

Adobe® Flash® Media Server 3.5

Die nächste Generation der preisgekrönten Adobe-Software für Video-Streaming und Kommunikation in Echtzeit

Inhalt

1	Einführung
2	Die wichtigsten Neuerungen in Adobe Flash Media Server 3.5
3	Die Adobe Flash Media Server 3.5-Familie
5	Die Flash Media Server-Editionen im Vergleich
6	Systemanforderungen für Flash Media Server
7	Upgrades für Flash Media Server
8	Unterstützung für Adobe Flash Player
9	Funktionsprinzip von Flash Media Server 3.5
11	Erweiterungen für Flash Media Server
11	Vorteile von Streaming im Vergleich zur HTTP-Bereitstellung
13	Streaming
16	Übersicht über die Funktionen von Flash Media Server 3.5
22	Sicherheit
24	HD-Video
27	Live-Video
28	Bereitstellung für mobile Endgeräte
29	Programmierungsfunktionen
31	Implementierung von Flash Media Server 3.5
31	Installationsprüfung
35	Performance-Funktionen
41	Sicherheitsfunktionen
43	Allgemeine Einstellungen
45	Server-seitige Werkzeuge
46	Skalierung von Flash Media Server 3.5
50	Funktionen für Live-Video
53	Datenschutz mit Flash Media Server 3.5
55	Verschlüsselung von Streaming-Inhalten
57	Glossar
58	Online-Ressourcen
59	Flash Media Server-Community
59	Quellen
59	Über die Autorin

Die Adobe Flash Media-Produktfamilie hat sich als Technologieplattform für Video-Streaming und Kommunikation in Echtzeit etabliert. Aufgrund ihrer weiten Verbreitung gewährleistet die Adobe Flash-Technologie konsistente Anzeigelerlebnisse auf allen Betriebssystemen. Mit Flash Media Server 3.5 profitieren Kunden nun auch von den Vorteilen des dynamischen Streamings, DVR- und erweiterten H.264-Funktionen sowie umfassender Unterstützung für die Bereitstellung über HTTP.

Dieses White-Paper beschreibt die wesentlichen Merkmale von Flash Media Server mit dem Schwerpunkt auf der herausragenden Performance und Funktionalität von Version 3.5. Ferner werden die Skalierbarkeit der Software und die Vorzüge von Streaming erörtert. Auf der Grundlage der Informationen in diesem Dokument können Sie fundierte Entscheidungen darüber treffen, wie Sie Video- und Kommunikationsdienste gewinnbringend einsetzen und die größtmögliche Reichweite für Ihre Online-Inhalte erzielen.

Einführung

Mit den Lösungen der Adobe Flash Media Server-Familie stellen Sie Rich-Media-Inhalte sicherer und effizienter als mit anderen Technologien bereit – und erreichen noch dazu ein größeres Publikum. Von anwendergenerierten Inhalten über Filme und Fernsehsendungen bis zu firmeninternen Schulungen können Unternehmen und Organisationen mit Flash Media Server Distributions- und Kommunikationslösungen für die unterschiedlichsten Zwecke entwickeln. Die leistungsstarke Plattform erleichtert die Erstellung innovativer Umgebungen mit fortschrittlicher Mehrkanalkommunikation und DVR-Funktionalität sowie die sichere Bereitstellung von Videoinhalten in HD-Qualität, integrierten Live-Streams oder Inhalten für mobile Endgeräte. Eine erweiterbare Architektur ermöglicht zudem umfassende Interaktivität. Zu den wichtigsten Vorteilen zählen:

Herausragende Anwendererlebnisse

- Schnellere Wiedergabe
- Unterstützung für Live-Video

Qualität

- Streaming und Aufzeichnung mit den Video-Codecs On2 VP6 und H.264/AAC (Advanced Audio Coding)
- Automatische Bandbreitenerkennung und dynamisches Streaming (neu)

Universelle Clients

- Plattformübergreifende Client-Unterstützung (Windows®, Mac und Linux®)
- Neuer integrierter HTTP-Server zur Bereitstellung aller Ressourcen einer Anwendung vom gleichen Server aus und für den bedarfsweisen Wechsel zum progressiven Download
- Weltweiter Einsatz von Adobe Flash Player – der Client wird täglich millionenfach heruntergeladen und ist auf 99 % aller PCs weltweit installiert
- Windows- oder Linux-Server-Distributionen, jetzt mit Unterstützung für mehr Betriebssysteme

Interaktivität

- Integriertes Video-, Audio- und Daten-Streaming für Mediennetzwerke
- Erweiterte DVR-basierte Wiedergabesteuerungen wie Kapitelnavigation, Überspringen, Interaktivität und andere benutzerdefinierte Funktionen
- Anpassbare Server-Funktionalität (mithilfe von Server-seitigem ActionScript® und Plug-in-Architektur)

Sicherheit

- Erhöhte Sicherheit bei der Bereitstellung durch Verschlüsselung und Zugriffskontrolle und Verzicht auf Client-Caching
- Unterstützung für verschlüsseltes H.264-Streaming (neu)
- Verbesserte SWF-Verifizierung autorisierter Anwendungen

Unterstützung bestehender Strukturen

- Bessere Nutzung der Bandbreite
- Signifikante Kosteneinsparungen durch optimierte Bereitstellungsprofile
- Große und aktive Entwickler-Community für Flash und Adobe Flex®
- Bewertung der Effizienz und Qualität des Streaming-Prozesses mittels benutzerdefinierter Protokolle
- Optionen zur präzisen Steuerung des Wiedergabeerlebnisses auf Anbieter- oder Distributorseite
- Unterstützung für Strategien zur Markenpositionierung

Diese und zahlreiche weitere Vorteile machen Flash Media Server 3.5 zur ersten Wahl für Medienanbieter, die Inhalte live oder auf Abruf („on demand“) bereitstellen möchten. Die Server-Lösung von Adobe ist stabil, effizient und kostengünstig. Mit den flexiblen Skalierungsoptionen lässt sie sich zudem mühelos an neue Anforderungen anpassen.

Die Tatsache, dass die Adobe Flash-Plattform bereits auf der großen Mehrheit der Desktop-Computer und anderen Endgeräte eingesetzt wird, ist ebenfalls ein wichtiger Vorteil. Mit Flash Media Server bereitgestellte Videoinhalte lassen sich in Webbrowsern mit Flash Player, auf Desktops mit Adobe AIR™ und auf mobilen Endgeräten mit Adobe Flash Lite™ 3 wiedergeben. Mit geringem Aufwand bereiten Sie Inhalte für ein breites Publikum auf und erschließen auf diesem Weg zahlreiche neue Absatzchancen.

Im Gegensatz zu anderen Technologien für die Videobereitstellung, die meist vorkonfigurierte Player im Corporate Design des Herstellers enthalten, lässt sich Flash Media Server 3.5 mit flexibel anpassbaren Adobe Flash Player- und Adobe AIR-Clients integrieren. Funktionen für den Echtzeit-Datenaustausch, Server-seitige Plug-ins, Protokolle und Überwachungs-APIs bieten Entwicklern und IT-Teams die erforderlichen Werkzeuge zur Entwicklung und Verwaltung abwechslungsreicher Rich-Media-Erlebnisse.

Die wichtigsten Neuerungen in Adobe Flash Media Server 3.5

Adobe Flash Media Server 3.5 ist ein skalierbarer Medien-Server für die plattformübergreifende Live- oder On-Demand-Bereitstellung hochwertiger Audio- und Videoinhalte (bis HD) mit höchster Effizienz und Qualität (Quality of Service, QoS). Er ermöglicht On-Demand- und Live-Videoübertragungen und -aufzeichnungen, Musik-Streaming, Video-gestütztes Blogging, Video-Messaging, Chat-Umgebungen mit Multimedia, Data-Casting in Echtzeit, Mehrspieler-Games u. v. m.

Flash Media Server stellt Daten und Streams für Flash Player, Adobe AIR, Flash Lite 3 und Adobe Media Player konsistent auf allen gängigen Browsern und Betriebssystemen bereit.

Zu den zahlreichen Neuerungen und Verbesserungen in Flash Media Server 3.5 zählen u. a.:

- Dynamisches Streaming – eine verbesserte QoS-Funktion, die von Flash Player unterstützt wird
- DVR-Funktionen zum Anhalten der Wiedergabe und Durchsuchen von Live-Videos
- Integrierter HTTP-Server, der als Host für interaktive und Streaming-Anwendungen eingesetzt werden kann und bei Bedarf auf progressive Bereitstellung umschaltet

Vorteile von Adobe Flash Media Server 3.5:

- Für Verbraucher: Hochqualitative Videoerlebnisse
- Für Medienanbieter: Universelle und sichere Plattform
- Für Werbetreibende: Mehrkanalkommunikation
- Für IT-Profis: Skalierbare Enterprise-Lösung für hochwertiges Streaming
- Für Entwickler: APIs zur Erstellung moderner Multiway-Medienanwendungen

- Bereitstellung verschlüsselter Inhalte, einschließlich Unterstützung von H.264 für Adobe AIR und Adobe Media Player
- Neue ActionScript-Klassen für dynamisches Streaming
- Verbesserte Code-Beispiele für Plug-ins in C++
- Erweiterte Plattformunterstützung, z. B. für Windows 2008 und Linux Red Hat® 5.2
- Senkung der Kosten für Streaming und Infrastruktur, da bei geringerer CPU-Auslastung pro Server mehr Streams für eine größere Anzahl von Anwendern übermittelt werden
- Effizienterer Zugriff auf Client-Statistiken über das Autorisierungs-Plug-in
- Unterstützung für Video-Streaming mit eingebetteten XMP-Metadaten (Adobe Extensible Metadata Platform)
- Verbesserte SWF-Verifizierung, einschließlich Remote-Validierung
- Unterstützung für Server-seitige Aufzeichnung im MPEG-4-Format
- Neues Simulationswerkzeug zur Überprüfung der Netzwerkauslastung (separat erhältlich)
- Vollständige Kompatibilität mit vorhandenen Flash Media Server 2- und 3-Anwendungen
- Einfachere Implementierung dank erweiterter Dienste, verbesserter Dokumentation und Vorkonfiguration

Diese Verbesserungen belegen die Weiterentwicklung von Flash Media Server mit dem Ziel, Entwicklern die Werkzeuge an die Hand zu geben, die sie für die Erstellung leistungsfähiger Rich-Media-Anwendungen benötigen. Dank der erweiterten Support-Dokumentation sowie einfacherer Veröffentlichungsoptionen und einer Vielzahl an Beispielanwendungen lassen sich mit Flash Media Server 3.5 rasch professionelle Ergebnisse erzielen.

Die Adobe Flash Media Server 3.5-Familie

Flash Media Server ist in drei Editionen erhältlich:

- Adobe Flash Media Development Server 3.5 (kostenlos)
- Adobe Flash Media Streaming Server 3.5
- Adobe Flash Media Interactive Server 3.5

Flash Media Interactive Server 3.5 kann zur effizienten Lastenverteilung als Origin- oder Edge-Server implementiert werden. Näheres zur Origin-Edge-Konfiguration erfahren Sie im Abschnitt „Skalierung von Flash Media Server 3.5“.

Im Folgenden werden die Funktionen der einzelnen Editionen erläutert, um Ihnen die Auswahl der für Sie besten geeigneten Lösung zu erleichtern.

Adobe Flash Media Development Server 3.5

Diese Edition von Adobe Flash Media Server kann kostenlos von der Adobe-Website heruntergeladen werden und eignet sich für einfache Streaming- oder Kommunikationslösungen mit geringem Datendurchsatz. Sie kann aber aufgrund ihrer beliebig erweiterbaren Funktionalität auch für komplexere Anwendungen und soziale Netzwerke genutzt werden. Auch Multipoint-Publishing wird unterstützt. Dazu erstellen Sie einen Live-Publishing-Point im Netzwerk, binden Datennachrichten in den Live-Stream ein, und leiten das Video dann an ein größeres CDN weiter. Der kostenlose Medien-Server kann allerdings nur von 10 Anwendern gleichzeitig genutzt werden. Die Anzahl der zulässigen ausgehenden Verbindungen hängt allein von der verfügbaren Bandbreite des Servers und der Rechenleistung ab.

Adobe Flash Media Streaming Server 3.5

Diese Server-Edition steht für sicheres, unidirektionales Video-Streaming.

Flash Media Streaming Server 3.5 ist eine kosteneffiziente Lösung für die rasche und sichere Übertragung hochqualitativer Videos. Die Plattform enthält alle erforderlichen Funktionen zum Streamen von Video und Audio und kann gemeinsam mit Adobe Flash Media Live Encoder auch für Live-Video eingesetzt werden. Flash Media Streaming Server bietet vor allem folgende Vorteile:

- Geringer Kostenaufwand für die Bereitstellung von Streaming-Inhalten für Flash Player, Adobe AIR, Adobe Media Player und Flash Lite
- Einfache Installation und Anwendung
- Unterstützung mehrerer Bit-Raten und automatische Bandbreitenerkennung beim dynamischen Streaming
- HD-Qualität (mit Funktionen für den Videostandard H.264)
- Erweiterte Datenerfassung und -auswertung
- Höhere Datenkapazität
- Optimale Netzwerkauslastung
- Verbesserte Suchfunktionen
- Verschlüsseltes Streaming
- Einfache Zugriffskontrolle
- Hochwertiges Live-Video

Flash Media Streaming Server 3.5 wird mit zwei Diensten ausgeliefert, mit deren Hilfe Sie direkt nach Installation der Software Inhalte bereitstellen können:

- **Live-Video** – Die standardmäßige Streaming-Anwendung für Live-Video ist sofort einsatzbereit. Sie unterstützt Adobe Flash Media Live Encoder sowie die Komponente FLVPlayback (ab Flash 8) und wird mit einer Beispiel-SWF für Live-Stream-Abonnements ausgeliefert.
- **Video-on-Demand (VOD)** – Die VOD-Anwendung enthält alle Standardfunktionen wie Erkennung der Server- oder Client-Bandbreite, domänenbasierte Authentifizierung, vollständige Unterstützung der FLVPlayback-Komponente (ab Flash 8) sowie FLV-, SWF- und HTML-Beispieldateien zur Wiedergabe.

Der Server-seitige Code für die Flash Media-Dienste kann nicht geändert werden. Die bereitgestellten Beispieldateien können Sie für Testzwecke nutzen, oder Sie laden eigene hoch und führen mehreren Instanzen aus.

Beide Dienste unterstützen die Funktion von Flash Media Server 3 (und höher) für den Zugriff auf die Bitmap-Daten eines Streams, die auch mit Flash Media Interactive Server kompatibel ist. Flash Media Interactive Server bietet zusätzlich die Möglichkeit, die Dienste durch eigene Funktionen zu erweitern, Streams aufzuzeichnen, gemeinsame Objekte per Remote-Zugriff zu nutzen und weitere Skalierungsfunktionen zu wählen. In der Dokumentation zu Adobe Flash Media-Diensten finden Sie weitere Informationen.

Adobe Flash Media Interactive Server 3.5

Adobe Flash Media Interactive Server 3.5 bietet erweiterte Streaming-Funktionen wie

- DVR-Funktionalität
- Edge-Server-Caching
- APIs zur Zugriffskontrolle
- Server-Umleitung (ähnlich einer HTTP 302-Umleitung)
- Plug-ins
- Benutzerdefinierbare Videodienste
- Server-seitige Aufzeichnung – jetzt auch im H.264-Format
- Mehrkanalkommunikation

- Lösungen für Mediennetzwerke
- Verteilte Prozesse
- Multipoint-Publishing
- Einbetten von Metadaten in den Live-Stream

Flash Media Interactive Server lässt sich mit mehrwertigen Multiway-Lösungen integrieren, um Streaming-Inhalte mit erweiterten Funktionen für Kommunikation und Zusammenarbeit in Echtzeit zu kombinieren. Flash Media Interactive Server ist die einzige hochleistungsfähige und skalierbare Server-Lösung auf dem Markt mit Unterstützung für Multiway-Anwendungen, z. B. für Webcam-Chats, Aufzeichnungen, VoIP-Konferenzen oder Online-Spiele. Flash Media Interactive Server ist das vielseitigste Produkt der Flash Media Server-Familie.

Diese Edition eignet sich für allem für

- Streaming in großem Umfang
- Entwicklung eigener Lösungen für die Videobereitstellung
- Erstellung interaktiver Kommunikationsanwendungen
- Einbinden von Interaktivität in Streaming-Dienste für Live- oder On-Demand-Video

Flash Media Interactive Server ist mit Adobe Flash Media Live Encoder integriert und interagiert mit zweckgerichteten Daten-Servern wie LDAP (Lightweight Directory Access Protocol) zur Authentifizierung sowie Flash Remoting, SOAP (Simple Object Access Protocol) und XML. Zur weiteren Anpassung können Sie eigene Server-seitige ActionScript-Routinen implementieren und Plug-ins in C++ entwickeln.

Die Flash Media Server-Editionen im Vergleich

Funktionen	Flash Media Interactive Server 3.5	Flash Media Streaming Server 3.5	Flash Media Development Server 3.5
<ul style="list-style-type: none"> • Dynamisches Streaming • XMP-Metadaten • HD-Video (H.264/AAC) • Streaming mit Echtzeit-Verschlüsselung • Protokollierung • SWF-Verifizierung • Live-Streaming • Streaming von Aufzeichnungen • Effiziente Nutzung der Netzwerkressourcen • Maximale Hardware-Kapazität • Erweiterter Cache • Unterstützung von 2 GB-Dateien • Integrierte Bandbreitenerkennung • IPv6-Unterstützung • Adobe Media Player-Unterstützung • Unterstützung von Flash Lite 3 für mobile Endgeräte • Daten-Keyframes • Zugriff auf Stream-Daten 	Unterstützt alle Funktionen	Unterstützt alle Funktionen	Unterstützt alle Funktionen
Gleichzeitige Verbindungen	Unbegrenzt	Unbegrenzt	10

Funktionen	Flash Media Interactive Server 3.5	Flash Media Streaming Server 3.5	Flash Media Development Server 3.5
Bandbreitenunterstützung	Unbegrenzt	Unbegrenzt	Unbegrenzt
Maximale Anzahl Prozessoren	8-faches SMP (Mehrkern)	4-faches SMP (Mehrkern)	8-faches SMP (Mehrkern)
Flash Media Server-Dienste (Live-Video und VOD)	•	• (Nicht anpassbar)	•
Definition des Prozessumfangs und Verteilung von Core-Prozessen	•		
Core-Prozesse	Unbegrenzt	1	Unbegrenzt
Server-Archivierung (Aufzeichnung) von Video	•		•
Eigene Server-seitige Anwendungen (Server-seitiges ActionScript)	•		•
Edge-Server-Konfiguration	•		•
Unterstützung von Plug-ins in C++	•		•
Server-seitige Wiedergabelisten (Internet-TV)	•		•
Multipoint-Publishing/Umleitung	•		•
Remote Shared Objects	•		•
AMF3-Unterstützung	•		•
Server-Umleitung	•		•
Plug-in-Architektur für LDAP-Authentifizierung	•		•
Datenverfolgung mit Adobe Media Player	•		

Systemanforderungen für Flash Media Server

Windows	Linux
Microsoft® Windows Server® 2003 mit Service Pack 2 Microsoft Windows Server 2008 Microsoft Windows XP (nur Flash Media Development Server)	Linux Red Hat 4 Linux Red Hat 5.2
Hardware	
<ul style="list-style-type: none"> • Intel® Pentium® 4 mit 3,2 GHz (Intel Xeon®-Dualprozessor oder schneller empfohlen) • 2 GB RAM (4 GB empfohlen) • Gigabit-Ethernet-Karte 	

Die aktuellen Systemanforderungen finden Sie unter www.adobe.com/go/learn_fms_sysreqs_de.

Upgrades für Flash Media Server

Upgrade von Adobe Flash Media Server 3

Ob Sie ein Upgrade von Flash Media Server 3 vornehmen können, hängt von der Art und vom Kaufdatum Ihrer aktuellen Lizenz ab:

- Wenn Sie Flash Media Interactive Server 3 besitzen, sind Sie zum Upgrade auf Flash Media Interactive Server 3.5 berechtigt.
- Wenn Sie Flash Media Streaming Server 3 besitzen, können Sie auf Flash Media Streaming Server 3.5 oder Flash Media Interactive Server 3.5 aufsteigen.
- Education-Lizenzen sind nicht zum Upgrade berechtigt.

Upgrade von Macromedia® Flash Media Server 2

Ob Sie ein Upgrade von Flash Media Server 2 vornehmen können, hängt von der Art und vom Kaufdatum Ihrer aktuellen Lizenz ab. Für das Upgrade benötigen Sie die Seriennummer, die Sie für Flash Media Server 2 erhalten haben.

- Wenn Sie Flash Media Interactive Server 2 Professional, Origin oder Edge besitzen, sind Sie zum Upgrade auf Flash Media Interactive Server 3.5 berechtigt.
- Education-Lizenzen sind nicht zum Upgrade berechtigt.
- Wenn Sie Flash Communication Server 1.x (Personal und Professional) besitzen, können Sie nicht auf Flash Media Server 3.5 aufsteigen.
- Wenn Sie Flash Media Server 2 besitzen, sind Sie zum Upgrade auf Flash Media Interactive Server 3.5, jedoch nicht auf Flash Media Streaming Server 3.5 berechtigt.

Upgrade von Flash Media Streaming Server

Für ein Upgrade von Flash Media Streaming Server 3 stehen Ihnen zwei Optionen zur Auswahl:

- Upgrade auf Flash Media Streaming Server 3.5
- Upgrade auf Flash Media Interactive Server 3.5 mit erweiterter Funktionalität

Für das Upgrade von Flash Media Streaming Server 3.5 auf Flash Media Interactive Server 3.5 müssen Sie nur eine Upgrade-Seriennummer erwerben, da die zugrunde liegende Server-Software sich nicht ändert.

Hinweis: Mit dem Upgrade für Flash Media Interactive Server wird die zugehörige Dokumentation nicht installiert. Diese müssen Sie im Rahmen Ihrer Upgrade-Bestellung separat erwerben.

