



AVX v2 Whitepaper

What is AVX v2?

Avid eXtensions (AVX v2) is Avid's **next generation cross-platform software architecture** designed to allow **plug-in software components** to be seamlessly integrated with Avid **host applications** that support the AVX v2 application programming interface (API). AVX v2 is an enabling technology that allows plug-ins to dynamically extend the available features and behaviors supported by Avid host applications.

The AVX environment consists of host provided utilities and an application programming interface that handles the flow of image data, the passing of effect parameters and meta data, and the communication necessary to enable effects to be included in the host application's user interface or to invoke its own user interface.

The AVX v2 environment supports plug-ins implemented in C++. The AVX v2 plug-in API is a component-programming framework. The interface is independent from the underlying operating system and host application.

The AVX v2 architecture attempts to maintain the original **design goals** of the AVX v1 environment, including:

- Platform and host application independence
- Simplicity in plug-in implementation and scaled complexity
- Extensibility—support for new features and new types of plug-in components can easily be added at any time without affecting existing components
- Versioning and backward compatibility through component-based architecture and feature negotiation
- Built-in user interface controls provided by the host application
- Support for custom plug-in provided user interfaces
- Support for several types of image buffer organizations and colorspaces
- Image resolution independence
- Elements of the API are familiar to an AVX v1 plug-in developer to minimize developer learning curve
- Next generation effects framework for internal Avid effects as well as third party effects

New Features in AVX v2

- Negotiation of new image formats and color spaces—RGB, YCbCr601, YCbCr709; image sizes including HD; 16-bit processing
- Advances in keyframing and more flexible parameter layout & manipulation
- Layered component architecture—better versioning, no link dependencies, enabling future support for additional component types (e.g., video monitoring tools, character generation tools, MetaSync™ plug-ins)
- Stateless, thread-safe components to support multi-threading
- Realtime image effects
- Playmode enhancements

Layered Architecture

In order for Avid's new plug-in architecture to support a variety of plug-in types, in addition to image effects, we've designed it in layers. The base layer is agnostic of component type, and only includes the interfaces and infrastructure necessary to implement a generic software component that can be plugged into an Avid host. The interfaces required to implement image effects and their supporting components are layered over the base plug-in framework. The base plug-in framework is called the Avid Component Framework (ACF). Interfaces specific to image effects will be referred to as AVX v2. Avid has already internally implemented and shipped other classes of components layered over ACF. Additional component types will be publicly supported in the future.

AVX v2	Avid Subsystem
Avid Component Framework	



How does AVX v2 differ from AVX v1?

The fundamental object model for AVX v2 has changed from AVX v1 to a more component-oriented architecture. However, the standard image-processing concepts still hold. Unlike AVX v1, however, AVX v2 provides a flexible model to phase in new component types and new capabilities for existing component types over multiple releases of the host. Plug-ins can negotiate feature and component type support with the current host at run-time. This allows a single plug-in module to build a set of capabilities that adapt to the host that is loading it.

An AVX v1 effect was implemented with an `avxModel/avxInstance` pair of objects that contained state, such as parameter instance data and render context information. In AVX v2, image effects are now pure rendering engines. One image effect component is used to render all effects that appear in the timeline. State is maintained by the host and passed in to each method, freeing the effect to concentrate on the rendering algorithms. Stateless effect components set the stage for a host to render frames or frame tiles concurrently in separate threads.

Avid will support AVX v1 and v1.5 in its host applications for the foreseeable future. However, all further feature development will be focused on the next generation ACF/AVX v2 framework.

AVX Third Party Program

We encourage third parties to join the AVX program! We are currently evaluating AVX plug-in vendors to participate in the program. Preference will go to vendors who have a product that will stress the limits of AVX v2, and have experience building plug-ins. See the contact information below if you are interested.

AVX v2 Software Development Kit

- A Quickstart Guide to Writing AVX v2 Plug-Ins
- AVX v2 Wiz: developer tool to automatically generate a base plug-in framework
- AVX v2 Developer Guide
- AVX v2 Developer Reference pages
- Sample AVX v2 plug-ins
- Additional utility methods (not part of AVX v2 interface) to aid in plug-in development
- Avid Xpress® Pro build with expiring dongle
- Developer support

Contacts

For more information about the AVX program or to inquire about being a third party AVX developer register interest on avid.com: <http://www.avid.com/forms/avxpartnerinfo.asp>

Disclaimer

Please note that the information contained herein was compiled as of April 14, 2004. Avid has used reasonable efforts to ensure the accuracy and completeness of this information as of that date. However, this information may contain inadvertent errors or omissions. Avid specifically disclaims any obligation to update the information contained herein.

© 2004 Avid Technology, Inc. All rights reserved. Product features, specifications, system requirements and availability are subject to change without notice. Avid, Avid Xpress, MetaSync and AVX are either registered trademarks or trademarks of Avid Technology, Inc. in the United States and/or other countries. All other trademarks contained herein are the property of their respective owners.