

BoxLang Enterprise Playbook

A practical guide to reducing ACF licensing costs, modernizing CFML applications, and evaluating BoxLang without a risky rewrite — designed for CTOs, CIOs, and engineering leaders.

Run a Readiness Assessment

Start a 30-Day Pilot

Why ACF Customers Are Evaluating BoxLang

BoxLang offers a practical modernization strategy: reduce licensing pressure, preserve CFML knowledge, and move toward a modern JVM runtime — without a risky rewrite.

Reduce Licensing Pressure

Challenge the cost of proprietary ACF licensing across all environments.

Preserve CFML Knowledge

Keep your team's expertise and existing business logic intact.

Modern JVM Runtime

Move to a flexible, cloud-ready, open-ecosystem platform.

Incremental Path

Evaluate with evidence before committing to migration.

The Adobe ColdFusion Licensing Problem

Guaranteed Savings

35–50%

Licensing cost reduction compared to Adobe ColdFusion — a clear business case for evaluation.

Hidden Costs Beyond Production

Licensing pressure extends to QA, staging, CI/CD, disaster recovery, and cloud expansion. When runtime licensing influences architecture decisions, it becomes a business constraint — not just a technical one.

Licensing Question	Why It Matters
Production license cost baseline	Establishes the real renewal exposure
Non-production environment costs	Reveals hidden operational friction
Three-year renewal exposure	Turns licensing into strategic planning

Status Quo Is Not Free

Staying on ACF may feel safe, but risk accumulates in renewals, upgrades, talent, security, and modernization. The decision is financial, operational, and strategic — not just technical.

License Dependency

Renewal pressure and limited negotiation leverage

Cost Creep

Higher costs as environments, servers, and workloads grow

Talent Risk

Harder to attract developers to legacy CFML-only stacks

Innovation Drag

Harder to adopt cloud, containers, serverless, and AI workflows

Security Risk

Delayed patching, older runtimes, and dependency exposure

Upgrade Risk

Large jumps between ACF versions become increasingly painful

❏ Standing still is not free. It moves cost into future renewals, deferred upgrades, and modernization delays.

Why Not Just Rewrite Everything?

ColdFusion applications contain years of business logic, integrations, edge cases, and undocumented behavior. A full rewrite introduces high cost, timeline risk, regression risk, and business disruption.

Stay on ACF

Familiar platform, low immediate disruption — but licensing pressure, vendor dependency, and limited modernization flexibility.

Rewrite in Another Stack

Clean-slate architecture — but high cost, long timeline, regression risk, and loss of CFML knowledge.

Evaluate BoxLang

Incremental modernization, CFML knowledge preserved, potential licensing reduction, lower rewrite risk.

BoxLang vs Adobe ColdFusion: Strategic Comparison

BoxLang is not just a runtime replacement — it is a modernization bridge with lower licensing pressure and greater deployment flexibility.

Dimension	BoxLang	Adobe ColdFusion
Pricing	Open source core + transparent subscriptions	Proprietary licensing, higher cost
Licensing Savings	35–50% reduction vs ACF	Current baseline cost
Runtime	Server, CLI, Docker, serverless, JSR223	Primarily web application runtime
Resource Efficiency	Lower memory and disk footprint	Higher resource usage
Innovation Cycle	Frequent releases, active evolution	Major releases every few years
Ecosystem	ForgeBox, Ortus tooling, open modules	Limited third-party ecosystem

Performance & Resource Efficiency

Internal benchmarks comparing Adobe ColdFusion 2025 and BoxLang on CommandBox show significant differences. Customers should validate with their own workloads.

85–88%

Faster Cold Startup

~30–38 sec vs ~4 min 12 sec

95%

Smaller Engine

8.5 MB vs 169 MB on disk

83%

Lower Idle RAM

128 MB vs 768 MB minimum

69%

Faster CFC Creation

300 ms vs 964 ms (1M creations)

Brad Wood's public benchmark shows BoxLang Query of Queries as 5× faster than Lucee and 17× faster than Adobe ColdFusion on average — with open source test code on GitHub for full reproducibility.

Compatibility & Migration Assessment

The first step is not migration — it's evidence. A controlled assessment replaces assumptions with technical facts across three layers.



Four Migration Paths

BoxLang adoption should match your risk profile — from a structured assessment to a full enterprise exit strategy.



Path 1: Readiness Assessment

Review ACF environment, licensing, compatibility, and migration risk. Ideal before a renewal decision.



Path 2: 30-Day Pilot

Validate BoxLang with real code, real dependencies, and real performance comparisons in a controlled environment.



Path 3: Phased Migration

Prioritize applications by business value and complexity. Reduce ACF footprint progressively over time.



Path 4: Enterprise ACF Exit

Full portfolio audit, governance, training, and executive-aligned modernization program.

Beyond Migration: An AI-Ready Development Ecosystem

BoxLang delivers AI capabilities on two fronts — building AI into apps and using AI to build them.

BOXLANG AI

Unified AI Framework

One API across OpenAI, Claude, Gemini, AWS Bedrock, DeepSeek, Ollama, and more. Build agents, RAG systems, vector search, multimodal workflows, and local AI — without rebuilding your stack.

SKILLS HUB

AI Agent Skills Ecosystem

Reusable, audited skills for BoxLang, ColdBox, TestBox, CommandBox, CFML modernization, testing, CI/CD, and Ortus standards. Install into AI coding agents to improve consistency, accelerate onboarding, and standardize team workflows.



Strategic Value: Preserve existing applications while gaining a practical foundation for AI-enabled products, modern workflows, and more productive engineering teams — beyond licensing savings.

Next Step: BoxLang ACF Readiness Assessment

Before renewing Adobe ColdFusion by default, evaluate BoxLang with evidence. A structured assessment gives CTOs and CIOs the clarity to decide responsibly.

Licensing Review

Current ACF footprint and 35–50% savings potential

Compatibility Analysis

CFML syntax, ORM, queries, scheduled tasks, integrations

Performance Baseline

Startup, memory, QoQ, CFC creation vs ACF

Executive Recommendation

Pilot, remediate, phased roadmap, or stay on ACF

"We should evaluate whether BoxLang can reduce cost and risk before our next ACF decision."

[Run a Readiness Assessment](#)

[Learn More About BoxLang](#)