Im vorherigen Abschnitt sehen Sie alle Funktionen der Editionen von Flash Media Server auf einen Blick.

Upgrade Plan für Flash Media Server 3.5

Inzwischen gibt es einen Upgrade Plan für Flash Media Server 3.5, der Ihre Investition für maximal zwei Jahre schützt. Der Plan stellt sicher, dass Sie stets über die aktuelle Version von Flash Media Server verfügen. Preis- und sonstige Informationen erhalten Sie von Ihrem zuständigen Adobe-Vertriebsmitarbeiter.

Unterstützung für Adobe Flash Player

Funktionen	Version 6	Version 7	Version 8	Version 9	Version 9,0,115,0	Version 10
Video-Codec von Sorenson (H.263): Wiedergabe und Aufnahme	•	•	•	•	•	•
Video-Codec On2 VP6: nur Wiedergabe			•	•	•	•
H.264 und HE/AAC: nur Wiedergabe					•	•
RTMPE/RTMPTE					•	•
SWF-Verifizierung					•	•
Hardware-Beschleunigung für Grafikwiedergabe					•	•
Unterstützung von Mehrkern- prozessoren					•	•
Intelligentes Zwischenspeichern					•	•
Dynamisches Streaming						•
XMP-Metadaten						•
RTMFP						•
Audio-Codec Speex						•

Adobe AIR (ab Version 1.0)

Adobe AIR ist eine plattformübergreifende Laufzeitumgebung, die es Entwicklern ermöglicht, Kenntnisse in gängigen Web-Technologien wie HTML, Ajax, Flash und Flex auch beim Erstellen von Rich-Internet-Anwendungen (RIAs) für den Desktop zu nutzen.

Adobe AIR-Anwendungen lassen sich nativ mit dem Desktop integrieren und unterstützen Zwischenablage- und Drag-and-Drop-Vorgänge, lokale Dateiverwaltung, Systembenachrichtigungen u. v. m. AIR-Anwendungen können sich mit Flash Media Server verbinden, um wie SWF-Dateien Audio und Video zu streamen oder Daten bereitzustellen.

Adobe Media Player (ab Version 1.0)

Adobe Media Player ist eine kostenlose, eigenständige Anwendung auf Basis der Adobe AIR-Plattform. Der Player wird für die Bereitstellung von Videoinhalten eingesetzt und kann durch eigene Branding-Elemente, Optionen zur Nutzungsüberwachung und DRM-Funktionen ergänzt werden. Der Mechanismus, mit dem Adobe Media Player Inhalte streamt sowie Client-Ereignisse und Anzeigehistorien erfasst, stammt aus Flash Media Interactive Server.

Adobe Flash Lite (ab Version 3.0)

Mit Flash Media Server 3.5 lassen sich Inhalte auch an mobile Endgeräte übermitteln, die Flash Lite 3 installiert haben. Daraus ergeben sich neue Absatzmöglichkeiten für interaktive Streaming-Inhalte. Flash Lite bietet folgende leistungsstarke Funktionen:

- Echtes FLV- und MP3-Streaming (mit den Codecs On2 VP6-E, Sorenson Spark und MP3)
- ActionScript 2.0-API (NetConnection und NetStream)
- Ermittlung der Gerätekennung zur modellspezifischen Optimierung von Inhalten
- Unterstützung für RTMP-Tunneling (Real-Time Messaging Protocol)

Flash Media Server 3.5 unterstützt sowohl die Übertragung aufgezeichneter Inhalte als auch Live-Streaming. Auf mobilen Endgeräten mit Flash Lite 3 lässt sich somit dasselbe hochwertige Videoerlebnis bereitstellen wie im Browser.

Funktionsprinzip von Flash Media Server 3.5

Die Lösungen der Flash Media Server-Familie haben sowohl eine Server-seitige als auch eine Client-seitige Architektur. Auf dem Client wird eine SWF- oder AIR-Datei ausgeführt, die mit Flash oder Flex erstellt wurde. Das Client-Erlebnis findet in einem Webbrowser (Flash Player), auf mobilen Endgeräten (Flash Lite 3) oder auf dem Desktop (Adobe AIR) statt. Als Client kann auch ein weiterer Flash Media-Server oder Adobe ColdFusion® 8, Adobe Flash Media Live Encoder sowie die lizenzierte Technologie eines Drittanbieters dienen, die mit Flash Media Server streamen oder kommunizieren kann. Der Server verwaltet Client-Verbindungen und Sicherheitsfunktionen, führt Lese- und Schreibvorgänge im Dateisystem des Servers durch u. v. m.

Der Client initiiert die Verbindung zum Server. Sobald eine Verbindung besteht, kann der Client mit dem Server und anderen verbundenen Clients kommunizieren. Clients verbinden sich mit Instanzen von Anwendungen. Wenn Sie beispielsweise eine Chat-Anwendung in mehrere Räume aufgliedern, können Sie jeden Raum als Instanz der Chat-Anwendung definieren. Es können mehrere Instanzen gleichzeitig ausgeführt werden. Jede Instanz erhält eine eindeutige Bezeichnung und stellt den verbundenen Clients jeweils eigene Ressourcen zur Verfügung.

Kommunikationsprotokoll von Flash Media Server (RTMP)

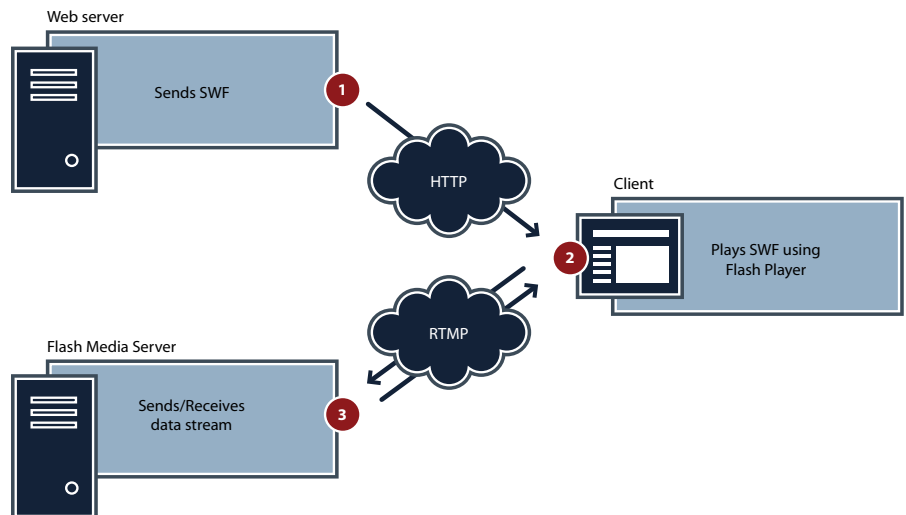
Flash Media Server kommuniziert mit Clients mit dem von Adobe patentierten Protokoll RTMP über TCP (Transmission Control Protocol). Über eine bidirektionale Verbindung werden dabei Daten, Video- und Audioinhalte zwischen Server und Clients ausgetauscht. Flash Media Server 3.5 bietet zudem die Möglichkeit, die Streaming-Sicherheit durch den Einsatz von verschlüsseltem RTMP (RTMPE) zu erhöhen. Die Verschlüsselung von Streams ist mit RTMPE einfacher und schneller als mit SSL (Secure Socket Layer). RTMPE ist nur eine der vielen neuen Sicherheitsfunktionen von Flash Media Server 3.5.

RTMP wird von Flash Media Server 3.5 in fünf Konfigurationen unterstützt:

- **RTMP** – Standardmäßiges, nicht verschlüsseltes RTMP. Der Standard-Port ist 1935. Ist kein Port vorgegeben, versucht der Client, Verbindungen zu den folgenden Ports in der angegebenen Reihenfolge aufzubauen: 1935, 443 und dann 80 über RTMPT. Port 1935 ist ein registrierter IANA-Port.
- **RTMPT** – Über HTTP „getunneltes“ RTMP. Die RTMP-Daten werden als gültige HTTP-Daten erfasst. Der Standard-Port ist 80.
- **RTMPS** – Über SSL übermitteltes RTMP. SSL ermöglicht sichere TCP/IP-Verbindungen. Flash Media Server bietet native Unterstützung für eingehende und ausgehende SSL-Verbindungen. Der Standard-Port ist 443.
- **RTMPE** – Erweiterte, verschlüsselte Version von RTMP. RTMPE ist schneller als SSL und erfordert im Gegensatz zu SSL keine Zertifikatverwaltung. Falls Sie für RTMPE keinen Port angeben, scannt der Flash Player genau wie beim Standard-RTMP die Ports in folgender Reihenfolge: 1935, 443, 80 und 80 (RTMPTE) (unterstützt von Flash Player 9,0,115,0 und höher, Adobe AIR und Adobe Media Player).
- **RTMPTE** – Verschlüsselt den Kommunikationskanal durch das Tunneln über HTTP. Der Standard-Port ist 80 (unterstützt von Flash Player 9,0,115,0 und höher, Adobe AIR und Adobe Media Player). Die wesentlichen Vorteile gegenüber SSL (RTMPS) sind bessere Performance, einfache Implementierung und geringe Auswirkungen auf die Server-Auslastung.

Mit dem richtigen RTMP-Typ kann Flash Media Server Streams durch fast alle Firewalls hindurch übertragen und den Schutz vertraulicher oder mit Nutzungsrechten versehener Inhalte erhalten.

Die folgende Abbildung veranschaulicht die grundlegende Struktur einer Flash Media Server-Verbindung.



Client-Server-Architektur von Flash Media Server

Unterstützte Dateiformate

Flash Media Server 3.5 ist vollständig abwärtskompatibel zu Flash Player 6 und höher, Adobe AIR und Flash Lite 3. Mit den neuen Versionen des Flash Player werden weitere Formate und Funktionen unterstützt.

Flash Media Server 3.5 unterstützt weiterhin FLV- und MP3-Inhalte sowie AMF0 für Data-Messaging. In Kombination mit Flash Player 9,0,115,0 und höher wird das Standardformat für digitales Video MPEG-4 unterstützt.

Die von Flash Media Server unterstützten Dateiformate werden in der folgenden Tabelle aufgeführt. Adobe AIR unterstützt alle Formate.

Dateiformat	Typ	Container	Flash Player-Version	Typisches Paar
Sorenson Spark	Video	FLV	ab Version 6	Nellymoser/MP3
On2 VP6	Video	FLV	ab Version 8; Flash Lite 3	Nellymoser/MP4
H.264*	Video	MPEG-4: MP4, M4V, F4V**, 3GPP	ab Version 9,0,115.0	AAC+/MP3
Nellymoser	Audio	FLV	ab Version 6	Sorenson Spark/ On2 VP6
MP3	Audio	MP3	ab Version 6; Flash Lite 3	Sorenson Spark/ On2 VP6
AAC+, HE AAC, AAC v1 oder AAC v2	Audio	MPEG-4: MP4, M4A, F4V, 3GPP	ab Version 9,0,115.0	H.264
Speex	Audio	FLV	ab Version 10	Sorenson Spark/ Speex
AMF0	Daten		ab Version 6; Flash Lite 3	
AMF3	Daten		ab Version 8	

* Die H.264-Wiedergabe in Flash Player unterstützt die gängigsten Profile wie Base, Main und High Profile (HiP).

** Das F4V-Format basiert auf den Spezifikationen von MPEG-4 ISO 14496-10 und AAC+ (ISO 14496-3).

Hinweis: Um H.264/AAC ohne ActionScript in Flash zu verwenden, benötigen Sie die aktualisierte FLVPlayback-Komponente, die Sie als Update für Flash CS3 Professional erhalten. Auch für den Einsatz des erweiterten RTMP (RTMPE) ist dieses Update erforderlich. Ohne die FLVPlayback-Komponente können Entwickler nur mit ActionScript 1, 2 oder 3 Anwendererlebnisse mit H.264 erstellen. Für die neue Funktion für dynamisches Streaming benötigen Sie die aktualisierte FLVPlayback-Komponente, die in Flash CS4 Professional enthalten ist.

Weitere Informationen zur Unterstützung von H.264 und AAC finden Sie in den häufig gestellten Fragen zum Flash Player 9-Update unter http://labs.adobe.com/wiki/index.php/Flash_Player:9:Update:H.264 (Englisch).

Erweiterungen für Flash Media Server

Für die Entwicklung eigener Anwendungen und zur Überwachung von Flash Media Server-Aktivitäten stehen mehrere APIs zur Verfügung.

API für Client-seitiges ActionScript

Sie können jede Version von Adobe Flash (ab MX 2004) oder Adobe Flex Builder™ für Client-seitige Skripten verwenden, die mit Flash Media Server kommunizieren (z. B. für das Streamen oder Aufnehmen von Live-Audio und -Video bzw. das Aufrufen von Server-seitigen Funktionen). Die Skripten können in jeder Version von ActionScript verfasst werden.

API für Server-seitiges ActionScript

Flash Media Interactive Server ermöglicht den Zugriff auf Server-seitigen ActionScript-Code. Der Code wird zur Überwachung von Anmeldegerichtlinien, zur erneuten Veröffentlichung von Inhalten auf anderen Servern, zur Freigabe oder Sperrung von Anwenderzugriffen auf Server-Ressourcen oder zur Genehmigung der Aktualisierung und Freigabe von Informationen eingesetzt. Server-seitiges ActionScript ähnelt ActionScript 1.0.

API für Plug-ins

Flash Media Interactive Server bietet in C++ entwickelte Plug-ins, mit denen Sie die Funktionalität des Servers erweitern können. Einige Plug-ins kontrollieren Zugriffe, ermöglichen die zielgruppenspezifische Anpassung von Inhalten, erfassen statistische Daten zu Client-Aktivitäten und führen netzwerkbasierete Dateivorgänge durch. Im Lieferumfang von Flash Media Server 3.5 sind neue, einfach zu bedienende Beispiel-Plug-ins enthalten.

Verwaltungs-API

Die Verwaltungs-API stellt Ihnen die erforderlichen Werkzeuge zur Erstellung von Flash Player- oder Adobe AIR-Clients zur Verfügung, die die Aktivitäten in allen Editionen von Flash Media Server überwachen und verwalten können.

Vorteile von Streaming im Vergleich zur HTTP-Bereitstellung

Für die Bereitstellung von Video über das Internet mit Adobe Flash Player gibt es drei Optionen:

- Eingebettetes Video
- Progressiver Download
- Streaming

Da eingebettetes Video selten und nur in ganz speziellen Anwendungen für kurze Video-Clips in geringer Qualität genutzt wird, werden im Folgenden nur der progressive Download und die Streaming-Methode näher erläutert.

Sowohl beim progressiven Download als auch bei der Bereitstellung per Streaming werden Videoinhalte außerhalb der SWF-Datei verwaltet. Zur Veröffentlichung der Inhalte im Web müssen die SWF- und die Videodatei auf einen Server hochgeladen werden.

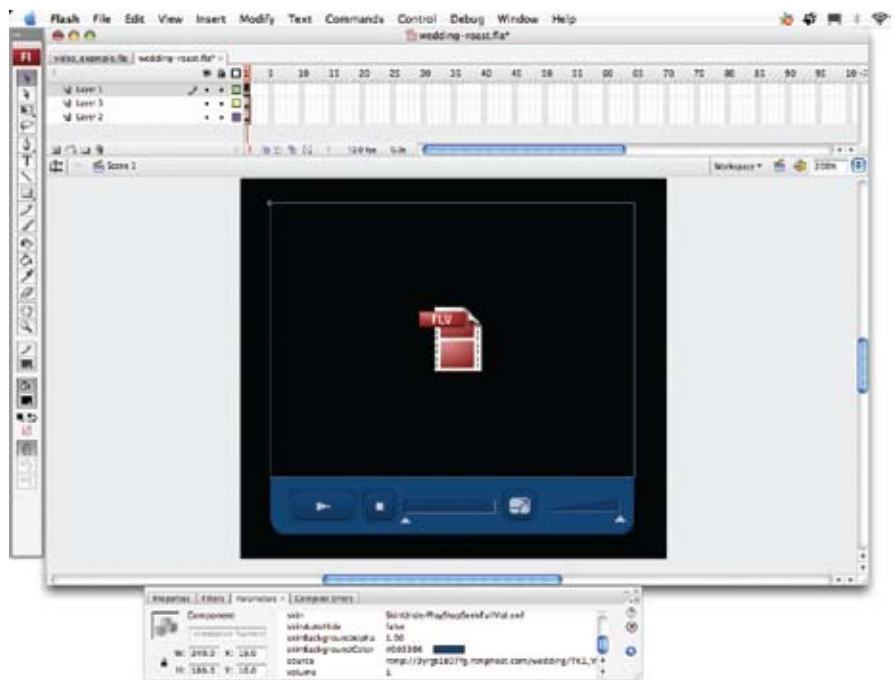
Die separate Verwaltung der Videodateien hat gegenüber eingebettetem Video u. a. folgende Vorteile:

- Einfache Aktualisierung – Die in der Datei enthaltenen Inhalte sind dynamisch. Außerdem lassen sich Inhalte unabhängig vom Video-Player hinzufügen oder ändern, ohne dass die SWF-Datei erneut veröffentlicht werden muss.
- Kompakte Größe der SWF-Datei – Die SWF-Datei ist klein und lässt sich schneller laden, d. h., wenn ein Anwender ein Video anfordert, kann es sofort abgespielt werden.
- Bessere Performance – Durch die separate Verwaltung von FLV- und SWF-Dateien profitiert der Endanwender von einem optimierten Wiedergabeergebnis.

Hinweis: Die in diesem Abschnitt behandelten Verfahren zur Bereitstellung von Videodateien gelten genauso für Audiodateien.

Bereitstellung per progressiven Download

Seit Flash MX 2004 wird der progressive Download als Bereitstellungsmethode für Videos unterstützt. Dabei werden externe Videodateien in eine Flash- oder Flex-Benutzeroberfläche geladen und zur Laufzeit wiedergegeben. Entwickler können zu diesem Zweck entweder ActionScript-Befehle für das Videoobjekt oder die Wiedergabekomponenten erstellen oder in der Authoring-Umgebung Parameter für die Wiedergabekomponenten konfigurieren (siehe Abbildung).



Die Parameter einer FLVPlayback-Komponente werden für die Übertragung (Streaming) einer externen Videodatei an eine SWF-Datei eingerichtet.

Wenn ein Anwender die Wiedergabe startet, wird zunächst die Videodatei auf seine Festplatte heruntergeladen. Danach wird das Video abgespielt. Die Wiedergabe beginnt, sobald ausreichend viele Daten auf die Festplatte geladen wurden. Die Datei wird genauso wie eine normale Web-Seite oder ein sonstiges herunterladbares Dokument mittels einer HTTP-Anforderung von einem Standard-Webserver bereitgestellt.

Im Vergleich zum Streaming hat der progressive Download nur einen klaren Vorteil: Sie brauchen keinen Streaming-Server. Videodaten können von jedem beliebigen Webserver übertragen werden. Obwohl dieses Verfahren bequem und meist auch kostengünstig ist, sollten Sie bedenken, dass das Suchen und Navigieren innerhalb progressiv heruntergeladener Inhalte nur eingeschränkt möglich ist. Zudem können andere auf Ihre Inhalte zugreifen und sie für eigene Zwecke nutzen. In einigen Situationen, wie etwa bei der Wiedergabe innerhalb einer restriktiven Netzwerkumgebung, werden Streaming-Inhalte nicht zugelassen. In diesen Fällen ist es sinnvoll, die progressive Methode als Alternative bereitzuhalten.

Wann ist die Bereitstellung per progressiven Download sinnvoll?

Der progressive Download eignet sich vor allem für private Websites oder Websites mit geringem Datenverkehr und auch nur dann, wenn der Anbieter nichts dagegen hat, dass seine Inhalte auf dem Computer des Endanwenders zwischengespeichert werden, wenn kein Video länger als 10 Minuten dauert und wenn Endanwender aus bestimmten Gründen keine Video-Streams empfangen können. Falls Sie ein erweitertes Funktions-Set und mehr Kontrolle über die Bereitstellung benötigen, falls Sie Inhalte für ein größeres Publikum (z. B. Hunderte von gleichzeitigen Verbindungen) bereitstellen, falls Sie Nutzungs- und Anzeigestatistiken erfassen und auswerten müssen oder falls Sie großen Wert auf ein hochwertiges, interaktives Wiedergabeerlebnis legen, dann ist Video-Streaming die bessere Wahl. Beim Streaming wird zudem eine geringere Bandbreite als bei der progressiven Bereitstellung verwendet, da nur die gerade wiedergegebene Videosequenz tatsächlich bereitgestellt wird.

Streaming

Video- und Audio-Streaming war erstmals mit Flash MX, Flash Player 6 und Flash Communication Server MX möglich.

Beim Streaming werden die Videodateien wie beim progressiven Download getrennt von den übrigen Inhalten abgelegt. Entwickler können mithilfe von ActionScript-Befehlen (und entsprechenden Parametern in den Medienkomponenten) die externen Videodateien in eine SWF-Datei laden und zur Laufzeit wiedergeben. Der ActionScript-Code für Streaming-Prozesse und progressive Downloads ist nahezu identisch.

Anders als beim progressiven Download stellt beim Video-Streaming jeder Client eine dauerhafte Verbindung zum Streaming-Server her. Der Server überträgt die Videodaten an den Client, wo sie vom Player abgespielt und umgehend wieder verworfen werden.

