

Adobe ColdFusion: Past, Present and Future

Gartner RAS Core Research Note G00165146, Mark Driver, 13 May 2009, R3112 06252010

A wide range of IT organizations has leveraged ColdFusion at one time or another over the last decade. While some have migrated from the toolset in recent years, many are rediscovering this technology and its potential role in a modern Web IT strategy.

Key Findings

- ColdFusion version 8 introduces a number of new features (for example, significantly increased performance, PDF, Flex and .NET integration, server monitoring and Ajax) that will motivate users to upgrade from older, out-of-date versions.
- Many previous ColdFusion developers are rediscovering the toolset after having invested in alternative technologies (for example, Java) with limited success.
- ColdFusion is nearly unique in the industry because it provides a balance of ease of use and advanced features that can support “entry level” business unit application developers, as well as “advanced” centralized IT development teams.
- ColdFusion will enjoy small market growth in the next several years, but will lose relative market share to industry-leading platforms, such as Java, Microsoft .NET and PHP.
- Investments in ColdFusion technology are safe for the next five years and potentially longer.

Recommendations

- Consider ColdFusion to be a viable option for continued investment, as well as new development initiatives for at least the next five years.
- Consider including ColdFusion within a larger Web application development (AD) technology portfolio, including Microsoft .NET, Java and Adobe technologies.
- Expect decreased future availability of third-party ColdFusion expertise in most market geographies; instead, be prepared to train developers in-house (for example, expand from Java skills), rather than hiring ColdFusion skills “off the street.”

WHAT YOU NEED TO KNOW

ColdFusion is a well-established Web AD tool that excels at delivering robust Web applications with exceptionally high developer productivity levels. While we believe the customer base for ColdFusion has held steady over the last decade, it has seen a consistent uptick in recent years. We believe the ColdFusion developer base will remain consistent during the next several years, but will be outpaced by the growth of competing technologies (for example, Java and Microsoft .NET). Given ColdFusion's loyal user and continued support and investment by Adobe, we believe that strategic commitments to ColdFusion are safe for at least the next five years — very possibly longer. In short, today ColdFusion is a safe investment and a solid technology option within the portfolios of many IT organizations, as long as they accept the proprietary nature of the toolset and its tight link to Adobe.

ANALYSIS

Past

First released in 1995, ColdFusion was the first Web application. There remain at least 500,000 (some estimates range as high as 789,000) active ColdFusion developers worldwide. We've seen limited adoption of ColdFusion by new developers in recent years, but the existing community remains tightly entrenched and loyal to the toolset, with existing customers steadily increasing the number of deployed ColdFusion servers and applications.

Present

Adobe reports that the newest edition (version 8) is enjoying an uptick in license sales; this growth is a strong testament to the quality and features delivered in version 8. We believe much of the growth is driven by users on older, out-of-date and unsupported implementations who are now finding sufficient value to upgrade their investments. However, we believe the majority of this growth is confined to existing and former ColdFusion developers who are expanding existing investments and rediscovering the toolset.

We expect to see a slight uptick in license revenue for ColdFusion during the next several years, but even this growth will not keep pace with the advance of competing technologies. The end result will be a smaller percentage of the overall developer market share for ColdFusion year over year. We do not, however, expect a rapid migration from ColdFusion among current customers in the short term. The volume and subjects of Gartner client inquiries related to ColdFusion indicate that most developers who could have

otherwise easily migrated from ColdFusion have done so by now. Those who continue to use ColdFusion typically report that they do so today because:

- They have a large, installed code base that works well; migration efforts to alternative technologies cannot be financially justified.
- They have a pool of developer resources who know ColdFusion well and can deliver applications quickly and efficiently.

Interestingly, we see a number of users who migrated away from ColdFusion in recent years now rediscovering the toolset, many coming back to the strong productivity features of ColdFusion after struggling with Java technology. Today, the health of the ColdFusion developer community is much stronger than the larger IT industry may realize. A focus on .NET, Java and PHP has overshadowed ColdFusion to some degree, yet the community remains active and loyal to this toolset as well.

Future

Given its plateaued market share, ColdFusion, on first glance, would seem to naturally fit into a typical "containment" strategy (see Note 1). On inspection, however, we draw our recommendation for ColdFusion short of a "containment" classification, insofar as there remain compelling scenarios wherein ColdFusion can provide unique value that is not fully addressed by any competing alternative technology. Most notably, ColdFusion is unmatched by any competitor for ease of use and technical capabilities. When we combine this with cross-platform deployment, and significant integration into both Java and .NET, ColdFusion stands out as a compelling solution for many IT challenges. It is likely that ColdFusion will fall into a containment classification sometime beyond the next five years, but the time frame beyond that horizon is currently unknown.

A clear goal has emerged within the ColdFusion developer community in recent years. Many developers understand the tenuous nature of any proprietary Web technology in competition with more open technologies such as Java, PHP and Python. In an effort to establish ColdFusion as a more "open" and "safe bet" for IT investments, a number of open-source initiatives have emerged in recent years. Today, both Railo Technologies (www.railo-technologies.com) and Open

BlueDragon (www.openbluedragon.org) are established open-source alternatives to Adobe's ColdFusion ML engine. Railo recently became a JBoss project, and is being supported and extended to include tag-based access to JBoss features. One potential downside in multiple ColdFusion ML engines is the possibility of fragmentation; in an effort to avoid this, Adobe and key members of the ColdFusion community have created an advisory committee (www.opencfml.org) to establish a standard language definition for ColdFusion ML, and to denote the boundaries between the core ColdFusion ML language and product extensions.

