

VNA-SB-004 | Application Stack X to Application Stack Y | Discontinued APEX Plugin Framework to Native APEX 24.2 | Redwood Sandstone

FOEX Plugin Migration: Replace Discontinued FOEX (ExtJS) with Native Oracle APEX 24.2

Enterprises that built heavy back-office applications on the third-party FOEX plugin framework (Sencha ExtJS) are now version-locked on aging Oracle APEX releases. We systematically deconstruct FOEX viewports, grids, forms, and custom ExtJS Dynamic Actions into 100% native, upgrade-safe Oracle APEX 24.2 components.

Search Queries & Scope Covered: FOEX plugin migration Oracle APEX | FOEX discontinued replacement APEX 24.2 | upgrade Oracle APEX with FOEX Sencha ExtJS plugins | convert FOEX Grid to APEX Interactive Grid | remove COM.FOEX plugins from Oracle APEX

SOURCE LEGACY STACK (STACK X)		MODERNIZED TARGET ARCHITECTURE (STACK Y)	
FOEX Plugin Framework (Sencha ExtJS Wrapper on Legacy APEX 5.x / 18.x / 19.x / 20.x)		100% Native Oracle APEX 24.2 (Universal Theme 42, Native Grids, Dynamic Actions)	
THIRD-PARTY PLUGIN LOCK-IN	APEX UPGRADE PATH	PAGE PAYLOAD REDUCTION	PL/SQL BACKEND REUSE
0% (100% Native)	Unlocked to 24.2+	65-80% Smaller JS	95%+ Unchanged

1. Problem Space & Architectural Bottlenecks

- The FOEX Plugin Framework wrapped Sencha ExtJS inside Oracle APEX to build desktop-like multi-pane grids in the APEX 4.x/5.x era. Following corporate acquisitions and product discontinuation, FOEX is a dead-end dependency.
- Every time an organization attempts to upgrade Oracle Database or Oracle APEX to 23.2 / 24.2, broken FOEX plugin metadata, deprecated ExtJS JavaScript APIs, and custom CSS overrides halt the upgrade.
- FOEX pages bypass standard APEX Universal Theme rendering, loading multi-megabyte ExtJS bundles and proprietary AJAX handlers that cannot use modern APEX features like Native Approvals, AI Assistant, or Template Components.

2. Technical Solution & Reference Architecture

- 2.1** APEX Export Metadata & FOEX Plugin Scanner: We parse your application export SQL/YAML via SQLcl to inventory every `PLUGIN\_COM.FOEX.\*` region, item, and dynamic action across all pages, scoring each page by ExtJS customization depth.
- 2.2** Viewport & Splitter Replacement with Universal Theme 42: Complex FOEX Viewport, Tab, and Accordion containers are replaced with native Universal Theme 42 CSS Grid/Flexbox sub-regions, collapsible drawers, and modal dialogs.
- 2.3** FOEX Grid & TreeGrid to Native Interactive Grid: FOEX Grids are converted to native APEX 24.2 Interactive Grids with custom `apex.region().call('getViews', 'grid')` configurations, preserving frozen columns, summary rows, and inline editing.
- 2.4** Zero-Disruption Coexistence & Page-by-Page Cutover: Using APEX session-sharing and AMP navigation routing, we migrate high-complexity FOEX applications module-by-module without freezing ongoing business operations.

2.5 Query Fan-Out Rationalization & Runtime/Inference Filtering: Instead of migrating cloned reports and DataWindows 1-for-1, our AST SQL parser clusters queries by their canonical core JOIN graph, builds governed Oracle 23ai views only for those core queries, and generates dynamic WHERE clauses, faceted filters, and ad-hoc report variations at runtime and via AI inference (APEX 24.2 Smart Filters & Select AI).

3. Artifact-by-Artifact Conversion & Modernization Matrix

Legacy Artifact / Baseline Workload	Modernized Oracle / Cloud Target	Engineering Method & Outcome
FOEX Viewport / Border Layout / Panes	Universal Theme 42 Region Display Selector & CSS Grid	Responsive Native Layout Refactoring
FOEX Grid / Editable Grid Plugin	Native APEX 24.2 Interactive Grid + Server Processes	Metadata Remapping & Native DML Wiring
FOEX Form & Field Plugins (Combo, Lov, Date)	Native APEX Form Region, Popup LOV & Date Picker	Item Type Conversion + Validation Preservation
FOEX Dynamic Actions (Server/Client ExtJS Events)	Native APEX Dynamic Actions + <code>`apex.server.process`</code>	Clean JavaScript & PL/SQL Callback Refactoring
FOEX Master-Detail Linked Panes	Native APEX Master-Detail Interactive Grids	Declarative Parent-Child Region Binding
Cloned Reports / Queries Differing Only by WHERE Clause or Group-By (Query Fan-Out)	Canonical Core SQL View + APEX 24.2 Runtime / AI Inference Filters	AST Join-Graph Clustering (75-90% Report Count Reduction)

4. Oracle APEX & Ancillary Engineering Toolchain

- Oracle APEX 24.2 Universal Theme 42 & Template Components
- Oracle SQLcl Liquibase for page-level YAML/SQL export analysis and diffing
- Viscosity AMP for seamless cross-application session sharing during phased FOEX retirement
- Playwright / Selenium automated UI regression suites

Anonymized Production Proof: For Ivy League university administration systems and major county justice/prosecutor case management platforms locked on legacy FOEX versions, our engineering team cataloged and converted hundreds of FOEX plugin regions into 100% native Oracle APEX 24.2, eliminating third-party subscription fees and restoring one-click APEX patching.

5. Principal Architect Q&A

Q: Can we migrate off FOEX incrementally without rewriting the entire application at once?

A: Yes. By configuring Shared Session Cookies between your existing FOEX app and a new Native APEX 24.2 companion app in the same workspace, users can navigate seamlessly between legacy and modernized pages while we convert module by module.

Q: Do we have to rewrite our underlying PL/SQL packages when removing FOEX?

A: Almost all core PL/SQL business logic remains intact. Only the thin FOEX JSON/AJAX wrapper calls are refactored to standard ``apex_json`` or native Interactive Grid DML handlers.

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

A: The PDF Solution Brief includes the complete `COM.FOEX.\*` plugin-to-Native-APEX mapping catalog, SQL queries to audit your workspace's FOEX footprint in 60 seconds, and a phased coexistence architecture.