Aufgrund der engen Verbindung zwischen Server und Client sowie der Fähigkeit des Servers, jeden Teil eines Streams anforderungsgemäß zu überwachen und bereitzustellen, profitieren Entwickler u. a. von folgenden erweiterten Funktionen:

- Erkennung der Client-Bandbreite und Bereitstellung eines Streams mit passender Bit-Rate
- Messung und Verfolgung der Stream-Qualität und bei Bedarf Wechsel zu einer geringeren (oder höheren) Bit-Rate (z. B. wenn die Netzwerkauslastung steigt)
- Automatische Generierung von Miniaturansichten oder Wiedergabe einer kurzen Videovorschau, ohne dass separate Bilder oder Video-Clips erstellt und das gesamte Video im Hintergrund heruntergeladen werden muss
- Automatische Erstellung von Kapiteln (mit entsprechenden Miniaturansichten), die für die Navigation innerhalb längerer Video-Clips verwendet werden können, ohne dass das Video in kleinere Dateien aufgeteilt werden muss
- Reibungsloser Wechsel zwischen verschiedenen Kameraeinstellungen oder Stream-Versionen während des Streaming-Vorgangs
- Zusammenführen einzelner Videosequenzen zu einem vollständigen Video, z. B. die ersten 10 Sekunden von Clip 1, den Inhalt zwischen der 30. und der 40. Sekunde von Clip 2 plus die letzten 20 Sekunden von Clip 3
- Erstellung von Live-Webcasts oder Aufzeichnungen, bei denen alle Zuschauer gleichzeitig auf denselben Inhalt zugreifen

Vorteile von Streaming

Der progressive Download bietet nur wenige Kontrollmöglichkeiten bei der Bereitstellung von Video, denn im Grunde handelt es sich dabei um einen einfachen HTTP-Download-Aufruf. Beim Streaming dagegen können Anbieter auf jeden Aspekt des Videoerlebnisses Einfluss nehmen.

Video-Streaming mit Flash Media Server bietet zahlreiche Vorteile:

- **Schnelle Wiedergabe** – Streaming-Video wird schneller als andere Videos im Web wiedergegeben.
- **Erweiterte Steuerung** – Dazu zählen Funktionen wie die Erkennung der Bandbreite, QoS-Kontrolle, automatische Erstellung von Miniaturansichten, Server-seitige Wiedergabelisten usw.
- **Effiziente Nutzung der Netzwerkressourcen** – Kunden, die Host-Dienste oder Bandbreite auf Basis der übertragenen Daten-Bits bezahlen, können ihre Kosten mit dem Streaming-Verfahren senken, da nur die tatsächlich vom Client angezeigten Bits übertragen werden.
- **Besserer Schutz für bereitgestellte Inhalte** – Da Mediendaten beim Streaming nicht im Client-Cache gespeichert werden, können Endanwender die Video- oder Audiodatei auch nicht aus dem Ordner für temporäre Internet-Dateien abrufen. Weitere Sicherheitsfunktionen von Flash Media Server 3.5 verhindern außerdem die Aufzeichnung von Streams während der Wiedergabe („Streamripping“) und andere Risiken für die Dateisicherheit. Weitere Informationen finden Sie im Abschnitt „Datenschutz mit Flash Media Server 3.5“.
- **Minimale Auslastung der Client-Ressourcen** – Die Anforderungen an Arbeits- und Festplattenspeicher sind beim Streaming weitaus geringer, weil Clients nicht die gesamte Datei herunterladen oder speichern müssen.

- **Überwachung, Auswertung und Protokollierung** – Da es sich beim progressiven Download um das einfache Herunterladen einer Datei handelt, lassen sich bestimmte Statistiken nur schwer erfassen, z. B. wie lange das Video angezeigt wurde, ob der Endanwender vorgespult, zurückgespult oder unterbrochen hat, wie oft das Video abgespielt wurde, ob der Zuschauer die Web-Seite vor dem Ende des Videos verlassen hat usw. Beim Streaming lassen sich diese wichtigen Daten mühelos erfassen.
- **Vollständig durchsuchbare Videos** – Anwender können eine beliebige Stelle im Video suchen und von da aus die Wiedergabe starten. Streaming ist somit die ideale Lösung für Videos mit längerer Spieldauer oder Anwendungen für videogestütztes Blogging, Schulungen und Konferenzen, bei denen es nicht unbedingt erforderlich ist, dass Endanwender das gesamte Video anschauen.
- **Umfassende Interaktivität** – Dank der präzisen Kontrollmöglichkeiten beim Streaming können Entwickler Videos interaktiv gestalten, z. B. mit Optionen zur Änderung der Kameraperspektive, dem nahtlosen Übergang von einem Video zum anderen oder mit einem alternativen Ausgang der Geschichte.
- **Live-Video** – Live-Video und -Audio kann von jeder angeschlossenen Webcam oder DV-Kamera und in einigen Fällen sogar direkt von der Grafikkarte nativ im Flash Player wiedergegeben werden.
- **Aufnahme von Video** (nur Flash Media Interactive Server) – Neben Live-Streaming bietet Flash Media Server die Möglichkeit, Videoinhalte entweder mit dem Live-Stream (beispielsweise zur Archivierung einer Veranstaltung) oder alleine (etwa beim Video-Messaging) aufzuzeichnen.
- **Unterstützung für mehrere Anwender** (nur Flash Media Interactive Server) – Flash Media Server kann Live-Streams für mehrere Clients bereitstellen und ermöglicht somit das Streaming von Audio, Video und Daten für videogestützte Kommunikationsanwendungen.

Auch wenn Streaming im Vergleich zum progressiven Download komplexer erscheint, sind die Verfahren in Wirklichkeit sehr ähnlich. Beide verwenden dieselben Komponenten und dieselben ActionScript-Befehle. Streaming bietet Entwicklern jedoch mehr Möglichkeiten zur Erstellung abwechslungsreicher, interaktiver Videoanwendungen.

Der einzige potenzielle Nachteil beim Streaming besteht in der Notwendigkeit spezieller Server-Software. So wie Sie für eine robuste Datenanwendung einen Webserver und einen Anwendungs-Server installieren müssen, erfordert eine robuste Medienlösung zusätzlich zum Webserver einen Streaming-Server.

Kunden, die größere Datenvolumen streamen, beliebte Inhalte anbieten oder bestimmte Zeitvorgaben einhalten müssen, jedoch keine eigene Infrastruktur aufbauen möchten, können die Dienste eines FVSS-Partners (Flash Video Streaming Service) zur Bereitstellung von Video über Adobe Flash Player in Anspruch nehmen. FVSS-Partner bieten eine optimierte, redundante Implementierung von Flash Media Server über ein zuverlässiges Content Delivery Network. Weitere Informationen zum Flash Video Streaming Service erhalten Sie unter www.adobe.com/go/fvss_de.

Wann ist die Bereitstellung per Streaming sinnvoll?

Streaming mit Flash Media Server eignet sich für Anwendungen mit folgenden Anforderungen:

- Streaming von Dateien, deren Bereitstellung länger als 30 Sekunden dauert oder mehr als 100 Kbps erfordert
- Erkennung der Bandbreite, sodass Videoinhalte in der besten Qualität bereitgestellt werden, die mit der verfügbaren Hardware möglich ist
- QoS-Kontrolle
- Verfolgung in Echtzeit
- Bereitstellung von Videoerlebnissen mit Echtzeit-Datenaustausch und Interaktivität
- Streaming von Live-Video und -Audio
- Aufzeichnung von Video und Audio
- Übertragung von mehr Streams bei geringerem Bedarf an Bandbreite

Falls für Ihre Website oder Ihren Blog der Austausch größerer Mengen von Video, Audio oder Daten erforderlich ist, können Sie das Anwendererlebnis mit den Funktionen von Flash Media Server optimieren.

Vergleich der Bereitstellungsverfahren

Die folgende Tabelle vergleicht die drei Bereitstellungsverfahren, die Flash Media Server für Video bietet.

	Eingebettetes Video	Progressiver Download	Streaming
Codierung	Video- und Audiodaten werden beim Import in Flash mit einem Sorenson Spark- oder VP6-E-Codec konvertiert. Alternativ lassen sich FLV-Dateien, die anderweitig codiert wurden, importieren und ohne erneutes Codieren in die Flash-Zeitleiste platzieren.	Videodateien werden in der integrierten oder eigenständigen Version von Adobe Flash Media Live Encoder, mit Flash Video Exporter und dem nicht linearen Bearbeitungs- oder Codierungsprogramm eines Drittanbieters oder mit einem eigenständigen Video-Codec wie Sorenson Squeeze oder On2 Flix codiert.	Wie beim progressiven Download. Außerdem können Live-Video-Feeds von Client-seitigen Webcams oder DV-Kameras erfasst und aufgezeichnet werden. Ferner wird Adobe Flash Media Live Encoder unterstützt, und Sie können Codierungsvariablen wie die Bit-Rate, die Anzahl der Frames pro Sekunde und den Umfang der Videowiedergabe programmatisch steuern.
Dateigröße	SWF-Dateien enthalten Video- und Audio-Streams sowie die Flash-Oberfläche. Die finale Datei ist somit sehr groß.	SWF- und Videodateien werden separat gespeichert. Die finale SWF-Datei ist vergleichsweise klein.	Wie beim progressiven Download
Start der Wiedergabe	Die Wiedergabe startet aufgrund der SWF-Dateigröße mit Verzögerung. Darunter leidet die Qualität des Anwendererlebnisses.	Die Wiedergabe beginnt recht schnell, nämlich sobald ausreichend Daten heruntergeladen wurden.	Die Wiedergabe beginnt sofort. Dieses Verfahren bietet die kürzeste Zeitspanne zwischen Ladevorgang und Wiedergabe.
Zeitleistenzugriff	Wenn Videomaterial in die Flash-Zeitleiste eingebettet ist, wird es in einzelnen Frames angezeigt, die wie andere Bühnenobjekte bearbeitet werden können.	Das Video wird nur zur Laufzeit abgespielt. Einzelne Frames sind in der Bühne nicht sichtbar. Zeitleistenereignisse können zu ausgewählten Zeiten während der Wiedergabe mittels ActionScript ausgelöst werden.	Wie beim progressiven Download
Bereitstellung	Jedes Mal, wenn ein Flash-Film bereitgestellt oder getestet wird, wird die gesamte Videodatei erneut veröffentlicht. Wenn die Videodateien geändert werden, müssen sie erneut manuell in die Zeitleiste importiert werden.	Videodateien werden nur zur Laufzeit referenziert. Die Veröffentlichung an eine SWF-Datei ist wesentlich schneller als die Einbettung des Videos. Videodateien lassen sich aktualisieren oder ändern, ohne dass die SWF-Datei erneut kompiliert werden muss.	Wie beim progressiven Download. Videodateien können dynamisch aus virtuellen Speicherorten wie Ihrem SAN (Storage Area Network), über FVSS oder einem anderen CDN abgerufen werden.
Bildrate	Video und SWF müssen dieselbe Bildrate aufweisen.	Video und SWF können unterschiedliche Bildraten verwenden.	Wie beim progressiven Download. Bei der Erfassung von Live-Video kann die Bildrate programmatisch gesteuert werden.
ActionScript-Zugriff	Die Wiedergabe und Steuerung von Videos erfolgt durch Bearbeitung der Filmwiedergabe in der Zeitleiste.	Die NetStream-Klasse dient zum Laden, Wiedergeben und Unterbrechen externer Videodateien. Bereits heruntergeladene Videodateien können durchsucht werden.	Wie beim progressiven Download. Mit Serverseitigem ActionScript lässt sich die Funktionalität erweitern, z. B. für die Synchronisierung von Streams, Erstellung von Server-seitigen Wiedergabelisten, intelligente Bereitstellung gemäß Client-Bandbreite u. v. m.
Komponenten	Keine videospezifischen Komponenten	Mit Medienkomponenten (ab Flash 8 Professional) können Sie externe Video- und Audiodateien mit Wiedergabesteuerungen (Abspielen, Anhalten, Suchen usw.) einrichten und anzeigen.	Wie beim progressiven Download. Außerdem können Sie die Kommunikationskomponenten von Flash Media Server für Live- und Multiway-Video verwenden.
Suche und Navigation	Erst nachdem die SWF-Datei vollständig heruntergeladen wurde, können Endanwender im Video suchen oder navigieren.	Nur die bereits heruntergeladenen Videodateien können durchsucht werden.	Der Endanwender kann jederzeit und überall im Video suchen.

	Eingebettetes Video	Progressiver Download	Streaming
Web-Ausgabe	Erst nachdem die SWF-Datei vollständig heruntergeladen und gespeichert wurde, kann das Video abgespielt werden.	Videodateien werden progressiv heruntergeladen, im Zwischenspeicher abgelegt und dann von der lokalen Festplatte aus wiedergegeben. Der Video-Clip muss nicht als Ganzes gespeichert werden.	Videodateien werden von Flash Media Server gestreamt und auf dem Client-Bildschirm abgespielt. Bereits wiedergegebene Daten werden unmittelbar aus dem Speicher entfernt.
Performance	Die Synchronisation von Bild und Ton wird nach etwa 120 Sekunden Abspieldauer beeinträchtigt. Die Gesamtspieldauer hängt vom verfügbaren Arbeitsspeicher des Wiedergabesystems ab.	Bessere Performance gegenüber eingebettetem Video mit höherer Auflösung und zuverlässigerer Audio-synchronisierung. Beste Bildqualität, die einzig vom verfügbaren Festplattenspeicher auf dem Wiedergabesystem abhängt	In Bezug auf die Web-Ausgabe ist dieses Verfahren effizienter, da automatisch die jeweils optimale Bit-Rate ausgewählt wird.
Steuerung des Video-Streams	Keine	Keine	Vollständige Kontrolle über Zeitpunkt der Bereitstellung und Art der Streaming-Inhalte. Erweiterte Zugriffskontrolle mithilfe von Serverseitigem ActionScript.
Unterstützung für Live-Video	Nein	Nein	Ja
Kompatibilität	Flash Player ab Version 6	Flash Player ab Version 7	Flash Player ab Version 6

Übersicht über die Funktionen von Flash Media Server 3.5

Adobe Flash Media Server 3.5 wartet mit zahlreichen Neuerungen auf, die v. a. die Aspekte Performance beim Streamen und Sicherheit bedienen. Aufgrund der engen Integration mit dem universellen Flash Player, der leistungsstarken Server- und Client-seitigen APIs sowie eines hervorragenden Preis-Leistungsverhältnisses sind die Produkte der Flash Media Server-Familie die ideale Wahl für Entwickler, die robuste RIAs und Lösungen für effizientes Video-Streaming erstellen möchten.

Zu den wichtigsten Funktionen und Verbesserungen der Flash Media Server 3.5-Familie gehören:

- **Performance** – Optimale Nutzung der Hardware-Kapazitäten und geringere Bereitstellungskosten
- **Sicherheit** – Zuverlässiger Schutz für Medieninhalte
- **Qualität** – Bereitstellung hochwertiger Inhalte (bis HD), die für die Bandbreite des Endanwenders optimiert werden
- **Live-Streaming** – Maximale Kontrolle und umgehende Wiedergabe hochqualitativer Live-Videos
- **Bereitstellung für mobile Endgeräte** – Erstellung einer eigenen Version für die Wiedergabe auf mobilen Endgeräten

Performance von Flash Media Server 3.5

Flash Media Server 3 hat die Anzahl der Streams, die mit einer Server-Lizenz gleichzeitig bereitgestellt werden können, signifikant erhöht. Mehr Streams können mit weniger Servern und zu geringeren Kosten übertragen werden. Die Leistung von Linux-Implementierungen hat sich sogar um 300 % verbessert. Für Flash Media Server 3.5 wurde die Performance nochmals optimiert.

Von den zahlreichen Funktionen, die zur höheren Leistung beitragen, werden auf den nächsten Seiten folgende erläutert:

- Performance-Optimierung allgemein
- Gedrosselte Verbindungen
- Erweiterter Prozessumfang
- Automatisches Schließen nicht genutzter Verbindungen
- Enhanced RTMP (RTMPE)
- Integrierte Bandbreitenerkennung mit Unterstützung für dynamisches Streaming
- Prozessaufgliederung
- Verteilte Core-Prozesse

Einflussfaktoren für die Performance

Im Folgenden wird erläutert, welche Bedingungen die Performance von Flash Media Server 3.5 beeinflussen. Die Performance wird anhand der Anzahl gleichzeitiger Streams gemessen, die bei vorgegebener CPU-Auslastung unterstützt werden. Mit dieser Erkenntnis können Sie leichter ermitteln, wie viele Server Sie implementieren müssen.

Die Anzahl der Streams, die ein Server bereitstellen kann, hängt u. a. von folgenden Faktoren ab:

- **Protokoll** – RTMP ist das leistungsfähigste Protokoll, gefolgt von RTMPE.
- **Bit-Rate des Videos (Qualität)** – Je geringer die Bit-Rate des Videos, desto mehr Streams können gleichzeitig vom Server bereitgestellt werden.
- **Plattform** – Mit Linux Red Hat 4 können mehr Verbindungen bei geringerer CPU-Auslastung genutzt werden.
- **Hardware** – Parameter wie RAM, CPU, Festplatten- und Netzwerkgeschwindigkeit bestimmen die verfügbare Streaming-Kapazität.
- **Konfiguration** – Flash Media Server 3.5 ist für die meisten Anwendungszwecke vorkonfiguriert. Unter Umständen lässt sich die Streaming-Performance durch kleine Änderungen an der Konfiguration verbessern.
- **Komplexität der Anwendung** – Wenn Sie eigene Plug-ins verwenden oder komplexe Serverseitige Anwendungslogik entwickeln, kann sich dies sowohl positiv als auch negativ auf die Performance auswirken.
- **Verwendung** – Bestimmte Interaktionen mit Videoinhalten auf Anwenderseite können Auswirkungen auf die Server-Leistung haben. Dazu zählen beispielsweise der Aufbau und Abbruch von Verbindungen, Suchvorgänge oder Unterbrechungen. Näheres zu diesem Thema erfahren Sie im nächsten Abschnitt.

Performance-Kennzahlen für Flash Media Server 3.5

Flash Media Server 3 bot auf Windows-Systemen bereits die zweifache, auf Linux-Systemen sogar die dreifache Leistung. In der folgenden Tabelle sind die Performance-Kennzahlen für Flash Media Server in zwei Szenarien beschrieben:

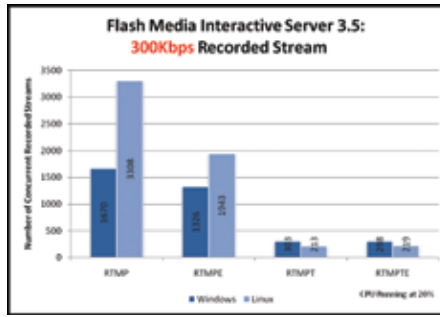
- VOD (Video-on-Demand) – Unter Verwendung aufgezeichneter Videodaten
- Live-Video – Unter Verwendung von Live-Streams von Flash Media Live Encoder 2

Die Messwerte wurden für folgende Hardware-Konfigurationen ermittelt. Für die Berechnung wurden 2 x 1 Gbps-Netzwerk-Adapter verwendet, die veröffentlichten Ergebnisse basieren jedoch auf einem Durchsatz von 1 Gbps.

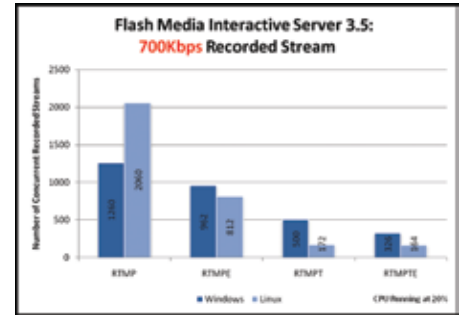
Server-Hardware	HP ProLiant DL360 G4p-Server mit 3,6 GHz-Dual-Core-Prozessor und Xeon-Prozessoren mit Hyperthreading Server-Konfiguration: 3,5 GB mit 200 MHz DDR-SDRAM, SCSI2, 64 GB 10K RPM RAID 0-Festplattenspeicher und verbundene 1 GB Intel Pro 100 XF Ethernet-Karten mit 133 MHz
Betriebssysteme	Windows Server 2003 mit Service Pack 1, Linux Red Hat 4, Kernel 2.6.9-22
Flash Player	Version 9,0,115,0
Testmedien	• 700 Kbps: FLV, 117 Minuten, 763 MB mit On2 VP6-Codec • 300 Kbps: FLV, 53 Minuten, 105 MB mit On2 VP6-Codec • 128 Kbps: MP3, 52 Minuten, 61 MB mit Mp3tag ID3v2.3
Flash Media Server	Version 3: Standardkonfiguration mit standardmäßiger Größe für Datenpaket und Cache

Video-on-Demand

Zum einfacheren Vergleich zeigt das erste der folgenden Diagramme die Gesamtzahl der Streams, die mit nur 20-prozentiger CPU-Auslastung erzielt wurden. Unter Linux wurde die optimale Nutzung eines 1 Gbps Netzwerk-Adapters mit einer CPU-Beanspruchung von etwas mehr als 20 % erreicht. Das zweite Diagramm veranschaulicht die Leistung jedes Protokolls bei höherer CPU-Nutzung.

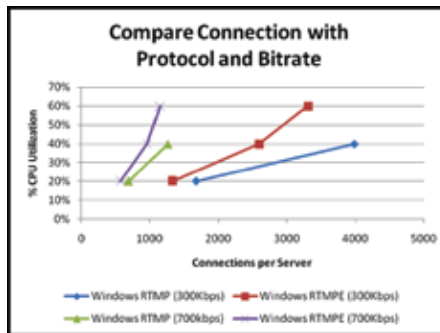


Videoaufzeichnung mit 300 Kbps (maximal 1 Gbps)

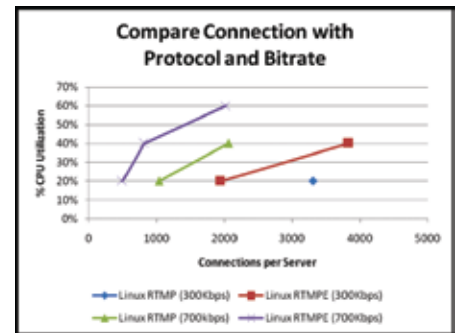


Videoaufzeichnung mit 700 Kbps (maximal 1 Gbps)

Die nächsten Diagramme zeigen die Anzahl der gleichzeitigen Streams bei unterschiedlicher CPU-Auslastung sowie verschiedenen Protokollen und Bit-Raten. Beachten Sie, dass bei einer höheren CPU-Nutzung mehr Streams bereitgestellt werden können. Die Höchstgeschwindigkeit wurde auf 1 Gbps beschränkt, und es wurde nie die komplette CPU-Auslastung erreicht. Höhere Datenraten führten zu schnellerer Lastenoptimierung, während bei niedrigeren Datenraten zur Bereitstellung ähnlicher Verbindungen eine höhere CPU-Auslastung erforderlich war.



