

VNA-SB-010 | Database X to Database Y | Cross-Engine Relational & Procedural Conversion | Redwood Clay

SAP Sybase ASE & SQL Anywhere to Oracle Database 23ai Migration (T-SQL to PL/SQL)

SAP's sunsetting of mainstream roadmaps for Sybase Adaptive Server Enterprise (ASE) and SQL Anywhere has left financial, government, and PowerBuilder-heavy organizations exposed. Our automated T-SQL-to-PL/SQL AST transpiler converts Sybase schemas, temporary table patterns (`#temp``), cursor loops, and triggers into high-concurrency Oracle Database 23ai packages.

Search Queries & Scope Covered: Sybase ASE to Oracle migration | SAP SQL Anywhere to Oracle Database conversion | convert Sybase Transact-SQL T-SQL to Oracle PL/SQL | Sybase ASE 16 end of mainstream maintenance migration | refactor Sybase `#temp` tables to Oracle 23ai

SOURCE LEGACY STACK (STACK X)		MODERNIZED TARGET ARCHITECTURE (STACK Y)	
SAP / Sybase ASE (12.5 / 15.x / 16.0), SQL Anywhere & Watcom-SQL		Oracle Database 23ai / Autonomous Database, PL/SQL & Oracle APEX	
T-SQL TO PL/SQL AST	LOCK ESCALATION BOTTLENECKS	IMPLICIT RESULT SETS	PAIRED UI MODERNIZATION
85-92% Automated	Eliminated (MVCC)	Native 23ai Support	PowerBuilder to APEX

1. Problem Space & Architectural Bottlenecks

- Sybase ASE and SQL Anywhere rely heavily on Transact-SQL (T-SQL) idioms that differ fundamentally from Oracle: ubiquitous `#temp`` tables created on the fly inside stored procedures, implicit multi-result-set returns without `REF CURSOR`` parameters, and page/row lock escalation.
- Naive line-by-line conversion tools translate Sybase `#temp`` tables into DDL `CREATE TABLE`` calls inside Oracle procedures, causing catastrophic library cache latch contention.
- Many Sybase databases are tightly coupled to legacy PowerBuilder or C++ (`CT-Lib`` / `DB-Lib``) client applications that must be migrated in lockstep.

2. Technical Solution & Reference Architecture

- 2.1 T-SQL / Watcom-SQL AST Analyzer & Transpiler:** We parse `sysobjects`/syscomments`` exports to build a complete call graph of all Sybase stored procedures, triggers, rules, defaults, and user-defined datatypes.
- 2.2 Idiomatic Oracle 23ai Refactoring:** Sybase `#temp`` table anti-patterns are refactored into set-based Common Table Expressions (`WITH`` clauses), Global Temporary Tables (GTTs), or Oracle Private Temporary Tables (`ORA$PTT_``). Implicit result sets use `DBMS_SQL.RETURN_RESULT`` so client call signatures remain clean.
- 2.3 Concurrency & Isolation Model Normalization:** Sybase's blocking read locks and `HOLDLOCK`` hints are converted to Oracle's non-blocking Multi-Version Concurrency Control (MVCC), eliminating reader-writer deadlocks.
- 2.4 Automated Data Migration & Row-Hash Reconciliation:** High-speed parallel extraction via `bcp`` / Python Arrow pipelines into Oracle External Tables / SQL*Loader Direct Path, verified by column-level SHA-256 checksums.

3. Artifact-by-Artifact Conversion & Modernization Matrix

Legacy Artifact / Baseline Workload	Modernized Oracle / Cloud Target	Engineering Method & Outcome
Sybase <code>#temp</code> Tables Inside Stored Procedures	Set-Based CTEs (<code>WITH</code>), GTTs, or 23ai <code>ORA\$PTT_</code> Tables	AST Dataflow Refactoring (Zero Runtime DDL)
T-SQL Stored Procedures Returning Implicit Result Sets	PL/SQL Packages with <code>DBMS_SQL.RETURN_RESULT</code> / <code>SYS_REFCURSOR</code>	Signature-Compatible Procedure Compiler
Sybase <code>IDENTITY</code> Columns & <code>@@identity</code>	Oracle 23ai <code>GENERATED BY DEFAULT ON NULL AS IDENTITY</code>	Sequence High-Water-Mark Synchronization
T-SQL <code>@@error</code> & <code>RAISERROR</code> Error Handling	Structured PL/SQL <code>EXCEPTION</code> Blocks & <code>RAISE_APPLICATION_ERROR</code>	Deterministic Transaction Rollback Mapping
Sybase <code>TEXT</code> / <code>IMAGE</code> Columns	Oracle SecureFiles <code>CLOB</code> / <code>BLOB</code> with Deduplication	Parallel Streaming LOB Ingestion

4. Oracle APEX & Ancillary Engineering Toolchain

- Viscosity T-SQL / Watcom-SQL to PL/SQL AST Transpiler
- Oracle SQL Developer Migration Workbench & Oracle SQLcl Liquibase
- Oracle GoldenGate for Sybase (for zero-downtime CDC cutover)
- utPLSQL for automated stored-procedure output parity testing
- Oracle APEX 24.2 (when paired with PowerBuilder-to-APEX UI conversion)

Anonymized Production Proof: Paired frequently with our PowerBuilder-to-APEX modernizations, we have migrated multi-hundred-gigabyte Sybase ASE and SQL Anywhere databases containing thousands of T-SQL stored procedures into Oracle Database 19c/23ai, eliminating nightly batch deadlocks and cutting end-of-month close times by 70%.

5. Principal Architect Q&A

Q: Should we convert our Sybase database and PowerBuilder front-end together or in phases?

A: We support both. We can migrate Sybase to Oracle 23ai first (repointing PowerBuilder’s database profile from `SYC/ASE` to `ORA`) and then convert the PowerBuilder UI to Oracle APEX in Phase 2, or execute a combined vertical-slice cutover.

Q: How do you handle Sybase’s case-insensitive or padded string comparisons in Oracle?

A: Oracle Database 23ai supports native column-level and schema-level collation (`BINARY_CI` / `USING_NLS_COMP`), allowing exact parity with Sybase sort orders without wrapping indexed columns in `UPPER()` functions.

Q: What is included in the downloadable PDF Solution Brief?

A: The PDF Solution Brief includes the 20-point Sybase-to-Oracle 23ai T-SQL translation guide, the `#temp` table refactoring decision tree, and our GoldenGate CDC cutover architecture.