We believe that increasing interest in support for open-source implementations of the ColdFusion ML engine will provide a stronger level of investment protection for ColdFusion technology overall. However, we do not believe these efforts are sufficient to provide a 100% vendor-independent and compatible platform for typical application efforts. Instead, ColdFusion ML — the core on which all ColdFusion applications are based — will be increasingly standardized, but ColdFusion applications built using a wide variety of modules and extensions will remain largely dependent on Adobe's technology. Developers who plan ahead and build systems with compatibility in mind can reduce the switching cost when moving to an open-source implementation, but should not consider this as a turnkey solution in decoupling dependence on Adobe's technology for existing code bases.

We also believe ColdFusion will play a pivotal role in Adobe's Flash Platform (www.adobe.com/flashplatform). Developers with investments in Flash, Flex, Air, BlazeDS and PDF Forms will find ColdFusion an integral supporting player in this technology family.

Finally, one scenario that stands out as particularly strong for ColdFusion investments lies among IT organizations with strong commitments to Java Platform, Enterprise Edition (Java EE) platform technologies. The No. 1 complaint Gartner receives related to Java development, both from beginners and seasoned veterans, is that the technology is too complex to develop and maintain many typical IT solutions. This concern is particularly acute for small- to moderate-scale projects where time to market and developer productivity outweigh long-term strategic freedom and flexibility. In these efforts, ColdFusion provides a potential wrapper around the complexity of Java, providing Web developers access to the power of the Java platform via the productivity of a fourth-generation language (4GL). This scenario applies most obviously to IT organizations with existing ColdFusion skills, but may also be compelling for new users as well.

Product Strategy

ColdFusion aims to support Web AD efforts with a highly productive developer experience, coupled with advanced features for next-generation IT solutions. In this effort, it largely succeeds at the cost of proprietary lock-in. ColdFusion is based on Java and is deployed on top of industry-standard Java EE servers, but abstracts this technology and provides a rich wrapper that fully shields the developer from the complexity of the underlying Java technology. ColdFusion also focuses on strong integration into Microsoft software that enables ColdFusion to coexist with the broader Microsoft technology ecosystem (for example, Microsoft Exchange Server integration). Finally, ColdFusion is a member of Adobe's own Web platform technology suite, and includes strong integration with PDF forms and files, as well as the Flex rich Internet application (RIA) toolset.

Strengths

- Easy learning curve, combined with advanced features for both architecturally simple and moderately complex Web applications
- Strong integration with Java and Microsoft .NET
- Support for Windows, Mac OS X, Unix and Linux
- Strong integration with Adobe PDF and Flex technologies

Challenges

- Declining overall market share compared to native Java, .NET and PHP solutions
- Proprietary technology with lock-in to one vendor (Adobe)
- Limited third-party ColdFusion expertise in most major geographical markets

Competitive Analysis

Today, ColdFusion faces competition from a number of areas. Most directly, Java and .NET software platforms both overlap with ColdFusion functionality. Consequently, many organizations that have moved to these two de facto platforms have reduced

or eliminated their investments in ColdFusion over the last several years. However, both platforms have much broader scopes than ColdFusion, while it remains squarely focused on Web-browser-based applications only. Interestingly, we have seen a number of prior ColdFusion customers return to the toolset in recent years after discovering the challenges of adopting and managing the deep technology stack within .NET and Java. Today, this is made technically easier, given the strong integration features between new versions of ColdFusion and these two platforms.

ColdFusion competes heavily against open-source dynamic (for example, scripting) languages, most notably PHP. While ColdFusion contains more-productivity-oriented features out of the box, more-open technologies like PHP that are not proprietary to one vendor have drawn a significant measure of the historical ColdFusion developer base. Finally, virtually all 4GL development toolsets support some degree of Web application deployment. As a proprietary 4GL, ColdFusion shares many of the same market dynamics with these technologies (for example, Delphi, PowerBuilder and others).

Consider This Product When

- You need to deliver robust Web applications, and alternatives such as .NET or Java are overkill
- You are invested in Java and Java EE, and need developer productivity to better your investment in Java
- You need to augment an otherwise heavily Java-centric IT strategy with a Web application layer that provides a higher level of developer productivity; integration with Java is particularly tight since ColdFusion applications are literally compiled into Java code for runtime execution

Note 1

Containment Strategy

In a containment strategy, we would advise that current customers limit their investments, maintain existing applications, and avoid new development projects altogether.

- You need to augment an otherwise heavily .NET-centric IT strategy with a platform that can be deployed to Linux and Unix, while retaining tight connectivity and integration with the Microsoft technology ecosystem
- You need to integrate Web applications with online Adobe Forms, or provide back-end processing to Flex and Air RIAs

Consider Alternatives When

- As part of a long-term IT architectural road map, you are focused on open technologies and on reducing lock-in to proprietary vendor technologies
- You are sufficiently satisfied with the Web development technologies supplied in either Java or .NET, and prefer to reduce the number of AD tools in your portfolio