Windows

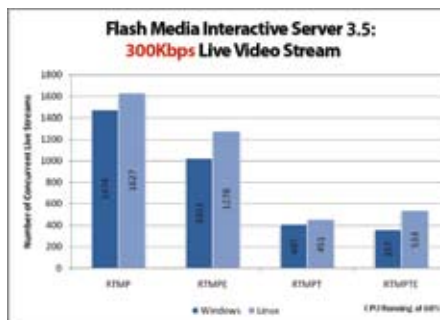


Linux

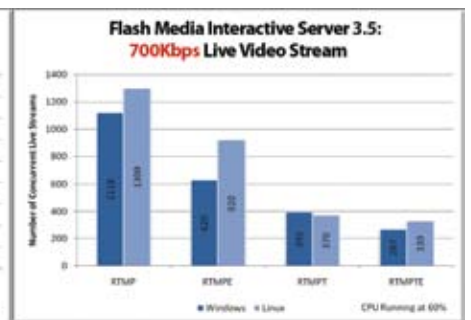
Mit RTMPE reduzierte sich die Leistung bei ähnlich hoher CPU-Beanspruchung im Durchschnitt um nur 25 % bis 30 %. Wenn Sie mit RTMPE arbeiten, können Sie davon ausgehen, dass die CPU zwar stärker ausgelastet ist, aber Sie können bei einer CPU-Nutzung von weniger als 70 % eine Netzwerkgeschwindigkeit von 1 Gbps erreichen.

Live-Video

Für das Streaming von Live-Video über Flash Media Server gelten dieselben Bedingungen wie für Video-on-Demand. Die folgenden Diagramme veranschaulichen die Gesamtleistung bei Live-Video. Für die Tests wurde Live-Video von Flash Media Live Encoder 2 mit dem Video-Codec On2 VP6 und dem Audio-Codec MP3 gestreamt.



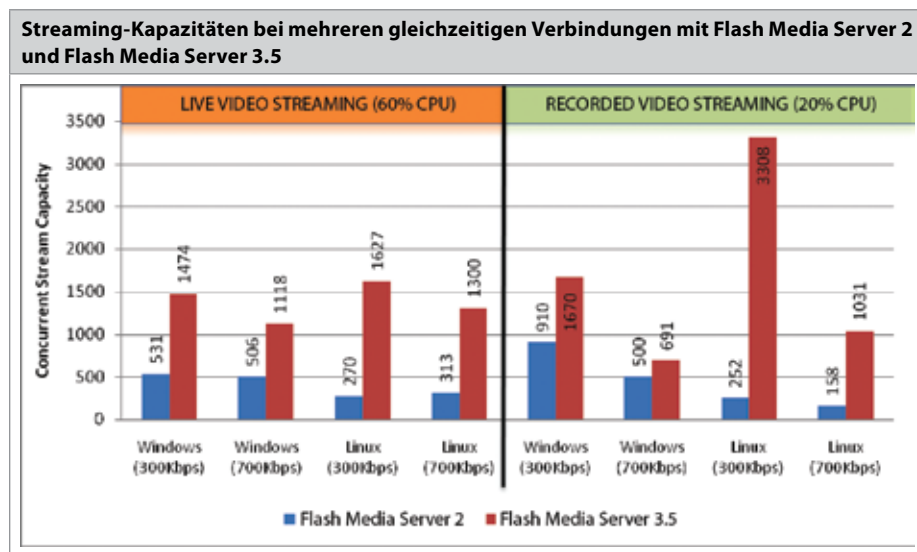
Live-Video mit 300 Kbps



Live-Video mit 700 Kbps

Vergleich mit Flash Media Server 2

Flash Media Server 3.5 unterstützt fast doppelt so viele aktive Streams wie Flash Media Server 2. Performance-Steigerungen wurden sowohl bei Live- als auch bei VOD-Anwendungen festgestellt. Zudem sind deutlich mehr Verbindungen bei geringeren Bit-Raten möglich als bei Flash Media Server 2.



Alle Tests erfolgten mit RTMP. Die Werte für Live-Streams wurden bei einer CPU-Last von 60 % ermittelt, für On-Demand-Streams bei 20 %.

Gedrosselte Verbindungen

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Server 3.5 enthält effiziente Werkzeuge für die Verwaltung von Verbindungen, die die Qualität für Anwender aufrechterhält, die bereits mit dem Server verbunden sind. Wenn neue Anwender versuchten, sich mit einem häufig genutzten Stream zu verbinden, führte dies bisher oft zu Wiedergabestörungen bei den bereits verbundenen Anwendern. Die Drosselung der Verbindungen eröffnet mehrere Möglichkeiten zur Erhaltung der Servicequalität:

- Einschränkung der Anzahl von Threads für eingehende Verbindungen
- Konfiguration der maximalen Anzahl von Verbindungen pro Sekunde. Wird dieser Grenzwert überschritten, werden Verbindungsanforderungen nicht sofort verarbeitet, sondern in die Warteschlange im Netzwerk-Stack des Betriebssystems verschoben. Die Anzahl der Verbindungen in der Warteschlange des Betriebssystems ist ebenfalls begrenzt. Wenn die Warteschlange voll ist, werden weitere Verbindungsversuche der Clients zurückgewiesen. Beachten Sie, dass der Grenzwert pro Listener gilt. Wenn Ihr Server beispielsweise so konfiguriert ist, dass er auf Verbindungsanforderungen für die Ports 1935 und 80 wartet, gilt die maximale Verbindungsanzahl für beide Ports. Bei einem Maximalwert von 10 Verbindungen pro Sekunde wären somit insgesamt 20 Verbindungen pro Sekunde möglich.
- Vorgabe einer Längenbegrenzung für die Warteschlange. Sobald der Höchstwert erreicht wird, weist der Server Verbindungen zurück, um die Qualität für die aktuell verbundenen Anwender zu erhalten.

Diese Einstellungen lassen sich jederzeit vom Server-Administrator konfigurieren.

Erweiterter Prozessumfang

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Mit Flash Media Server 3.5 lässt sich der Umfang von Server-Prozessen flexibel konfigurieren. Bei Flash Media Server 2 konnten Server-Administratoren einen Prozess entweder für jeden virtuellen Host, für jede Anwendung oder für jede Instanz ausführen. Daher waren oft sehr viele Prozesse erforderlich. Flash Media Server 3.5 beschränkt die Menge der Prozesse auf eine vorab festgelegte Anzahl. Der Server teilt die Prozesse anschließend unter allen aktiven virtuellen Hosts auf. Wenn Sie z. B. 10 Prozesse festlegen und 20 virtuelle Hosts aktiv sind, werden die Prozesse automatisch diesen Hosts zugewiesen.

Automatisches Schließen nicht genutzter Verbindungen

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Bis Flash Media Server 2 wurden Client-Server-Verbindungen vom Client aus gesteuert. Wurde die Verbindung unerwartet abgebrochen, bestand das Risiko, dass die Verbindung für unbegrenzte Dauer offen blieb. Bei Flash Media Server 3.5 kann der Server längere Zeit nicht genutzte Verbindungen erkennen und automatisch schließen. Die Leerlaufzeit wird in der Datei `Server.xml` definiert; der Standardwert beträgt 60 Sekunden.

Enhanced RTMP

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

RTMP ist ein patentiertes Protokoll, mit dem Flash Media Server Daten versendet und empfängt. In Flash Media Server 3.5 wurde die Leistung und Sicherheit von RTMP deutlich verbessert.

Für Flash Media Server 3 und die Nachfolgerversion wurde RTMP überarbeitet, um die Sicherheit und Effizienz des Protokolls zu erhöhen. Dabei wurden u. a. strengere Vorgaben für den RTMP-Handshake zwischen Flash Player und Flash Media Server eingeführt. Die neue SWF-Verifizierung stellt sicher, dass die Anforderung der SWF-Verbindung auch von der erwarteten Quelle stammt. Damit wird der Missbrauch von FLV-Daten und Bandbreiten verhindert.

Bis Flash Media Server 2 war der Einsatz von SSL die einzige Möglichkeit, Stream-Daten zu verschlüsseln. Das führte jedoch zu signifikant langsameren Verbindungen. Seit Flash Media Server 3 sichert RTMPE den Kommunikationskanal durch 128-Bit-Verschlüsselung zwischen Client und Server, ohne dass die Performance wie bei SSL leidet und ohne die Notwendigkeit eines Zertifikats.

Ähnlich wie bei der Implementierung von SSL (RTMPS) müssen Sie im Verbindungs-String des Clients einfach RTMPE als Kommunikationsprotokoll angeben. Beispiel:

```
nc.connect("rtmpe://www.example.com/myApplication")
```

RTMPE verhält sich exakt wie RTMP – mit dem Unterschied, dass hier die Daten verschlüsselt übertragen werden. Mit RTMPTE können Sie die verschlüsselte Verbindung zusätzlich tunnelt.

Dynamisches Streaming

*Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 10 oder höher erforderlich)*

Dynamisches Streaming ist eine neue Kontrollfunktion, mit der Sie nahtlos zwischen verschiedenen Versionen eines einzelnen Video-Streams wechseln können, die mit verschiedenen Bit-Raten codiert sind. Mit dieser Funktion lassen sich Medienanwendungen mühelos an Clients mit unterschiedlicher Ausstattung anpassen, wie etwa für mobile Endgeräte mit geringerer Prozessorleistung und kleineren Bildschirmen oder Clients mit langsameren Internet-Verbindungen. Die Qualität bleibt dabei unverändert. Dynamisches Streaming kann außerdem zum Austauschen von Inhalten in einer Wiedergabeliste genutzt werden. Zu diesem Zweck definieren Sie Ereignisse, die den Austauschvorgang auslösen.

Integrierte Bandbreitenerkennung

Funktion in *Flash Media Streaming Server* und *Flash Media Interactive Server*
(Flash Player 6 oder höher erforderlich)

Die in Flash Media Server integrierte Bandbreitenerkennung ist standardmäßig aktiviert. Die aktuelle Technologie setzt auf native statt skriptbasierte Bandbreitenerkennung und erhöht auf diese Weise die Performance und Skalierbarkeit der Software. Um die native Bandbreitenerkennung zu nutzen, vergewissern Sie sich, dass die Funktion aktiviert ist, und erstellen Sie Client-seitigen Code, der in Flash Media Server integrierte Funktionen aufruft. Die Funktion ist mit ActionScript 2 und 3 kompatibel. Auf Server-Seite muss kein Code programmiert werden.

Auf Wunsch können Sie die native Bandbreitenerkennung deaktivieren und stattdessen ein Server-seitiges Skript verwenden. Dies empfiehlt sich zum Beispiel, wenn Sie vorhandenen Code wiederverwenden möchten. Greifen Sie in dem Fall auf die Datei `main.asc` zu, die Adobe für die Bandbreitenerkennung bereitstellt, und deaktivieren Sie die native Erkennung in der Datei `Application.xml`. Weitere Informationen finden Sie im *Entwicklerhandbuch für Adobe Flash Media Interactive Server 3.5*.

Prozessaufgliederung

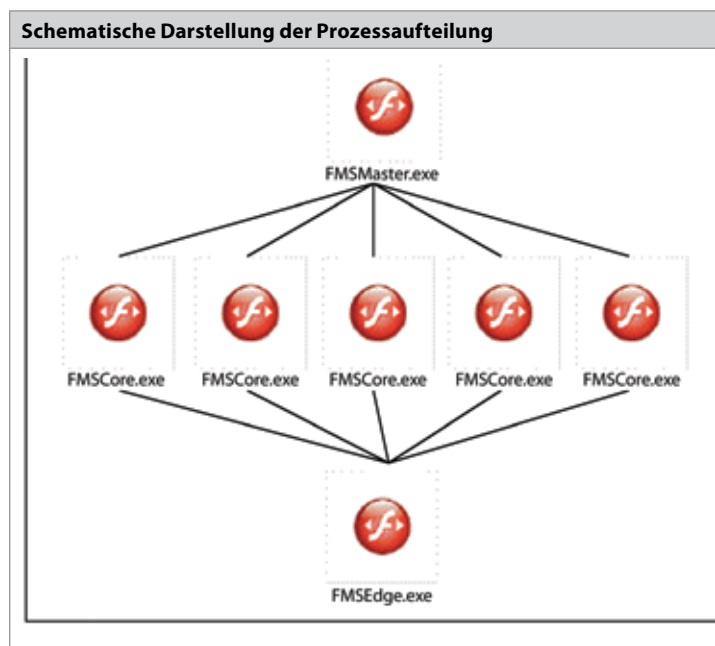
Funktion in *Flash Media Interactive Server* (Flash Player 6 oder höher erforderlich)

Mithilfe von Flash Media Interactive Server 3.5 können Sie festlegen, wie Server-Prozesse aufgeteilt werden. Wenn Sie den Server starten, beginnt ein Prozess namens `FMSMaster.exe` (Windows) bzw. `fmsmaster` (Linux). Anwendungsinstanzen werden im Prozess `FMSCore.exe` (Windows) bzw. `fmscore` (Linux) ausgeführt.

Flash Media Server umfasst mehrere Prozesse. Bei einer Standardinstallation werden vier Arten von Prozessen ausgeführt: „Master“, „Edge“, „Core“ und „Admin“.

Der Master-Prozess ist ein Monitor, der bei Bedarf Core-Prozesse startet. Es kann jeweils nur ein Master-Prozess ausgeführt werden. Die Anzahl der Core-Prozesse ist unbegrenzt. In der Datei `Application.xml` legen Sie fest, wie Anwendungen den Server-Prozessen zugeordnet werden. Sie können die Anzahl der Prozesse, den Prozessumfang, die Prozessdauer und die Anzahl der zulässigen Prozessfehler (bevor der Core-Prozess deaktiviert wird) angeben.

Die Client-Server-Verbindung erfolgt immer über den Edge-Prozess (unabhängig davon, ob eine Edge- oder Origin-Konfiguration vorliegt). Der Master-Prozess sorgt für den reibungslosen Start bzw. den Übergang von einem Core-Prozess zum nächsten. Clients bauen nie eine Verbindung mit dem Master-Prozess auf, und dieser kann nicht konfiguriert werden. Die folgende Abbildung veranschaulicht die Prozessarchitektur.



Verteilte Core-Prozesse

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Um die Leistung und Zuverlässigkeit Ihres Servers weiter zu verbessern, können Sie Verbindungen für einen bestimmten Geltungsbereich mehreren Prozessen zuweisen. Für den Prozessumfang „Adapter“ können Sie beispielsweise für jeden virtuellen Host Verbindungen auf beliebig viele Core-Prozesse verteilen.

Die Verteilung der Core-Prozesse kann genauso wie die Definition des Prozessumfangs die Leistung des Servers erhöhen. Beim verteilten Rechnen wird mehr RAM für das Caching und mehr Threading für prozessintensive Verbindungen verfügbar. Bei einer hybriden Multiway-Anwendung oder Live-Lösung ist dieses System nicht anwendbar, weil für den Datenaustausch Verbindungen demselben Core-Prozess zugeordnet sein müssen.

Sicherheit

Verschlüsselung

Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 10 oder höher erforderlich)

Flash Media Server 3.5 unterstützt nun auch Dateien, die durch Funktionen von Adobe Flash Media Rights Management Server geschützt sind. Flash Media Server 3.5 kann unterzeichnete und verschlüsselte Streaming- und interaktive Inhalte mühelos bereitstellen – auch über den integrierten HTTP-Server. Weitere Informationen zu Flash Media Rights Management Server finden Sie unter www.adobe.com/de/products/flashmediarightsmanagement.

SWF-Verifizierung

Erweiterte Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 9,0,115,0 oder höher erforderlich)

Die SWF-Verifizierung ist eine Sicherheitsfunktion von Flash Media Server 3.5, mit der Sie festlegen können, welche SWF-Dateien eine Verbindung mit Ihrem Server aufbauen können. Wenn Sie diese Funktion nicht implementieren, kann jede SWF-Datei mit der geeigneten URI (Uniform Resource Identifier) und Anwendungsbezeichnung uneingeschränkt Verbindungen herstellen und potenziell auf Ihre Streams und Server-Ressourcen zugreifen.

Mit der SWF-Prüffunktion können Sie den Server so konfigurieren, dass er bei jeder SWF-Datei, die versucht, auf eine bestimmte Anwendung oder Anwendungsinstanz zuzugreifen, verifiziert, dass sie einer Gruppe vorab genehmigter SWF-Dateien angehört. Zur Aktivierung dieser Funktion speichern Sie einfach eine Kopie der genehmigten SWF-Datei im Anwendungsverzeichnis und aktivieren die Funktion in der Datei `Application.xml`. Sobald eine SWF-Datei sich mit dem Server verbindet, prüft dieser, ob die Datei einer SWF-Datei in Ihrem Anwendungsverzeichnis entspricht, und lässt die Verbindung gegebenenfalls zu.

Um eine SWF-Datei für jede Instanz einer Anwendung zu genehmigen, platzieren Sie sie in das SWF-Verzeichnis des Anwendungsordners. Um eine SWF-Datei für eine bestimmte Instanz zu genehmigen, platzieren Sie sie im SWF-Verzeichnis des Instanzordners. Diese Funktion wurde in Flash Media Server 3.5 überarbeitet, sodass zur Verifizierung der SWF-Dateien keine Kopien auf dem Server erforderlich sind. Mit dem Datei-Plug-in können Sie SWF-Dateien abrufen, die extern gespeichert sind.

Hinweis: Beim Einsatz einer Adobe AIR-Lösung kopieren Sie die kompilierte SWF-Datei entweder in das Adobe AIR-Paket oder auf den Server, damit sie zur Verifizierung verfügbar ist. Alternativ können Sie mit dem Datei-Plug-in auf einen externen Speicherort verweisen.

Standardisierte Server-Umleitung

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Server 3.5 unterstützt die Stream-Umleitung mit RTMP, die im Wesentlichen einer HTTP 302-Umleitung entspricht. Diese Funktion kann mithilfe eines Zugriffs-Plug-ins oder mit einer ActionScript-Routine auf Server-Seite aktiviert werden. Wenn Sie einen Zugriffs-Adapter auf einem Edge-Server implementieren, können Sie zum Beispiel über die Umleitung einen Client darüber informieren, dass sich das angeforderte Video nicht am erwarteten Speicherort befindet, und eine neue URI für den Stream angeben. Die Funktion eignet sich ebenso für andere Systeme zur Lastenverteilung von Flash Media Server sowie zur Organisation von Inhalten.

Eine überarbeitete Server- und Client-seitige API ermöglicht die Verwaltung der Server-Umleitungsdaten. Beachten Sie, dass Sie hierzu die aktualisierte FLVPlayback-Komponente benötigen. Weitere Informationen finden Sie im *Entwicklerhandbuch für Flash Media Server 3.5-Plug-ins*.

Server-seitige Plug-in-Architektur

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Adobe Flash Media Interactive Server 3.5 unterstützt in C++ geschriebene Plug-ins, mit denen Sie die Server-Funktionalität erweitern können. Es gibt drei Plug-in-Klassifikationen: Datei, Autorisierung und Zugriff.

Jedes Plug-in kann direkt eingesetzt oder Ihren individuellen Anforderungen entsprechend angepasst werden. Flash Media Server umfasst verschiedene neue Beispiel-APIs bzw. API-Vorlagen, die Sie durch eigene Funktionen erweitern können. Von jedem Plug-in können mehrere Versionen eingesetzt werden. Weitere Informationen finden Sie im *Referenzhandbuch für Adobe Flash Media Server-Plug-in-APIs*.

Datei-Plug-in

Mit dem Datei-Plug-in können Entwickler eigene asynchrone Funktionen erstellen, die genau kontrollieren, wo und wie Flash Media Server Inhalte aus einem Dateisystem oder Dienst liest. Unterstützt werden lokale und externe Dateisysteme. Für die Kommunikation kann jedes Protokoll, z. B. HTTP oder FTP, verwendet werden.

Asynchrone Lese- und Schreibvorgänge sind eine wesentliche Erweiterung der aktuellen Versionen von Flash Media Server. Die älteren Versionen unterstützten nur synchrone Zugriffe auf ein lokalisiertes Dateisystem. Jede Anforderung eines Lesevorgangs in einer Datei musste so lange warten, bis die vorherigen Anforderungen in der Warteschlange abgeschlossen waren. Das Datei-Plug-in baut auf den neuen asynchronen Zugriff auf und erleichtert die Eingabe und Ausgabe von netzwerkbasierten und externen Dateien.

So können Sie beispielsweise externe Dateien per HTTP abrufen und über Flash Media Server an Clients übermitteln. Für den Streaming-Vorgang können Sie die Datei jetzt von jedem zugeordneten Speicherort auslesen, statt wie in der Vergangenheit nur vom Stream-Ordner der aktuellen Flash Media Server-Implementierung. Diese Funktion ist nur für VOD-Inhalte oder zur SWF-Verifizierung anwendbar.

Autorisierungs-Plug-in

Das Autorisierungs-Plug-in dient der strikten Kontrolle von Client-Zugriffen auf die Server-Ereignisse `NetConnection` und `NetStream` und unterstützt Sie u. a. bei folgenden Aufgaben:

- Zugriff auf Client-Statistiken (z. B. eingehende und ausgehende Datenmengen) vom Server aus, ohne dass die Server-Leistung beeinträchtigt wird
- Autorisierung von Verbindungen zum Server
- Autorisierung von Publishing-, Wiedergabe- oder Suchvorgängen innerhalb eines Streams
- Zuordnung von logischen zu physischen Stream-Anforderungen
- Anwendung von Sicherheitsrichtlinien auf Stream-Anforderungen
- Abbruch einer Verbindung zwischen Client und Server
- Bereitstellung von Inhalten an Clients nach Standort, Abonnement, Uhrzeit und Dauer des Zugriffs bestimmter Anwender auf bestimmte Streams usw.

