LogMiner Loader and TranSSend are both tools that allow you to replicate your database to some other target while you are migrating your application so that when the application is ready to go on the new platform, the data is there immediately."

Agenda

- Introduction
- Some basic considerations
- General approach
- Connectivity options
- How VSI can help
- CDD replacement
- VSI partner solutions
- **Summary**
- Questions

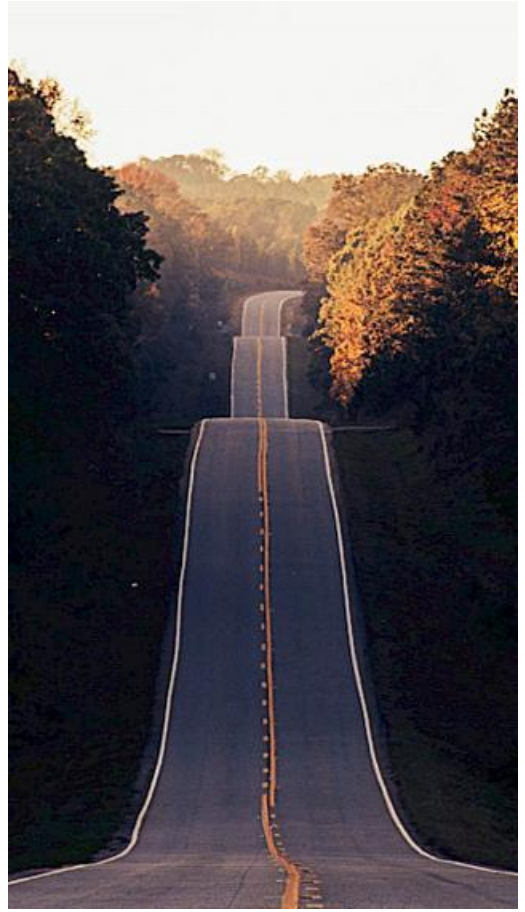
Summary

- Oracle Rdb will not be released for VSI OpenVMS x86-64
- Ongoing/future support for Oracle Rdb on OpenVMS Alpha and Integrity beyond the end of 2027 is limited

- These decisions will significantly impact many OpenVMS/Rdb users
- But there are clear paths forward

- Multiple viable modern database alternatives exist
  - On-platform and off-platform

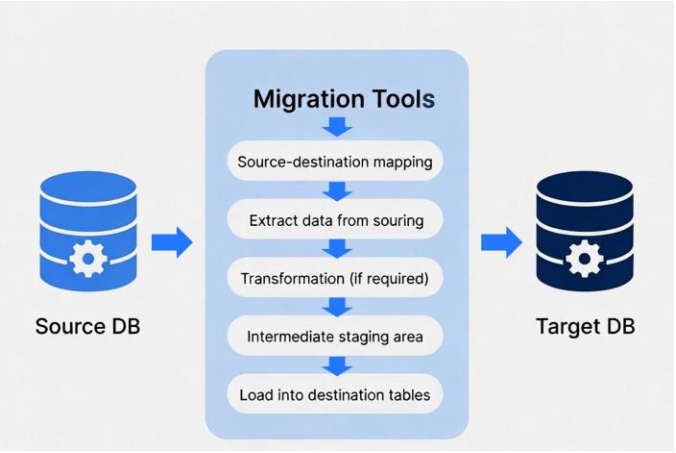
- There are service providers (including VSI) with proven experience to help
  - Referenceable (successful) projects
  - Proven tools, APIs, and service providers are available to support your migration
  - …



Summary

- Database migrations can be large and complex projects
- Every database migration project is different
    - Need to adapt strategies and planning based on data volume, complexity, business goals, ...

- Understand your requirements
- And evaluate your options for optimal fit with your requirements



Summary

- You have options
- You have support

(Modern database) + (proven connectivity) = (a sustainable future for your OpenVMS applications)

- Next steps... with VSI hat on...
  - Complete the VSI Oracle Rdb Migration Assessment and your friendly VSI account manager will soon be in touch!

Questions?