Das Autorisierungs-Plug-in kann beispielsweise für eine Anwendung mit verschiedenen Mitgliedschaftskategorien konfiguriert werden, sodass zahlende Mitglieder einen HD-Stream und Gäste einen SD-Stream empfangen. Dabei wird die Verbindungsanforderung abgefangen, um die Mitgliedschaft des Clients zu überprüfen, und dann wird die Verbindung gefiltert und der entsprechenden Stream-Datei zugeordnet. Das Plug-in kann auch für die Zugriffskontrolle, Protokollierung oder Implementierung benutzerdefinierter Sicherheitsfunktionen verwendet werden. Durch die Aneinanderreihung mehrerer Autorisierungs-Plug-ins können Sie eine Vorgangssequenz erstellen, die Zugriffe auf Ihre Inhalte nach präzisen Kriterien filtert. Mit den optimierten Funktionen für Client-Statistiken lassen sich außerdem eigene Überwachungslösungen auf C++-Anwendungsebene hinzufügen, ohne dass die Server-Leistung beeinträchtigt wird.

Zugriffs-Plug-in

Das Zugriffs-Plug-in ist ein weiteres Sicherheits-Tool für Entwickler, das eine zusätzliche Ebene zwischen Client und Server schafft. Das Plug-in fängt Verbindungsanforderungen ab und prüft sowohl den Client als auch den Server, um zu ermitteln, wie die Anforderung verarbeitet werden soll. Mit dem Zugriffs-Plug-in lassen sich folgende Aufgaben ausführen:

- Abgleich der Anmeldedaten mit einer Datenbank zur Legitimierung der Verbindung. Wird die Anforderung genehmigt, können Sie die Datenbank mit Informationen zum Client und einem Datensatz zur Verbindung aktualisieren.
- Ermittlung der aktuellen Server-Last und der Zulässigkeit der Client-Anforderung (mit anschließender Genehmigung, Zurückweisung oder Umleitung)
- Einrichtung von Lese- und Schreibberechtigungen für Dateien und Ordner auf dem Server
- Definition von Zugriffsrechten für Audio- und Video-Bitmap-Daten in Streams

Erweiterte Cache-Unterstützung

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Das Caching-Verhalten von Flash Media Server wurde optimiert. In puncto Speicherverwaltung und Server-Performance sind Flash Media Server 3 und 3.5 weitaus effizienter als ihre Vorgängerversionen. Server-Administratoren können jetzt einen Maximalwert für den RAM-Bereich festlegen, der vom Cache belegt wird.

In den Cache-Einstellungen von Flash Media Server 2 wurde die Anzahl der Dateien festgelegt, die im Cache-Ordner abgelegt werden sollten. In jeder Datei stand eine vordefinierte Menge von Segmenten zum Beschreiben zur Verfügung. Wenn der Server vollständig ausgelastet war, wurden nicht alle Segmente einer Datei verwendet. In der aktuellen Version ist die Zahl der Segmente maßgebend, nicht mehr die der Dateien. Dieses Verfahren ist wesentlich effizienter.

Kompatibilität mit IPv6

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Speziell für Behörden und andere Regierungseinrichtungen ist die Unterstützung von IPv6, dem Nachfolger von IPv4 (192.168.0.1), ein entscheidendes Auswahlkriterium. IPv6 erweitert die Adressierungsoptionen von 32 Bit auf 128 Bit und wird von Flash Media Server 3.5 umfassend unterstützt. Die Verwaltungskonsolle von Flash Media Server ist ebenfalls vollständig kompatibel mit IPv6.

HD-Video

Für die Datenrate und Videoqualität gelten keine Einschränkungen. Flash Media Server 3.5 unterstützt die klassischen Formate FLV (mit den Codecs Sorenson Spark oder On2 VP6/MP3) und MPEG-4 (mit den Codecs H.264/AAC+) sowie jede beliebige Datenrate – 50 Kbps-Streaming an ein DFÜ-Medium oder mobiles Endgerät oder bis zu 20 Mbps für HD-Video.

Videoprofile in Flash

Das gewählte Videoprofil bestimmt die Qualität Ihres Videos. Die Adobe Flash-Plattform unterstützt Light (LT), Standard (SD) und High Definition (HD).

- Video LT – Bereitstellung auf mobilen Endgeräten via Flash Lite 3 (nur On2 VP6)
- Video SD – Web-Video in SD (On2 VP6 und H.264)
- Video HD – High-Definition-Video (On2 VP6 und H.264)

Wie Sie folgender Tabelle entnehmen können, ist jedes Profil in drei Level unterteilt.

Level	Codec	Zielauflösung / Datenrate
LT Level 1	VP6 Mobile	GSM
LT Level 2	VP6 Mobile	3G
LT Level 3	VP6 Mobile	3G-High
SD Level 1	On2 VP6 und H.264	160 x 112
SD Level 2	On2 VP6 und H.264	320 x 240
SD Level 3	On2 VP6 und H.264	640 x 480
HD Level 1	On2 VP6 und H.264	640 x 480
HD Level 2	On2 VP6 und H.264	1280 x 720
HD Level 3	On2 VP6 und H.264	1920 x 1080

(Quelle: www.adobe.com/de/products/hdvideo/supported_technologies/h264.html)

Die Bildrate reicht meist von 5 bis 30 Frames pro Sekunde (fps). Höhere Bildraten und Bildschirmauflösungen erfordern bei der Wiedergabe eine höhere Rechnerleistung. Flash Player 9 unterstützt die Hardware-beschleunigte Wiedergabe im Vollbildmodus.

H.264-Video

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Der von Flash Media Server 3.5 unterstützte Codec H.264 liefert erstklassige Videoqualität. Für Video-Streaming werden von Flash die folgenden Komponenten des Standards MPEG-4 Teil 10 unterstützt:

- **Baseline** – Weit verbreitet in Anwendungen für Videokonferenzen sowie mobilen Anwendungen, die auf Endgeräten mit geringer Rechenleistung ausgeführt werden.
- **Main Profile (MP)** – Dieses ursprünglich für Broadcast- und Speicheranwendungen konzipierte Profil wurde größtenteils durch das „High Profile“ verdrängt.
- **High Profile (HiP)** – Das wichtigste Profil für TV-Anwendungen und die Disk-basierte Datenspeicherung. HiP wird von beiden DVD-Formaten mit HD-Auflösung verwendet: HD DVD und Blu-ray Disc.
- **High 10 Profile (Hi10P)** – Ein Profil, das auf dem „High Profile“ aufbaut und für die Auflösung der decodierten Pixeldaten 10 Bits/Sample definiert.

Flash Player 9 und höher unterstützt darüber hinaus das Streaming oder die progressive Wiedergabe von MPEG-4-Containerformaten wie MP4, M4A, MOV, MP4V, 3GP und 3G2, sofern diese H.264-Video und/oder HE-AAC-Audio enthalten.

Weitere Informationen zur Implementierung von H.264 finden Sie unter www.adobe.com/de/products/hdvideo/supported_technologies/h264.html.

Die vollständigen Systemanforderungen für die Wiedergabe von HD-Video im Flash Player finden Sie unter www.adobe.com/de/products/hdvideo/systemreqs.html.

HE-AAC-Audio

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Flash Media Server 3.5 unterstützt das Streaming von HE-AAC-Audio mit den Codec-Profilen gemäß ISO/IEC 14496-3 (MPEG-4 Teil 3):

- **AAC Main** erweitert die MPEG-2-Version des AAC-Codex um wahrnehmbare Geräuschformung (Noise Shaping), bietet höhere Qualität bei geringeren Bit-Raten und unterstützt in einem Audioprojekt bis zu fünf Kanäle sowie zusätzlich einen Subwoofer-Kanal (5.1).
- **AAC LC (Low Complexity)** ist nur geringfügig weniger effizient als das AAC-Main-Verfahren und erfordert eine geringere Rechenleistung zur Codierung und Decodierung. AAC LC ist das für Anwendungen mit geringer Bit-Rate, wie z. B. Streaming, optimierte Profil.

- **HE AAC v2 (auch HE-AAC+, eAAC und aacPlus v2)** ist eine Erweiterung des AAC-Codecs, die mithilfe von Spektralband-Replikation (SBR) und parametrischem Stereo (PS) insbesondere bei niedrigen Bit-Raten effizientere Codierungen der Frequenzbereiche bzw. der Stereosignale erzielt. HE-AAC v2 unterstützt bis zu 48 Audiokanäle sowie 5.1- und 7.1-Surround-Sound.

Weitere Informationen zur Implementierung von HE-AAC finden Sie unter www.adobe.com/de/products/hdvideo/supported_technologies/heaacv2.html.

On2 VP6-S

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 9,0,115,0 oder höher erforderlich)

Flash Media Server 3.5 unterstützt das Streaming mit dem neuen Codec-Profil VP6-S, das mit Flash Player 9,0,115,0 verfügbar ist. VP6-S vereinfacht die Codierung und Decodierung von hochauflösendem Vollbildvideo und damit die Übertragung und Wiedergabe von HD-Videos auf Heim-PCs mit begrenzter Prozessorleistung. VP6-S empfiehlt sich für Streaming-Inhalte mit hoher Auflösung, die mit mindestens 500 Kbps auf eben solch langsameren Rechnern bereitgestellt werden.

Weitere Informationen über diesen Codec finden Sie unter www.on2.com.

Dynamisches Streaming

Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 10 oder höher erforderlich)

Mit der neuen Funktion für dynamisches Streaming können Sie die Qualität während der Übertragung überwachen. Änderungen an der Bandbreite werden automatisch erkannt, sodass Sie auch während der Wiedergabe nahtlos zwischen Streams mit unterschiedlicher Bandbreite wechseln können. Dynamisches Streaming unterstützt die Standards H.264 und VP6 und kann in Flash CS4 und Flash Player 10 per ActionScript gesteuert werden.

Sie können den Server beispielsweise so einrichten, dass er bei reduzierter Client-Bandbreite zu einem SD-Stream mit geringerer Bit-Rate wechselt. Bei Verbesserung der Netzwerkbedingungen kann der Server wieder auf HD umschalten. Der Übergang verläuft fließend, sodass auf Client-Seite keine Unterbrechungen oder Störungen auftreten.

Folgende Voraussetzungen ermöglichen ein optimales Anwendererlebnis beim dynamischen Streaming:

- Die verschiedenen Versionen oder Teile des Inhalts müssen synchronisiert sein – die Video-Zeitleisten müssen übereinstimmen.
- Audio- und andere Daten in einem Stream müssen mit den Videodaten im Stream synchron sein.

Der Übergang zwischen zwei aufgezeichneten Sequenzen erfolgt je nach Inhaltstyp auf eine der drei folgenden Arten:

- Reine Video-Streams – Wechsel beim nächsten Keyframe in der Ziel-Zeitleiste
- Gemischte Streams (Video und Audio) – Wechsel beim Audio-Sample direkt vor dem nächsten Keyframe in der Zeitleiste des Ziel-Streams. Die Audio-Zeitleisten von Quell- und Ziel-Stream müssen übereinstimmen, damit keine Störgeräusche entstehen.
- Reine Audio-Streams – Wechsel beim nächstmöglichen Sample

Die Implementierung von Übergängen zwischen Live-Videoinhalten ist etwas komplizierter. Die Streams müssen auf 3 bis 5 Millisekunden synchronisierte Zeitstempel enthalten, damit der Server präzise Übergangspunkte auswählen kann.

Stream-Übergänge erfolgen zwar auf dem Server, doch der Befehl zum Wechsel und die zugehörige Programmlogik stammen von der Client-Anwendung. Mithilfe von ActionScript werden Skripten zur Überwachung der Download- und Wiedergabestatistiken entwickelt, die im Bedarfsfall den Wechsel von einem zum anderen Stream vorgeben.

Adobe empfiehlt dynamisches Streaming für folgende Inhalte:

- Videos mit langer Spieldauer
- Große Videodateien
- HD-Video
- Videos mit größeren Abmessungen wie Videos im Vollbildmodus
- Inhalte für Rechner, die anfälliger für Bandbreitenprobleme sind, z. B. Heim-PCs

Intelligentes Zwischenspeichern

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 9,0,115,0 oder höher erforderlich)

In Flash Player 9,0,115,0 ist die Wiedergabe-Engine erweitert worden, damit der zwischengespeicherte Anteil eines aufgezeichneten Streams erhalten bleibt, wenn Sie die Wiedergabe unterbrechen. Das heißt, der Pufferspeicher von Flash Media Server wird nicht mehr gelöscht, wenn ein Video angehalten wird, und die Wiedergabe kann umgehend fortgesetzt werden, ohne dass eine erneute Pufferung erforderlich ist. Wenn Sie diese Funktion zum Vorabpuffern von Videoinhalten verwenden, schaffen Sie nahtlose Übergänge zwischen den Elementen einer Wiedergabeliste.

Live-Video

Flash Media Server 3.5 umfasst mehrere neue Funktionen für leistungsstarke Live-Streaming-Anwendungen.

DVR-Funktionalität

Funktion in Flash Media Interactive Server (Flash Player 10 erforderlich)

Flash Media Interactive Server 3.5 ermöglicht die Server-seitige Aufzeichnung von Spark- und H.264-Streams, die anschließend wiedergegeben und durch Steuerungen zum Anhalten oder Durchsuchen des Live-Videos und sonstigen benutzerdefinierten Funktionen ergänzt werden können. Die Wiedergabesteuerungen werden in Client- und Server-seitigem ActionScript definiert.

Live und sofort

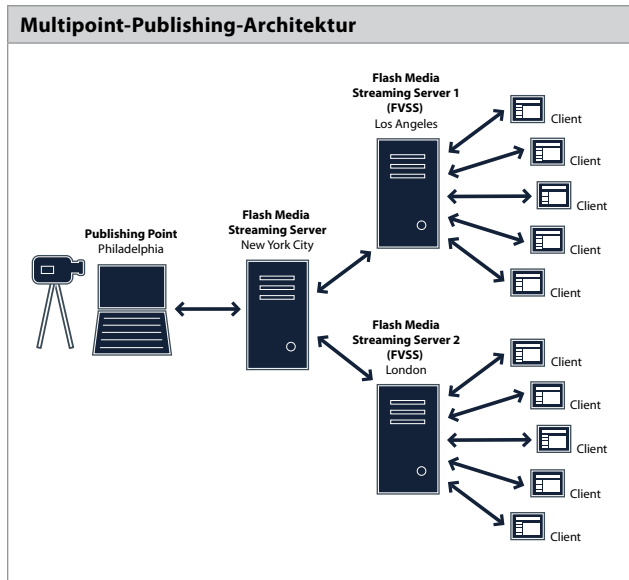
Funktion in Flash Media Interactive Server und Flash Media Streaming Server (Flash Player 6 oder höher erforderlich)

Video-Streams, die mit Flash Media Server bereitgestellt werden, werden sofort abgespielt, sobald die entsprechende Web-Seite geladen oder die Wiedergabe-Schaltfläche gewählt wurde. Dies ist ein entscheidender Vorteil gegenüber anderen Streaming-Technologien wie dem progressiven Download und gilt sowohl für Live- als auch On-Demand-Streams.

Multipoint-Publishing

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Diese leistungsstarke Funktion ist in allen Editionen von Flash Media Server 3.5 verfügbar und erhöht die Flexibilität und Skalierbarkeit von Streaming-Anwendungen. Falls Sie Inhalte bisher über ein CDN (Content Delivery Network) bereitgestellt haben, hatten Sie keine Möglichkeit, eigenen Server-seitigen Code zu implementieren oder Datennachrichten in den ausgehenden Stream einzufügen. Beim Multipoint-Publishing können Sie Ihren eigenen Flash Media-Server (oder Flash Media Live Encoder) verwenden, um den Feed zum CDN zu steuern. Das CDN leitet dann die Inhalte an Ihre Clients weiter (siehe folgende Abbildung). Sogar die kostenlose Entwickler-Edition kann als lokaler Live-Publishing-Point für Streaming-Anwendungen verwendet werden.



Zugriff auf Stream-Daten

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Der Zugriff auf Stream-Daten ermöglicht es Ihnen zu kontrollieren, ob Clients einen Snapshot eines Streaming-Videos als Bitmap erfassen können. Mit dieser Funktion lassen sich beispielsweise dynamische Miniatur- und Videovoransichten erstellen.

Der Zugriff auf Stream-Daten wird im Server-seitigen Code konfiguriert. Dazu werden die Eigenschaften `audioSampleAccess` und `videoSampleAccess` – ähnlich wie `readAccess` und `writeAccess` – für Streams eingerichtet. Flash Media Server ermittelt die Berechtigungen für jeden Stream und sendet eine spezielle Datennachricht an den Player. Da die Funktion Bitmap-Daten verwendet, ist sie erst ab Flash Player 9 verfügbar.

Daten-Keyframes für Live-Video

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Bei der Live-Übertragung von Video stellt die Bereitstellung von Stream-Metadaten an Anwender, die sich mitten im Streaming-Prozess mit dem Server verbinden, eine der größten Herausforderungen dar. Anders als On-Demand-Streams, bei denen Metadaten immer als erstes gestreamt und empfangen werden, sobald der Anwender den Inhalt abonniert, können Live-Streams zu jeder Zeit abonniert werden. Daher kann es passieren, dass Abonnenten, die sich zu spät hinzuschalten, die Metadaten des Live-Streams nie erhalten. Flash Media Server verwendet Daten-Keyframes, um dieses Problem zu beheben, und versendet Metadaten an neue Abonnenten, sobald sie den Stream anfordern.

Bereitstellung für mobile Endgeräte

Flash Media Server 3.5 kann Video im FLV-Format (On VP6/MP3) an mobile Endgeräte streamen, auf denen Flash Lite 3 installiert ist. Flash Lite 3 beinhaltet eine eingeschränkte Implementierung der Klassen `NetConnection` und `NetStream`, sodass nur Video-, Audio- und Metadaten-Pakete empfangen werden können.

Sobald Flash Lite 3 eine Verbindung mit Flash Media Server aufbaut, werden User-Agent-Daten versendet. Anhand dieser Informationen können Sie den Zugriff filtern oder Videodaten bereitstellen, die für das betreffende Gerät optimiert sind. Der von Flash Lite 3 gesendete User-Agent-String enthält folgende Elemente:

- Software-Familie und -Version, z. B. Flash Lite, Version 3.0
- Geräteerkennung, z. B. Motorola RAZR V3x
- Profilkennung – die RDF-URI (Resource Description Framework, Uniform Resource Identifier) für das Gerät
- Netzwerk/Typ, z. B. Code Division Multiple Access (CDMA), 3G o. Ä.

Der entsprechende User-Agent-String könnte folgendermaßen aussehen:

```
FlashLite/3.0.1 Device/RAZRV3x Profile/razrv3x Network/2.5G
```

Geräteinformationen und eine Beispiel-URL für das Handy-Modell Motorola Razor finden Sie unter <http://motorola.handango.com/phoneconfig/razrv3x/Profile/razrv3x.rdf>.

Flash Media Server kann diese Flash Lite 3-Daten auf verschiedene Arten nutzen:

- **Virtuelle Schlüssel** – Mit virtuellen Schlüsseln können Sie Flash Media Server ohne Programmierung für den automatischen Zugriff auf Videos aus einem anderen Ordner konfigurieren. Die Videos in diesem Ordner können beispielsweise für einen bestimmten Client optimiert sein. Bisher wurden auf diese Weise Clients herausgefiltert, die den Codec On2 VP6 nicht unterstützen. Jetzt kann dieses Verfahren auch für Flash Lite verwendet werden.
- **Autorisierungs-Plug-in** – Sie können das Plug-in so konfigurieren, dass eine Anforderung des User-Agents umgeleitet und auf einen optimierten oder alternativen Video-Stream verwiesen wird, falls der angefragte Stream nicht verfügbar ist.
- **Server-seitiges ActionScript** – Mithilfe des Client-Objekts auf dem Server können Sie auf den User-Agent zugreifen, ihn analysieren und einfache Zugriffskontrollen erstellen, die festlegen, welche Geräte auf das Video zugreifen können.

Programmierfunktionen

ActionScript-Klassen für dynamisches Streaming

*Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Flash Media Server 3.5 kann Befehle zum Umschalten zwischen zwei Versionen eines Streams empfangen, die mit verschiedenen Bit-Raten codiert sind. So kann sich die Medienanwendung an neue Netzwerkbedingungen anpassen.

ActionScript 3.0 enthält die Eigenschaft `NetStream.info`, das `NetStreamInfo`-Objekt sowie zugehörige Klassen, mit denen Entwickler Download- und Wiedergabestatistiken überwachen können. Die Methode `NetStream.play2()` und die entsprechende Klasse `NetStreamPlayOptions` ermöglichen den Wechsel zu einem anderen Stream während der Wiedergabe. (Ähnliche APIs stehen auch in ActionScript 2.0 zur Verfügung.) Weitere Informationen finden Sie im *ActionScript 3-Referenzhandbuch für Flash Media Server*.

Integrierte Dienste für Live-Video und VOD

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Im Lieferumfang von Flash Media Streaming Server und Flash Media Interactive Server sind zwei integrierte Anwendungen bzw. Dienste enthalten:

- **VOD** – Der VOD-Dienst ist ein Publishing-Point, von dem aus Sie Video- und Audiodateien auf Ihren Server hochladen und abspielen können, ohne einen eigenen Dienst entwickeln oder den Server konfigurieren zu müssen.
- **Live** – Der Live-Dienst ist ein Publishing-Point, der in Kombination mit Flash Media Live Encoder das Streaming von Live-Video ohne Server-seitige Programmierung oder Konfiguration vereinfacht.

Die integrierten Dienste können sofort zum Streaming mit dem Flash Media Server eingesetzt werden. Sie stehen in allen Versionen von Flash Media Server zur Verfügung. Ausführliche Anweisungen zur Verwendung der Dienste finden Sie im *Installationshandbuch für Flash Media Server*.

AMF3-Unterstützung

Funktion in Flash Media Interactive Server (Flash Player 8 oder höher erforderlich)

Flash Media Server unterstützt den AMF3-basierten Datenaustausch zwischen Server und Client. Flash Media Server kann jetzt alle grundlegenden Datentypen wie Number, Boolean, String, Null, Undefined, Array, Date, XML, Object und ByteArray serialisieren oder deserialisieren.

Hinweis: ByteArray-Daten können mit Server-seitigen Skripten weder erstellt noch geprüft, jedoch sicher zwischen Clients ausgetauscht werden.

Diese Implementierung ist vollständig abwärtskompatibel. Im AMF-Mischmodus können gleichzeitig Verbindungen von AMF3- und AMF0-Clients mit einer Anwendung erfolgen. Sobald eine AMF3-Nachricht gesendet wird, hebt der Server jedoch alle Verbindungen mit Clients auf, die nur AMF0-Daten unterstützen.

- Wenn Clients durchweg vom Typ AMF0 bzw. AMF3 sind, ist ungeachtet der Anwendungscodierung die Kommunikation zwischen Clients möglich.
- Wenn zwei Clients mit unterschiedlicher Codierung sich mit einer myAMF-Anwendung verbinden, können sie miteinander Daten austauschen, sofern die Daten keine neuen ActionScript 3-Datentypen wie ByteArray oder XML oder externalisierte AMF3-Objekte enthalten.
- Wenn der AMF3-Client beispielsweise ein ByteArray versendet, wird die Verbindung des AMF0-Clients beendet, weil AMF0-Clients keine ByteArrays verarbeiten können. (Das Ereignis wird in den Protokolldateien des Servers vermerkt.)

Jedes Mal, wenn der Server Daten für Clients serialisieren muss, geschieht dies zunächst in AMF0. Falls das nicht möglich ist, serialisiert der Server automatisch die Daten in AMF3. Diese Funktion steht in allen grundlegenden Kommunikationsmethoden wie `NetConnection.call`, `Client.call`, `application.broadcastMsg`, `SharedObjects` und `NetStream.send` sowie in Daten zu Verfügung, die in Videodateien eingebettet sind. Eine `NetConnection` zwischen zwei Servern (über Server-seitiges ActionScript) wird standardmäßig in AMF3 umgewandelt.

Die Standardcodierung von Objekten kann auch in der Datei `Application.xml` festgelegt und für jede einzelne `NetConnection` mit der Eigenschaft `NetConnection.objectEncoding` überschrieben werden. Auch wenn in der Datei `Application.xml` beispielsweise AMF3 vorgegeben ist, können Sie eine `NetConnection` als AMF0 definieren, indem Sie die Eigenschaft entsprechend konfigurieren.

Die AMF3-Unterstützung ist ein hochflexibles Tool für den Datenaustausch und eröffnet zahlreiche Möglichkeiten für interaktive Anwendungen. Oft entfällt damit die Integration mit anderen Server-Technologien im Back-End.

Verwaltungs-API

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Mit der Verwaltungs-API erstellen Sie eigene Tools zur Überwachung, Konfiguration und Verwaltung von Flash Media Server 3.5. Die Methoden der API lassen sich mit einer beliebigen Version von Client-seitigem ActionScript über einen Webclient per HTTP oder über Flash Player bzw. einen Adobe AIR-Client per RTMP aufrufen. Mit diesen Verwaltungsmethoden können Sie Administratorkonten hinzufügen oder entfernen, den Server konfigurieren, Speicher freigeben, virtuelle Hosts verwalten, Anwendungen und Streams überwachen u. v. m.

Weitere Informationen zum Einsatz der Verwaltungs-API finden Sie im *Konfigurations- und Administrationshandbuch für Adobe Flash Media Server* und im *Referenzhandbuch für die Verwaltungs-API von Adobe Flash Media Server*.

Implementierung von Flash Media Server 3.5

Die Installation von Flash Media Server 3.5 auf Linux- oder Windows-Systemen ist ganz einfach. Zuerst müssen Sie die Art der Implementierung festlegen. Flash Media Server kann aus einzelnen Installationen, einem Publishing-Point mit Verbindung zu einem CDN oder einer komplexen Edge-Origin-Architektur bestehen. Bei der Entscheidung, welche Struktur Ihren Streaming-Anforderungen am besten gerecht wird, helfen die Informationen und Hinweise in der *Technischen Übersicht über Adobe Flash Media Server*.

Das Installationsprogramm muss in jedem Fall auf allen Rechnern ausgeführt werden, auf denen Flash Media Server implementiert werden soll. Anschließend müssen Sie jeden Server einzeln konfigurieren, Edge- und Origin-Server bestimmen usw. Ausführliche Installationsanweisungen finden Sie im *Installationshandbuch für Adobe Flash Media Server*.

Upgrade von Flash Media Server 3

Alle Flash Media Server 3-Anwendungen sind vollständig kompatibel mit Flash Media Server 3.5. Selbst entwickelte Anwendungen für Flash Media Interactive Server werden jedoch nicht von Flash Media Streaming Server unterstützt.

Legen Sie vor dem Upgrade auf Flash Media Server 3.5 Sicherungskopien der Konfigurationsdateien, Lizenzdateien, Module und Anwendungsordner an.

Plug-ins in C++ (Adapter), die Sie weiterhin verwenden möchten, müssen erneut kompiliert werden.

Ausführliche Upgrade-Anweisungen stehen im *Installationshandbuch für Adobe Flash Media Server*.

Upgrade von Flash Media Server 2

Da alle Macromedia Flash Media Server 2- und Macromedia Flash Communication Server-Anwendungen mit Flash Media Interactive Server 3.5 kompatibel sind, ist die Umstellung ganz einfach. Ein Upgrade von Flash Media Server 2 auf Flash Media Streaming Server 3.5 ist jedoch nicht erhältlich, da es sich um ein neues Produkt handelt. Für Flash Media Streaming Server müssen Sie daher eine Lizenz für die Vollversion erwerben.

Für Flash Media Server 3.5 wird ein anderer Installationsordner als für Flash Media Server 2 angelegt. Unter Windows steht der Standardordner in `Programme\Adobe\Flash Media Server 3.5`. Unter Linux ist es der Ordner `/opt/adobe/fms`.

Die Konfigurationsdateien von Flash Media Server 2 sind *nicht* kompatibel. Erstellen Sie vor dem Upgrade Sicherungskopien von allen Konfigurationsdateien. Die Konfiguration muss manuell auf den neuen Server übertragen werden. Server-seitiges und Client-seitiges ActionScript werden von Flash Media Server 3.5 vollständig unterstützt.

Die Flash Media Server 2-Komponenten für Flash werden auch von der Nachfolgerversion unterstützt. Einige Komponenten sind jedoch nicht mehr verfügbar und werden in künftigen Versionen nicht aktualisiert.

Die Verwaltungskonsole von Flash Media Server 2 („Management Console“) wurde umbenannt und heißt jetzt „Flash Media Administration Console“. Es gibt keine wesentlichen Änderungen; lediglich die Debug-Funktion unterstützt nun auch die Wiedergabe von H.264-Inhalten.

Ausführliche Upgrade-Anweisungen stehen im *Installationshandbuch für Adobe Flash Media Server*.

Installationsprüfung

Nach der Installation von Flash Media Server auf Ihrem Server ist eine Prüfung auf Korrektheit erforderlich. Stellen Sie zunächst mithilfe der Verwaltungskonsole eine Verbindung zum Server her.

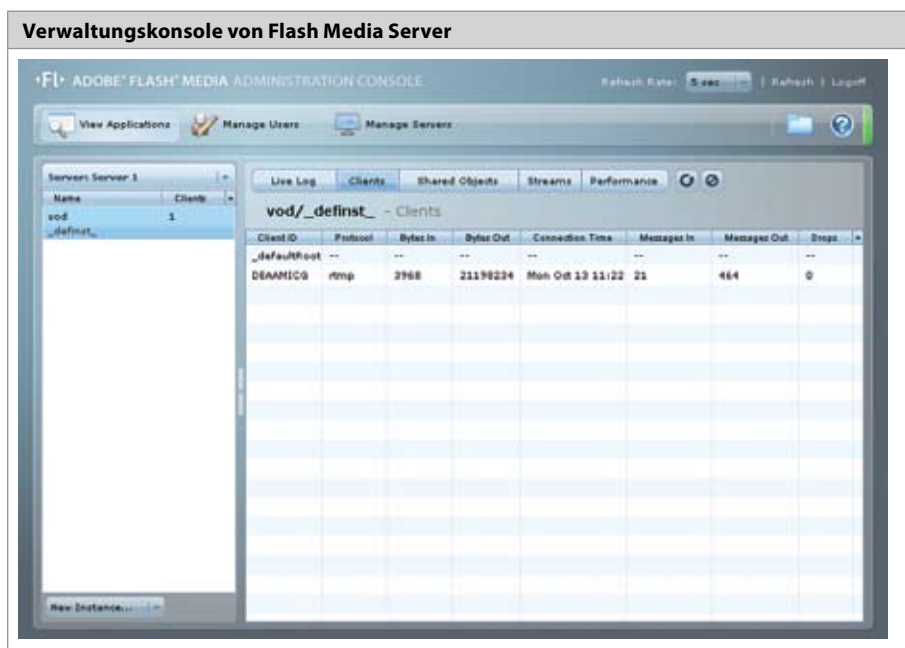
```
Start > Programme > Adobe > Flash Media Server > Administration  
Console (Windows)
```

```
opt/adobe/fms/fms_adminConsole.htm (Linux)
```

In der Konsole können Sie prüfen, welche Anwendungen sich ausführen lassen. Starten Sie versuchsweise die VOD- oder Live-Dienste. Nach der Anmeldung klicken Sie auf „Video Applications“, dann auf „New Instance“ und wählen „VOD“ oder „Live“. Wenn der Server korrekt ausgeführt wird, erscheinen die gestarteten Anwendungen in der Liste im Bedienfeld für die Dienste (siehe Abbildung auf der nächsten Seite).

Alternativ können Sie die Installation mit der VOD-Beispielanwendung unter `samples/applications/vod/vodtest.html` testen. Wählen Sie ein Video (FLV oder H.264) für die Wiedergabe aus.

Im *Installationshandbuch für Adobe Flash Media Server* finden Sie eine Liste der installierten Dateien und der jeweiligen Speicherorte. Falls Flash Media Server sich nicht ohne Weiteres ausführen lässt, können Sie anhand dieser Liste nachprüfen, ob alle Komponenten installiert wurden.



Integrierter Apache HTTP-Server

Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Alle Editionen von Flash Media Server 3.5 enthalten eine spezielle Version des Apache HTTP-Servers. Wenn Sie Apache installieren und aktivieren, können Sie Client-SWF-Dateien, Container-HTML-Dateien und Medieninhalte vom selben Server aus bereitstellen.

Über Client-seitige ActionScript-Routinen lassen sich außerdem Apache-Medieninhalte per progressiven Download mit HTTP bereitstellen, falls Streaming mit RTMP nicht möglich ist. Wenn ein Client zum Beispiel versucht, ein Video per RTMP-Streaming zu empfangen, und keinen Erfolg hat, versucht der Server, eine Tunnelverbindung von RTMP über HTTP herzustellen. Wenn auch dies nicht gelingt, stellt der Server das Video per HTTP bereit.

Bei der Installation von Apache stehen folgende Optionen zur Verfügung:

- Sie installieren Apache, lassen den Server aber deaktiviert (Standardeinstellung bei der Installation).
- Sie installieren Apache und aktivieren die HTTP-Bereitstellung.
- Sie installieren nur die Konfigurations- und Beispieldateien.

Installieren Sie Apache nicht, wenn Sie einen eigenen Apache HTTP-Server einsetzen möchten. Oder beschränken Sie die Installation auf die Konfigurationsdateien und Beispiele.

Flash Media Server startet und stoppt den Webserver automatisch. Wenn Sie mit Apache bereits vertraut sind, können Sie diese Funktion in der Datei `fms.ini` oder `server.xml` deaktivieren und sie selbst verwalten. Falls Sie Apache manuell starten und stoppen oder Apache als Dienst einrichten möchten, starten Sie Apache Flash Media Server, und stoppen Sie den Apache-Dienst nach Flash Media Server.

Weitere Informationen finden Sie im *Installationshandbuch für Adobe Flash Media Server*.

Konfiguration von Adaptern und virtuellen Hosts

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Flash Media Server ist in mehrere hierarchische Ebenen unterteilt: Server, Adapter, virtueller Host und Anwendung. Der Server stellt die höchste Ebene dar und umfasst einen oder mehrere Adapter. Jedem Adapter sind ein oder mehrere virtuelle Hosts zugeordnet. Jeder virtuelle Host verwaltet eine oder mehrere Anwendungen oder Dienste. Sie können Adapter und virtuelle Hosts hinzufügen, um den Server als Host für mehrere Anwendungen und Websites einzurichten.

Wenn Sie auf einem Server mehrere Websites verwalten, verwenden Sie am besten virtuelle Hosts, um Kunden eigene Stammordner zuzuordnen. Auf diese Weise lassen sich die Einstellungen, Inhalte und Protokolldaten für jeden Endanwender separat verwalten.

Einem Adapter können Sie eine IP-Adresse oder Port-Nummer zuweisen, einem virtuellen Host jedoch nicht. Falls ein Anwender beispielsweise ein eigenes SSL-Zertifikat benötigt, ordnen Sie seinen virtuellen Host dem eigenen Adapter zu.

Optimierung der Server-Leistung

Flash Media Server 3.5 kann auf mehreren Ebenen konfiguriert werden:

- Server
- Adapter
- Virtueller Host
- Anwendung

Hinweis: Alle Editionen von Flash Media Server 3 sind vorkonfiguriert und können direkt nach der Installation zum Streamen eingesetzt werden. Da Änderungen an den Einstellungen sich auf die Performance und Zuverlässigkeit der Server-Software auswirken können, sollten Sie dabei sehr sorgfältig vorgehen und vorher alle ursprünglichen XML-Dateien sichern. Die am häufigsten geänderten Einstellungen befinden sich in der Datei `FMS.ini`.

Auf jeder Ebene werden die Konfigurationseinstellungen als XML-Dateien im Verzeichnis `RootInstall/conf` gespeichert, wo auch separate Konfigurationsdateien enthalten sind, die Administratorkonten und Protokolle überwachen.

Die Konfigurationsdateien lassen sich in jedem Text- oder XML-Editor bearbeiten. Damit die Änderungen in Kraft treten, müssen Sie Flash Media Server erneut starten. Bei neuen Einstellungen für Administratorkonten (`Users.xml`) muss auch Flash Media Administration Server neu gestartet werden.

Einstellungen auf Server-Ebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Alle Einstellungen für die Initialisierung des Servers befinden sich in einer Datei: `fms.ini` im Verzeichnis `RootInstall/conf`. Diese Datei enthält wichtige Einstellungen wie den Benutzernamen und das Kennwort des Administrators sowie die Einstellungen, die Sie bei der Installation ausgewählt haben. Es gibt auch nur eine `Server.xml`-Datei, die Angaben wie die Höchstzahl von Verbindungsanforderungen, Einstellungen für die Zeitüberschreitung bei nicht genutzten Verbindungen, IPv6-Einrichtung, SWF-Verifizierung, zulässige Domänen, SSL/RTMPE-Konfiguration und Protokollierung umfasst. Änderungen an der Datei `Server.xml` betreffen den gesamten Server, sofern sie nicht in einer nachfolgenden Konfigurationsdatei überschrieben werden.

Einstellungen auf Adapter-Ebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Die Datei `Adaptor.xml` enthält die Einstellungen für einzelne Netzwerk-Adapter, z. B. die Anzahl der Threads, die vom Adapter verwendet werden können, die Kommunikations-Ports, über die der Adapter Verbindungen herstellen kann, die IP-Adressen und Domänen, von denen der Adapter Verbindungen akzeptiert, und die verfügbaren RTMP-Versionen. Falls jeder Adapter ein anderes digitales Zertifikat verwenden soll, können Sie zusätzlich SSL für die Datei `Adaptor.xml` implementieren.

In `RootInstall/conf` befindet sich für jeden Adapter ein eigenes, nach ihm benanntes Verzeichnis. Jedes Adapterverzeichnis muss die Datei `Adapter.xml` enthalten.

Wenn beispielsweise mit dem Server der Standard-Adapter `_defaultRoot_` installiert wird, wird ein Verzeichnis namens `conf/_defaultRoot_` eingerichtet. Um die Adapter-Einstellungen zu ändern, müssen Sie die Werte in der zugehörigen Datei `Adapter.xml` bearbeiten.

Einstellungen auf Host-Ebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Wenn Sie den Server mit virtuellen Hosts einrichten, lassen sich die zugehörigen Konten auf dem Server für Flash Media Server eindeutig definieren. Jedes Verzeichnis für einen virtuellen Host enthält eine Datei namens `Vhost.xml`, in der die individuellen Einstellungen des virtuellen Hosts festgelegt sind, z. B. Alias, Speicherort des Anwendungsverzeichnisses, maximale Anzahl der verwendbaren Ressourcen usw.

Jeder virtuelle Host muss über ein eigenes Verzeichnis im Adapter-Verzeichnis verfügen. Das Verzeichnis muss den gleichen Namen wie der virtuelle Host haben, z. B. `streaming.adobe.com`. Jeder definierte virtuelle Host muss einem DNS-Eintrag (Domain Name Server) oder einer anderen Namensauflösung zugeordnet sein, wie etwa einer WINS-Adresse (Windows Internet Name Service) oder einer Host-Datei, in der eine IP-Adresse auf dem Server angegeben ist.

Jeder Adapter muss neben den benutzerdefinierten virtuellen Hosts ein Verzeichnis `_defaultVHost_` umfassen. Wenn eine Client-Anwendung versucht, eine Verbindung zu einem nicht vorhandenen virtuellen Host aufzubauen, versucht der Server, den Client mit `_defaultVHost_` zu verbinden. (Falls Sie einen sicheren Port für den Adapter verwenden, zu dem der virtuelle Host gehört, können Sie außer `_defaultVHost_` nur einen weiteren virtuellen Host für den Adapter definieren.)

Einstellungen auf Anwendungsebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Die Datei `Application.xml` enthält die Einstellungen für Flash Media Interactive Server-Anwendungen, z. B. die Größe der Server-seitigen ActionScript-Laufzeit-Engine, den Speicherort für Streams und gemeinsam genutzte Objekte, die AMF-Standardcodierung und Einschränkungen bezüglich der Bandbreite.

Diese Datei im Verzeichnis des virtuellen Hosts konfiguriert die Standardeinstellungen für alle Anwendungen innerhalb dieses virtuellen Hosts. Falls für eine bestimmte Anwendung andere Einstellungen verwendet werden sollen, können Sie die Datei `Application.xml` in das registrierte Anwendungsverzeichnis der Anwendung kopieren (`/applications/app_name`) und eigene Einstellungen festlegen.

In den meisten Fällen überschreiben die Einstellungen in der Datei `Application.xml` im Anwendungsverzeichnis die Einstellungen in der Datei `Application.xml` im Verzeichnis des virtuellen Hosts. Weitere Informationen erhalten Sie im *Konfigurations- und Administrationshandbuch für Adobe Flash Media Server*.

Einstellungen auf Anwenderebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Um Berechtigungen für Flash Media Server-Administratorkonten hinzuzufügen, zu entfernen oder einzurichten, müssen Sie die Datei `Users.xml` auf Stammebene des Konfigurationsverzeichnisses bearbeiten. Zudem lassen sich innerhalb dieser Datei Aufrufe der Server-Management-API an Flash Media Administration Server konfigurieren, um beispielsweise den Zugriff auf bestimmte HTTP-Aufrufe zu genehmigen oder abzulehnen.

Einstellungen auf Protokollebene

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Flash Media Server 3.5 bietet leistungsstarke Protokollfunktionen, die sich variabel anpassen lassen.

Die Datei `Logger.xml` befindet sich ebenfalls auf der Stammebene des Konfigurationsverzeichnisses und enthält die Einstellungen für Flash Media Server-Protokolle. Hier können Sie festlegen, welche Daten protokolliert, wo Protokolldateien gespeichert und wie oft diese Dateien archiviert werden, damit neue Daten in die leere Datei geschrieben werden können. Standardmäßig werden die Protokolldateien im Installationsverzeichnis im Protokollordner abgelegt (`RootInstall/logs`).

Der Abschnitt „Logging“ in der Datei `Server.xml` aktiviert oder deaktiviert die Protokolldateien. `Logger.xml` enthält die Einstellungen für Protokolle.

Hinweis: Protokolldateien und Feldbezeichnungen innerhalb dieser Dateien sind auf Englisch. Je nach Betriebssystem können jedoch auch Inhalte in einer anderen Sprache erfasst werden.

Performance-Funktionen

Trotz der zahlreichen integrierten Optimierungen in Flash Media Server 3.5 können Sie die Installation zugunsten maximaler Effizienz individuell anpassen. Einige Funktionen, die sich positiv auf die Effizienz auswirken, werden hier erläutert. Ausführliche Informationen finden Sie im *Konfigurations- und Administrationshandbuch für Adobe Flash Media Server*.

Hinweis: Tipps zur Optimierung von Flash Media Server für Linux-Installationen finden Sie im englischsprachigen Artikel „Performance-tuning Flash Media Server 2 for live webcasts using Linux“ unter www.adobe.com/devnet/flashmediaserver/articles/performance_tuning_webcasts.html.

Streaming-Cache/Streaming-Datenpakete

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Bei der Anforderung eines Streams von einem Server werden Segmente des Streams in einem Cache auf dem Server gespeichert. Sobald der Cache voll ist, entfernt der Server nicht verwendete Segmente beginnend mit dem zuletzt am wenigsten genutzten Segment. Die Größe des Cache-Speichers richten Sie in der Datei `fms.ini` des Servers über den Parameter `SERVER.FLVCACHE_MAXSIZE` entsprechend ein. Der Standardwert beträgt 500 MB.

Streaming-Inhalte werden beim Versenden über RTMP in Datenpakete aufgeteilt. Die Größe eines Datenpakets richten Sie ebenfalls in der Datei `fms.ini` ein. Für den Parameter `APP.DEFAULT_CHUNKSIZE` sind alle Werte zwischen 128 (Standardwert) und 1024 KB zulässig. Für den VOD-Dienst kann die Paketgröße optional separat definiert werden. Hierzu bearbeiten Sie den Parameter `APP.VOD_CHUNKSIZE`. Höhere Werte verringern zwar die CPU-Auslastung, können jedoch gleichzeitig die Performance von Clients beeinträchtigen, die nur über eine geringe Bandbreite verfügen.

Zu hohe Werte für Cache und Datenpakete können sich auf die Performance auswirken. Falls die Cache-Größe zum Beispiel den verfügbaren Speicher oder der Server-Prozess den Grenzwert des Betriebssystems von 2 GB übersteigt, könnte dies zum Abbruch des Server-Prozesses führen. Werden zu kleine Werte gewählt, könnten theoretisch alle Segmente verwendet und daher nicht gegen neue Stream-Segmente ausgetauscht werden. In dem Fall wird die Wiedergabe des Streams, der die neuen Segmente anfordert, gestoppt.

Prozessaufgliederung

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Interactive Server kann in verschiedene Prozesse aufgeteilt werden, um die Server-Kapazitäten effektiver zu nutzen. Dabei werden die plattformtypischen Höchstwerte für CPU, RAM und Prozessisolation umgangen. Die Prozessaufteilung bietet folgende Vorteile:

- Flash Media Server kann Verbindungen schneller annehmen.
- Flash Media Server kann mehr FLV- und MP3-Daten im RAM speichern.
- Das Speicher-Limit von 2 GB wird erhöht.
- Instanzen, Anwendungen, virtuelle Hosts oder Adapter werden bei fehlerhaften Skripten oder DOS-Angriffen (Denial of Service) isoliert.

Weitere Informationen zur Prozessaufgliederung erhalten Sie im *Leitfaden zur Verwaltung von Flash Media Server*.

Zur weiteren Optimierung von Kapazitätsauslastung und Qualität bietet Flash Media Interactive Server die Möglichkeit des verteilten Rechnens. Mehr dazu erfahren Sie im nächsten Abschnitt.

Flash Media Server kann automatisch FMSCore-Prozesse auslösen. Zu diesem Zweck konfigurieren Sie den Knoten für den Prozessumfang in der globalen Datei `Application.xml`. Definiert wird der Umfang für `adaptor`, `vhost`, `app` und `inst`.

Je nach gewähltem Umfang können Core-Prozesse separat konfiguriert werden. Falls Sie beispielsweise das System so konfigurieren, dass Adapter auf verschiedene Core-Prozesse verteilt werden, dienen alle Einstellungen in der zugehörigen Datei `adaptor.xml` und alle nachfolgenden XML-Dateien zur Konfiguration dieses Core-Prozesses. In dem Fall können Sie jedem Core-Prozess einen anderen Port zuweisen, wo er auf Ereignisse warten soll, oder die Tunneleinstellungen für SSL oder HTTP ändern.

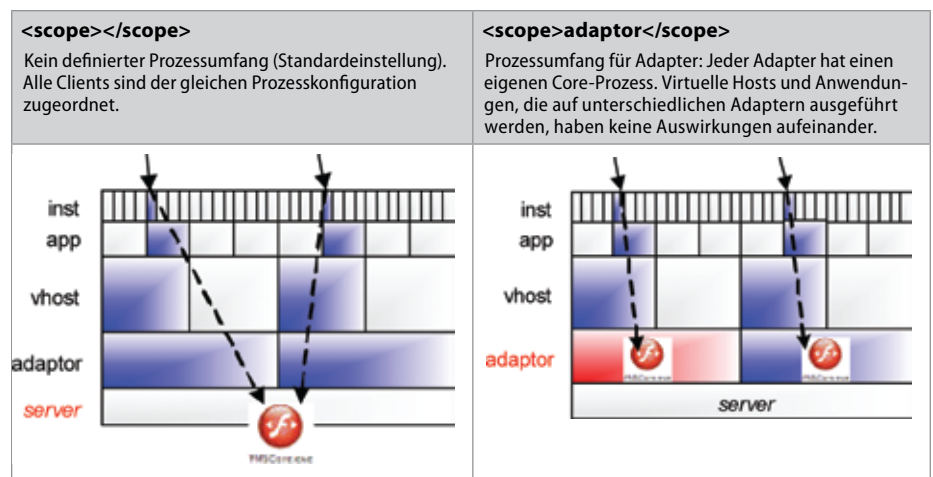
Wenn Sie andererseits die Prozesse auf Anwendungsbasis aufteilen, können Sie für jeden Core-Prozess Bandbreiten-Limits angeben, sofern für jede ausgeführte Anwendung eine separate `application.xml`-Datei vorhanden ist.

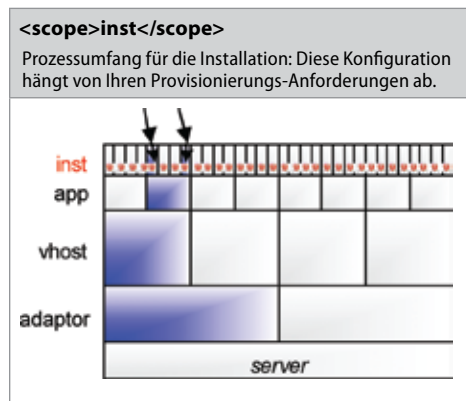
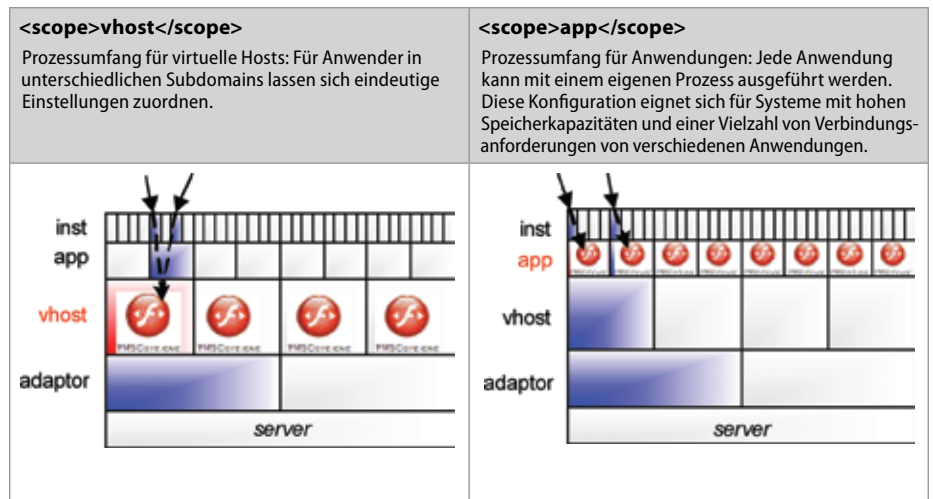
Dies ist die standardmäßige Prozesskonfiguration:

```
<Process>
  <Scope></Scope>
  <Distribute numprocs="1"></Distribute>
  <LifeTime>
  <RollOver></RollOver>
  <MaxCores></MaxCores>
</LifeTime>
  <MaxFailures>2</MaxFailures>
  <RecoveryTime>300</RecoveryTime>
</Process>
```

Die folgenden Abbildungen veranschaulichen die Auswirkungen unterschiedlicher Prozesskonfigurationen auf das System.

Diese Einstellungen eignen sich optimal für zustandsbetonte Anwendungen, bei denen Clients miteinander kommunizieren müssen (Chats, Live-Video, Spiele oder Datenaustausch).





Für einen CDN-Kunden (VOD-Profil) hängt die Auswahl der richtigen Prozesskonfiguration von den Provisionierungs-Richtlinien für Kundenkonten ab. Bedenken Sie, dass Flash Media Server Kunden wie Anwendungen behandelt. In einem System mit vielen Kunden ist Prozessaufgliederung für Anwendungen nicht unbedingt die beste Wahl. Für jeden Prozessumfang sind maximal 4 GB RAM verfügbar, d. h., zu viele Prozesse könnten sich nachteilig auswirken. Wenn Sie andererseits nur wenige Kunden haben, für die Sie größere Mengen von Inhalten bereitstellen, dann ist Aufgliederung für Anwendungen ideal.

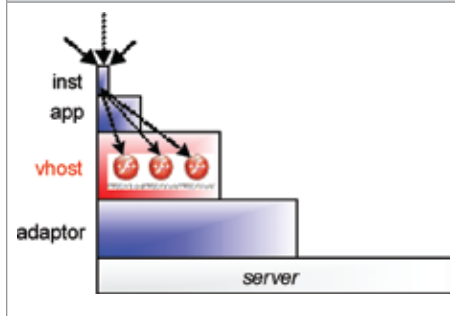
Verteilte Core-Prozesse

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Zur weiteren Optimierung der Leistung und Stabilität des Servers lassen sich Verbindungen für einen bestimmten Prozessumfang auf mehrere Prozesse aufteilen. Wenn Sie beispielsweise den Prozessumfang „Adapter“ wählen, können sich Verbindungen für jeden virtuellen Host über beliebig viele Core-Prozesse verteilen. In der folgenden Abbildung sind jedem virtuellen Host vier Core-Prozesse zugeordnet. Pro Host stehen drei Prozesse zur Verfügung, auf die die Verbindungen verteilt werden können. Bei zwei virtuellen Hosts verteilen sich die Verbindungen entsprechend auf sechs Core-Prozesse.

Verteilte Core-Prozesse

Drei Prozesse können jeweils Verbindungen akzeptieren. Mit derselben Anwendung verbundene Anwender haben u. U. keinen Zugriff auf den RAM-Speicher, wenn sie mit einem anderen Prozess verbunden sind. Geeignet für zustandslose Implementierungen.



Die Funktion zur Verteilung der Core-Prozesse kann genauso wie die Definition des Prozessumfangs die Leistung des Servers erhöhen. Beim verteilten Rechnen wird mehr RAM für das Caching und mehr Threading für prozessintensive Verbindungen verfügbar. Das Verfahren eignet sich am besten für VOD-Lösungen (z. B. Mediennetze oder Content-Provider). Bei einer hybriden Multiway-Anwendung oder Live-Lösung ist dieses System nicht empfehlenswert, weil für den Datenaustausch Verbindungen demselben Core-Prozess zugeordnet sein müssen.

Entsprechend den Standardeinstellungen in der Konfigurationsdatei `Application.xml` sind verteilte Prozesse deaktiviert.

```
<Process>
  <Scope></Scope>
  <Distribute numprocs="0"></Distribute>
  <LifeTime>
    <RollOver></RollOver>
    <MaxCores></MaxCores>
  </LifeTime>
</Process>
```

In `<Scope>` können Verbindungen auf Prozesskonfigurationen aufgeteilt werden. Die folgende Abbildung veranschaulicht die verfügbaren Optionen. Links stehen die möglichen Werte für `<Scope>`, oben die Werte für `<Distribute>`. Für `<Scope>vhost</Scope>` zum Beispiel stehen als `<Distribute>`-Optionen „app“, „inst“ und „clients“ zur Auswahl.

Optionen für verteilte Prozesse

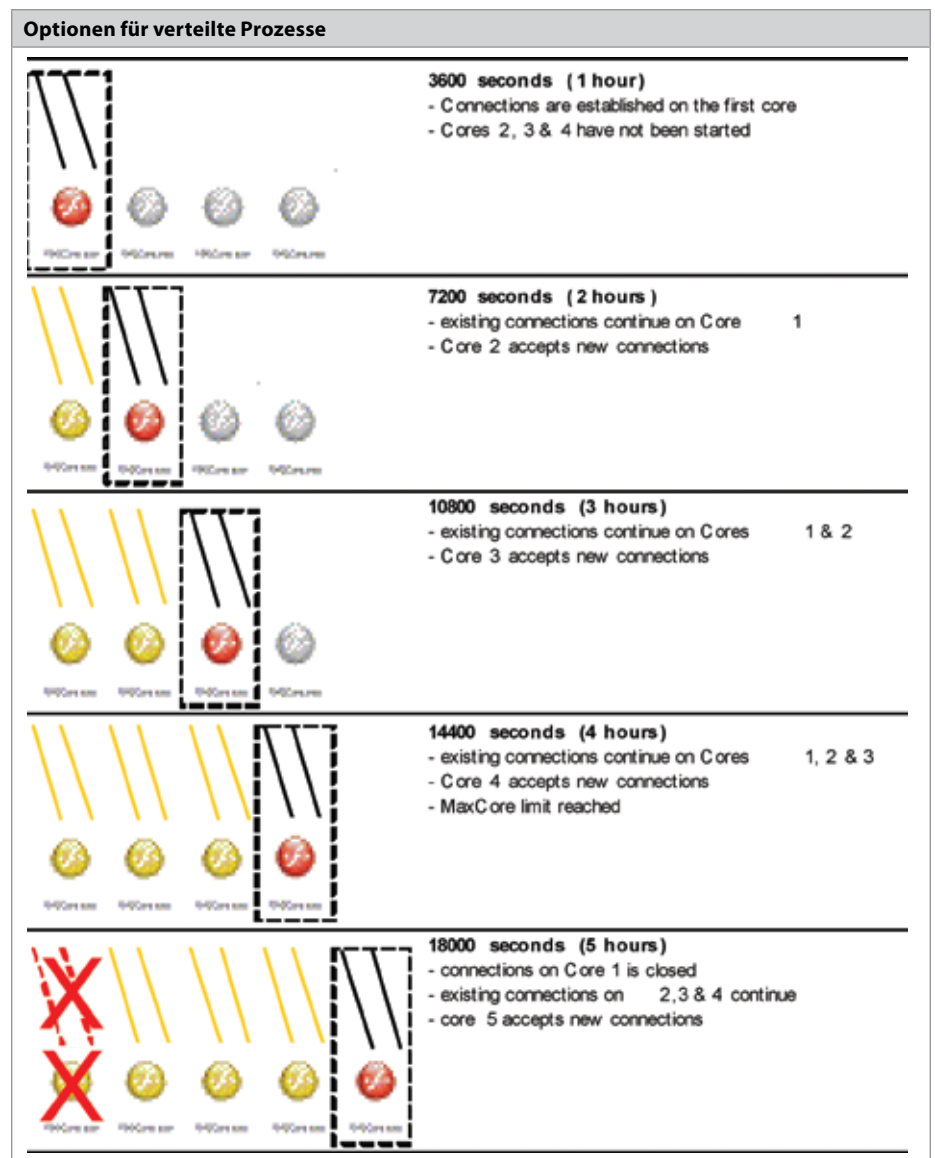
	Distributed Processes				
	adaptor	vhost	app	inst	clients
server	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
adaptor		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
vhost			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
app				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
inst					<input checked="" type="checkbox"/>

Im folgenden Beispiel sind Prozesse für einen virtuellen Host aufgliedert. Die Konfiguration gilt für einen Zeitraum von fünf Stunden mit einen Rollover-Intervall von 1 Stunde (3600 Sekunden).

```

<Process>
  <Scope>adaptor</Scope>
  <Distribute numprocs="1">vhost</Distribute>
  <LifeTime>
    <RollOver>3600</RollOver>
    <MaxCores>4</MaxCores>
  </LifeTime>
</Process>

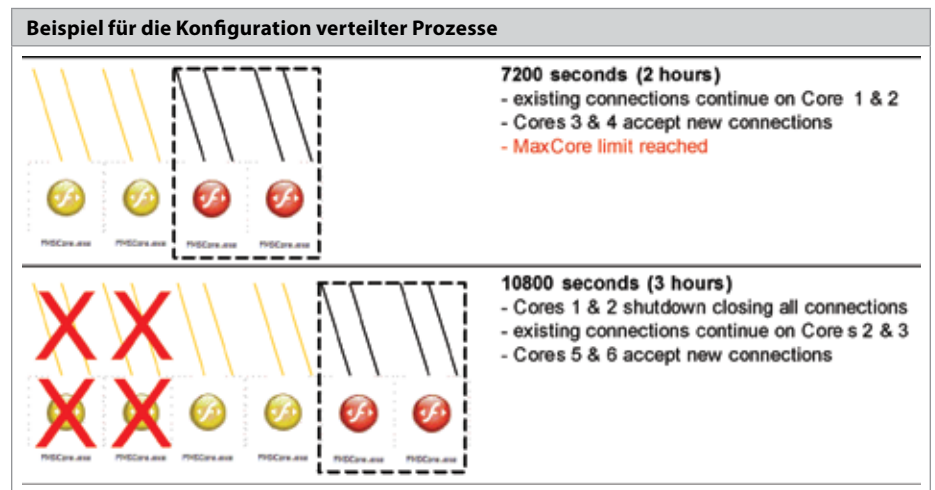
```



Die maximale Verbindungszeit in der oben genannten Konfiguration beträgt vier Stunden. Danach wird der Core-Prozess geschlossen, und alle Verbindungen werden getrennt. An diesem Punkt initiiert die SWF-Client-Datei eine Routine zur Wiederaufnahme der Verbindung und damit der Stream-Wiedergabe.

Mit der Einstellung `numprocs` legen Sie die Anzahl der Prozesse fest, auf die Verbindungen verteilt werden. Wenn Sie für `numprocs` 2 angeben und ansonsten alle Einstellungen beibehalten, verringert sich die maximale Verbindungszeit von vier auf zwei Stunden. Nach der zweiten Stunde ist der `MaxCores`-Wert von 4 erreicht. Damit die nächsten zwei Prozesse ausgeführt werden können, werden die ersten beiden beendet, und die Verbindungen werden geschlossen.

```
<Process>
  <Scope>adaptor</Scope>
  <Distribute numprocs="2">vhost</Distribute>
  <LifeTime>
    <Rollover>3600</Rollover>
    <MaxCores>4</MaxCores>
  </LifeTime>
</Process>
```



Die maximale Verbindungsdauer in dieser Konfiguration beträgt zwei Stunden – für `MaxCore` ist 4, für `numProcs` 2 vorgegeben.

Mit folgender Formel lässt sich die maximale Verbindungszeit für Clients berechnen:

$$(\text{MaxCore DIV NumProcs}) \times \text{Rollover}$$

Beispiel: $(4 \text{ DIV } 2) \times 3600 = 7200 \text{ Sekunden (2 Stunden)}$

Die Gesamtanzahl der Prozesse lässt sich anhand dieser Formel ermitteln:

$$\text{Anzahl Konfigurationen} \times \text{NumProcs} \times \text{MaxCores}$$

Beispiel mit `<Distribute>adaptor</Distribute>` und zwei Adaptern:

$$2 \times 2 \times 4 = 16 \text{ Core-Prozesse}$$

Zur Berechnung des erforderlichen RAMs für die Unterstützung der verteilten Prozesse multiplizieren Sie die Gesamtanzahl der Prozesse mit 4. Jeder Core-Prozess kann bis zu 4 GB RAM in Anspruch nehmen. Das vorherige Beispiel mit 16 Core-Prozessen würde somit bis zu 64 GB RAM belegen.

Schließen nicht genutzter Verbindungen

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Wenn Clients eine Anwendung verlassen, bleibt die Verbindung oft weiterhin offen. Flash Media Server 3.5 kann diese ungenutzten Verbindungen ermitteln und deren Ressourcen für neue, aktive Clients beanspruchen. Wenn ein Client die Verbindung länger als die maximal zulässige Zeit (Standardwert 10 Minuten) nicht genutzt hat, wird die Verbindung vom Server geschlossen. Zur Aktivierung dieser Funktion muss in der Datei `Server.xml` der Parameter `AutoCloseIdle-Clients` auf „true“ gesetzt werden. Anschließend können Sie die Option für einzelne virtuelle Hosts oder Anwendungen in den Dateien `Vhost.xml` und `Application.xml` deaktivieren.

Einschränkung der Verbindungsanforderungen

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Wenn viele Clients versuchen, eine Verbindung zum Server aufzubauen, kann sich dies negativ auf die Qualität bestehender Client-Verbindungen auswirken. Mit der Option `MaxConnectionRate` in der Datei `Server.xml` können Sie die Anzahl der zulässigen Verbindungsanforderungen pro Sekunde begrenzen.

Versenden gebündelter Nachrichten

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Gebündelte Nachrichten werden standardmäßig von Anwendungen aufgegliedert, bevor sie an Clients senden. In der Datei `Application.xml` können Sie anhand des Parameters `AggregateMessages` festlegen, ob Anwendungen gebündelte Nachrichten versenden sollen.

Die Übermittlung gebündelter Nachrichten reduziert die Beanspruchung der CPU und erhöht die Server-Performance. Da jedoch Latenz auftreten kann, ist das Verfahren für die 1:1-Kommunikation in Echtzeit nicht empfehlenswert. Gebündelte Nachrichten eignen sich am besten für On-Demand- und Live-Anwendungen, bei denen Latenz eine geringe Rolle spielt.

Konfiguration des Speichers

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Die Konfiguration des Speichers ist unumgänglich, wenn Sie optimale Server-Leistung erzielen möchten. Falls die Zugriffsgeschwindigkeit der Festplatte nicht mit der Bit-Rate der Inhalte Schritt halten kann, empfangen die Clients Benachrichtigungen über leere Puffer, und die allgemeine Qualität sinkt.

Der Server kann zur Bereitstellung von Medieninhalten lokalen oder Netzwerkspeicherplatz nutzen. In der Datei `Application.xml` können Sie den Standard-Speicherort für Streams und gemeinsame Objekte ändern und virtuelle Verzeichnisse physischen Verzeichnissen zuordnen. Diese Vorgehensweise eignet sich für Gruppenprojekte, bei denen Inhaltsersteller nicht immer direkten Zugriff auf Ihren Flash Media-Server haben sollen, oder für Anbieter mit umfangreichen Medienbibliotheken, die nicht in Flash Media Server kopiert werden sollen.

Sicherheitsfunktionen

Flash Media Server 3.5 verfügt über mehrere Sicherheitsfunktionen, die Sie in den Konfigurationsdateien Ihren Anforderungen gemäß einrichten können. Weitere Informationen über die Schutzfunktionen auf Server- und Inhaltsebene finden Sie im Abschnitt „Sicherheit“.

SWF-Verifizierung

*Erweiterte Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Der Server lässt sich so konfigurieren, dass er SWF-Client-Dateien überprüft, bevor er die Verbindung mit einer Anwendung zulässt. Auf diese Weise verhindern Sie, dass von Dritten definierte SWF-Dateien Ihre Inhalte streamen oder Ihre Server-Ressourcen nutzen. Der Server vergleicht die betreffende SWF-Datei mit den vorhandenen Dateien auf Ihrem Flash Media-Server und erlaubt die Verbindung nur, wenn er eine Übereinstimmung findet.

In der Datei `Application.xml` geben Sie unter `<SWFVerification>` einen oder mehrere Server-Ordner an, in denen SWF-Verifizierungsdateien lokal gespeichert werden. Die Dateien können auch prüfen, wie lange die Verifizierungsdaten im Cache gespeichert werden und ob Ausnahmen (z. B. Flash Media Live Encoder) gelten. Zusätzlich können Sie festlegen, wie oft der Server nach aktualisierten SWF-Prüfdateien sucht. Nachfolgend sehen Sie ein Beispiel für die Datei `Application.xml`. `SWFVerification` ist standardmäßig deaktiviert.

```
<SWFVerification enabled="true">
  <SWFFolder />
  <MinGoodVersion />
  <UserAgentExceptions>
    <Exception from="" to="" />
  </UserAgentExceptions>
  <Cache>
    <TTL>1440</TTL>
    <UpdateInterval>5</UpdateInterval>
  </Cache>
</SWFVerification>
```

Bei Flash Media Interactive Server 3.5 können Sie nun mithilfe des Datei-Plug-ins SWF-Dateien prüfen, die extern gespeichert sind. Damit wird die Flexibilität und Effizienz bei der Bereitstellung sicherer Inhalte erhöht. Weitere Informationen finden Sie im *Entwicklerhandbuch für Adobe Flash Media Interactive Server-Plug-ins*.

Verbindungen zwischen Domänen und virtuellen Hosts

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Sie können eine Liste mit Domänen erstellen, die eine Verbindung mit einem bestimmten virtuellen Host aufbauen dürfen. Standardmäßig sind Verbindungen von allen Domänen zulässig. Geben Sie für den Parameter `VHOST.ALLOW` in der Datei `fms.ini` eine kommagetrennte Liste der Host- und Domänennamen sowie vollständige oder Teile von IP-Adressen an, die auf den virtuellen Host zugreifen dürfen.

Eingeschränkter Zugriff auf Flash Media Administration Server

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Grundsätzlich kann ein Client sich von jeder beliebigen Domäne oder IP-Adresse aus mit Flash Media Administration Server verbinden. Dies kann u. U. die Sicherheit Ihrer Inhalte gefährden. Wenn Sie die standardmäßig zulässigen Verbindungen sperren möchten, ändern Sie den Parameter `AdminServer` in der Datei `Server.xml`. Fügen Sie einfach eine durch Kommata getrennte Liste aller Host- und Domänennamen sowie vollständige oder Teile von IP-Adressen hinzu, deren Verbindungen Sie zulassen möchten. Der Standardwert ist „all“.

Verwaltungs-API über HTTP

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Flash Media Server lässt sich mit einfachen HTTP-Befehlen überwachen. Die Methoden der Flash Media-Verwaltungs-API können über HTTP aufgerufen werden. Richten Sie für den Parameter `USERS.HTTPCOMMAND _ ALLOW` in der Datei `fms.ini` eine kommagetrennte Liste der APIs ein. Zusätzlich können Sie in der Datei `Users.xml` Methoden für die Überwachung auf Anwenderbene angeben.

Die meisten Anwender nutzen diese Funktion zur Erstellung eigener Überwachungsanwendungen. Diese zeigen zum Beispiel Informationen zur Server-Kapazität wie die Anzahl der Verbindungen, die genutzte Bandbreite und die Anzahl der Streams im Cache an. Über die API lässt sich der Server auch neu starten.

Verschlüsseltes RTMP (RTMPE)

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Verschlüsseltes RTMP (RTMPE) ist standardmäßig in der `Adaptor.xml`-Datei des Servers aktiviert und bietet die Verschlüsselung mit 128 Bit. Wenn Sie diese Einstellung ändern möchten, setzen Sie den Parameter `ADAPTOR.RTMPE_ENABLED` auf „off“.

RTMPE sollte nur dann deaktiviert werden, wenn Entwickler, die Anwendungen auf Ihrem Server implementieren, nicht darauf zugreifen sollen, denn RTMPE beansprucht die CPU stärker als das standardmäßige RTMP. Andernfalls lassen Sie das Protokoll aktiviert, damit es im Bedarfsfall verfügbar ist.

Secure Sockets Layer (SSL)

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Wie RTMPE ist auch SSL ein Protokoll für sichere Kommunikation. Für SSL ist jedoch ein Zertifikat erforderlich, das von einer Zertifizierungsinstanz ausgestellt wurde. SSL wird zunächst in der Datei `Server.xml` konfiguriert. Bei Bedarf können Sie Zertifikate für sichere unabhängige Adapter oder virtuelle Hosts einrichten.

Weitere Informationen über die Sicherheitseinstellungen für Flash Media Server finden Sie unter „Datenschutz mit Flash Media Server 3.5“.

Allgemeine Einstellungen

Zur Optimierung und Anpassung Ihrer Anwendung stehen zahlreiche Optionen zur Verfügung.

Debug-Verbindungen

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Die Verwaltungskonsole und -API von Flash Media Server können sich mit SharedObjects und Streams verbinden und auf zugehörige Ereignisse warten. Diese Funktion erleichtert das Debuggen komplexer Server-Anwendungen. Standardmäßig lässt der Server keine Debug-Verbindungen zu. Die Verwaltungskonsole von Flash Media Server muss jedoch eine solche Verbindung zum Server herstellen, um Streams wiedergeben und auf gemeinsame Objekt-daten zugreifen zu können.

Um Debug-Verbindungen zuzulassen, ändern Sie die Einstellungen in der Datei `Application.xml` des betreffenden virtuellen Hosts oder der betreffenden Anwendung. Richten Sie sich dabei nach dem nachfolgend abgebildeten Skript. Anschließend müssen Sie Flash Media Server und den Verwaltungsdienst von Flash Media Server neu starten und die Verwaltungskonsole erneut laden.

```
<Debug>
  <MaxPendingDebugConnections>50</MaxPendingDebugConnections>
  <AllowDebugDefault>true</AllowDebugDefault>
</Debug>
```

Alternativ lassen sich Server-Aktivitäten per HTTP überwachen.

Eigenschaften von Anwendungsobjekten

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Mit Flash Media Server 3.5 lassen sich globale Eigenschaften definieren, auf die alle Anwendungsinstanzen auf einem bestimmten virtuellen Host zugreifen können. Wenn Sie diese Anwendungseigenschaften in einem `Application.xml`-Ordner in einem bestimmten Anwendungsordner definieren, gelten sie ausschließlich für diese Anwendung.

Falls Sie etwa die globale Eigenschaft `application _ owner` erstellen, kann jede Instanz dieser Anwendung mit der folgenden Syntax darauf zugreifen:

```
Application.config.application _ owner
```

Native Bandbreitenerkennung

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Server 3.5 kann die Bandbreite eines Clients nativ oder mit Server-seitigem ActionScript ermitteln. Da die native Bandbreitenerkennung im Kern-Code der Server-Software integriert ist, wird sie viel schneller ausgeführt als die skriptbasierte Bandbreitenerkennung. Die native Funktion ist standardmäßig aktiviert und kann in der Datei `Application.xml` konfiguriert werden.

Aktivitätsprotokolle

Erweiterte Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Server 3.5 bietet Optionen zur Überwachung von Server-Aktivitäten in Echtzeit sowie umfassende Protokollfunktionen, die die Server-Verwaltung und Fehlerbehebung erleichtern. Die Protokolldateien erfassen Aktivitäten wie die allgemeine Netzwerk- und Server-Auslastung, Zugriffe auf den Server, Client-Verhalten und -Interaktionen sowie allgemeine Diagnosedaten.

Flash Media Server führt mehrere Protokolle, die folgende Daten erfassen:

- Zugriffsprotokoll – Client-Zugriffe auf den Server
- Anwendungsprotokoll – Aktivitäten in Anwendungsinstanzen
- Diagnoseprotokoll – Vorgänge auf Server-Ebene
- HTTP-Protokoll – Zugriffe und Fehler auf dem Apache HTTP-Server

Die Zugriffsprotokolle von Flash Media Server werden im W3C-Format gespeichert und können mit Standard-Parsern analysiert werden.

Zugriffsprotokolle

Im Zugriffsprotokoll werden Informationen über die Verbindungsanforderungen der Anwendungsinstanzen von Flash Player und Flash Media Server erfasst. Die Standardkonfiguration erstellt pro Server ein Zugriffsprotokoll (`access.XX.log`), das im Flash Media Server-Protokollverzeichnis gespeichert wird. Flash Media Server kann optional für jeden virtuellen Host eine separate Protokolldatei anlegen. „XX“ entspricht der zweistelligen Versionsnummer des Protokolls – die aktuellsten Daten stehen im Protokoll `access.00.log`.

Anhand des Zugriffsprotokolls können Sie nachvollziehen,

- an welchem Tag und um wie viel Uhr ein Client sich mit dem Server verbunden hat
- welche Bandbreite insgesamt während der Sitzung beansprucht wurde
- auf welche Streams während der Verbindung zugegriffen wurde
- ob der Client einen Stream veröffentlicht hat
- ob der Client innerhalb eines aufgezeichneten Streams vor- und/oder zurückgesprungen ist

Anwendungsprotokolle

Das Anwendungsprotokoll verfolgt die Aktivitäten in Anwendungsinstanzen und ist somit eine enorme Hilfe beim Debuggen von Anwendungen.

Standardmäßig wird ein Protokoll namens `application.XX.log` pro Anwendungsinstanz erstellt. Diese Datei befindet sich im Anwendungs- oder Instanzordner des entsprechenden virtuellen Hosts.

Anwendungsprotokolle zeichnen folgende Daten auf:

- Datum und Uhrzeit des Ereignisses
- Server-Prozesskennung des Ereignisses
- Status des Ereignisses (Warnung, Fehler, Information, Debug usw.)

Diagnoseprotokolle

In den Diagnoseprotokollen werden Informationen über Flash Media Server-Vorgänge erfasst. Sie werden primär zur Behebung von Problemen auf dem Server verwendet.

Flash Media Server ist standardmäßig so konfiguriert, dass für jede Prozessart ein Diagnoseprotokoll erstellt wird. Die Standardprotokolle heißen `master.XX.log`, `edge.XX.log`, `core.XX.log`, `admin.XX.log` und `httpcache.XX.log`. Alle Dateien werden im Protokollverzeichnis von Flash Media Server abgelegt.

Bei einer genauen Prüfung der Diagnoseprotokolle erhalten Sie viele nützliche Daten, z. B. über Streaming-Ereignisse, Anwendungsinstanzen, virtuelle Hosts oder Probleme auf Edge- und Origin-Servern.

HTTP-Protokolle

Aktivitäten des integrierten Apache HTTP-Servers werden in zwei Protokollen erfasst. `httperror.xx.log` verfolgt die Fehler und `httpaccess.xx.log` die Zugriffe auf den HTTP-Server.

Server-seitige Werkzeuge

Load Simulator

Neue Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Der Load Simulator ist ein neues, plattformübergreifendes Standalone-Werkzeug, das eine künstliche Streaming-Last auf einem Flash Media Server erzeugt. Anhand dieser Simulation können Sie Ihre Konfiguration und Implementierung von Flash Media Server testen. Sie erhalten Messwerte für die Anwendung, die Sie für Benchmark-Tests nutzen können. Das Tool umfasst eine einfache grafische Benutzeroberfläche sowie befehlszeilenbasierte Steuerungen für erfahrenere Anwender.

Das Simulationswerkzeug unterstützt dieselben Plattformen wie Flash Media Server: Windows 2003, Windows 2008, Windows XP sowie Linux RH4 und 5. Da Flash Development Server nur maximal 10 gleichzeitige Verbindungen zulässt, ist dieses Tool für den Einsatz mit autorisierten Flash Media-Servern gedacht.

Der Load Simulator steht als kostenloser Download auf der Adobe-Website zu Verfügung.

Verwaltungs-API

Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Mit der Verwaltungs-API kann der Server über Flash Player oder einen Adobe AIR-Client mittels RTMP oder von einem Web-Client mittels HTTP überwacht, verwaltet und konfiguriert werden. Die Verwaltungskonsolle von Flash Media Server wurde auf Basis dieser API entwickelt. Mit der Verwaltungs-API können Sie auch eigene Administrationswerkzeuge erstellen und z. B.

- Administratoren hinzufügen oder löschen
- Server, virtuelle Hosts und Anwendungen starten und stoppen
- Speicherbereinigungen initiieren
- die Server-Konfiguration abrufen und einrichten

Die API wird ausführlich im *Referenzhandbuch für die Verwaltungs-API von Adobe Flash Media Server* behandelt.

Dienstprogramm zur Analyse der Server-Auslastung

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Windows oder Linux erforderlich)*

Neben der Verwaltungskonsole und den Server-Protokollen können Sie mit dem Befehlszeilen-Programm FMSCheck den allgemeinen Zustand des Servers analysieren und bewerten. Das Dienstprogramm wird mit Flash Media Server installiert und von Windows und Linux unterstützt. Als befehlszeilenbasiertes Tool lässt es sich problemlos mit vorhandenen Überwachungssystemen im Back-End integrieren.

FMSCheck gibt u. a. an, ob der Server derzeit ausgeführt wird, wie schnell die Verarbeitung von Anforderungen, Ereignissen usw. erfolgt und welche Core-Prozesse oder Anwendungen momentan nicht reagieren. Zudem prüft das Programm alle aktiven Instanzen einer Anwendung, die zurzeit auf dem Server ausgeführt werden, um sicherzustellen, dass sie Verbindungen erwartungsgemäß akzeptieren.

Mit FMSCheck spart sich der Flash Media Server-Administrator die Zeit und den Aufwand, der sonst für eine manuelle Prüfung der Anwendungen und Streams erforderlich wäre. Testverbindungen können je nach Zielvorgabe parallel oder versetzt gestartet werden.

Dienstprogramm zur Videoprüfung

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Windows oder Linux erforderlich; Flash Player optional)*

Angesichts der zahlreichen Codierungstechnologien, die zur Konvertierung von Video in ein Flash Player-kompatibles Format verfügbar sind, können rasch Inkonsistenzen auftreten. Das Befehlszeilen-Programm FLVCheck prüft, ob ein bestimmtes Video von Flash Media Server gestreamt werden kann und ob es manipuliert oder fehlerhaft codiert wurde.

FLVCheck validiert FLV- und MPEG-4-Dateien. Es überprüft die Datenstruktur, einschließlich Kopfdaten, Zeitstempel und Metadaten. Die Validierung kann auf Skriptbasis oder in einer automatisierten Umgebung erfolgen. Bei FLV-Dateien (On2- und Sorenson-Codecs) werden kleinere Fehler behoben, um sicherzustellen, dass die Dateien fehlerfrei ausgeführt werden und die Server-Performance nicht beeinträchtigen.

Skalierung von Flash Media Server 3.5

Server haben begrenzte Kapazitäten, d. h., wenn der Datenverkehr und der Durchsatz steigen, müssen Anwendungen skaliert werden, um die gewohnte Servicequalität (QoS) zu erhalten. Flash Media Server bietet Ihnen mehrere flexible Optionen zur Optimierung von Anwendungen mit hohem Datenaufkommen.

Cluster-Implementierung

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Bei einer Cluster-Implementierung setzen Sie mehrere Server hinter einen Lastenverteiler, um die Anwendungslast gleichmäßig zu verteilen. Clustering mit Flash Media Server bietet den Vorteil, dass Sie eine Anwendung für mehrere Clients zuverlässig skalieren können und die nötige Redundanz erhalten, um Störungen zu vermeiden. Dieser Ansatz eignet sich am besten für das Streaming von Live-Video oder Video-on-Demand (VOD), wenn Clients nicht von spezifischen Anwendungsinstanzen aus miteinander kommunizieren müssen. Eine solche Cluster-Umgebung lässt sich mit Adobe Flash Media Streaming Server oder Adobe Flash Media Interactive Server aufbauen.

Intelligente Lastenverteilung mit Flash Media Server

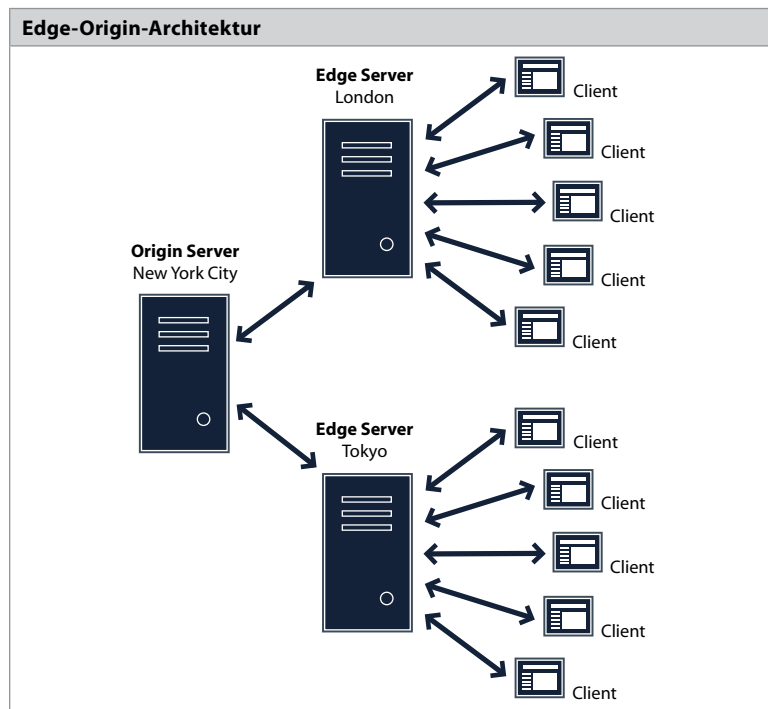
Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Flash Media Interactive Server unterstützt die Umleitung des Datenverkehrs an einen Cluster aus mehreren Servern anhand Server-seitiger Skripten. Diese Option wird normalerweise bei der Mehrkanalkommunikation eingesetzt, wo Verbindungen mit einer Anwendung auf einen bestimmten Server weitergeleitet werden müssen. Nachteilig bei dieser Option ist das Erfordernis recht komplexer Server-seitiger ActionScript-Routinen für die Verbindungsverwaltung.

Edge- und Origin-Konfigurationen

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Wenn Sie in der Vergangenheit die Vorteile des verteilten Cachings und Lastenausgleichs nutzen wollten, mussten Sie die speziellen Edge- und Origin-Editionen von Flash Media Server erwerben. Diese Funktionalität ist nun in Flash Media Interactive Server integriert. Die Server-Software umfasst eine für Unternehmen und Organisationen vorkonfigurierte Architektur, die die Lastenverteilung, Failover-Regulierung und Cluster-Implementierung vereinfacht. So erzielen Sie die größtmögliche Verfügbarkeit für Ihre Inhalte, auch in größeren Regionen. Die folgende Abbildung veranschaulicht die Edge- und Origin-Architektur.



Funktionsweise einer Edge-Origin-Architektur

Edge-Origin-Konfigurationen erhöhen die Server-Performance, indem sie die Server-Last auf mehrere Rechner im Netzwerk verteilen. Im Edge-Origin-Modell werden alle Verbindungsanforderungen der Clients an einen Edge-Server umgeleitet. In dieser Konfiguration lässt sich die Reichweite des Netzwerks vergrößern, falls Sie ein größeres lokales Netzwerk verwalten. Wenn Sie Edge-Server an externen Firmenstandorten aufstellen, speichern diese Server Medien-dateien im lokalen Cache, sodass nicht jeder Stream auf den Origin-Server (Host) zugreifen muss.

Edge-Origin-Konfigurationen eignen sich am besten für unidirektionale Streaming-Dienste. Beim Einsatz eigener Server-Anwendungen für Kommunikation in Echtzeit verarbeitet der Edge-Server sämtliche Anforderungen, die an den Origin-Server gerichtet sind. Die Client-Verbindungen werden dann zur Ausführung der Anwendung zurück an den Origin-Server geleitet.

Flash Media Interactive Server 3.5 bietet integrierte Unterstützung für die Erkennung der Bandbreite und Stream-Länge auf Edge-Ebene. Der erste Server im System (Edge oder Origin), der einen Stream-Aufruf empfängt, prüft auch die Bandbreite und Stream-Länge, ohne dass dazu die Skriptebene des Origin-Servers aufgerufen wird. Diese Funktion ist kompatibel mit der FLVPlayback-Komponente für Flash 8, Flash CS3 Professional und Flash CS4 Professional.

Nach Eingang einer Client-Anforderung führt der Edge-Server die Aufgabe soweit wie möglich aus und stellt nur dann eine Verbindung mit dem Origin-Server her, wenn weitere Daten erforderlich sind. Wenn der Origin-Server die Anforderung erfüllt hat, werden die Daten zunächst an den Edge-Server und dann an den Client zurückgesendet. Auf Client-Seite wirkt es so, als bestünde eine direkte Verbindung mit der Anwendung auf dem Origin-Server.

Der Edge-Server übernimmt Verwaltungsaufgaben in Bezug auf die Verbindung, Authentifizierung etc. und stellt auf diesem Weg wertvolle System- und Netzwerkressourcen für den Origin-Server frei. Jede Verbindung und jeder Verbindungsversuch beanspruchen beim Austausch von Stream-Daten Ressourcen. Wenn Anzahl und Häufigkeit der Verbindungen steigen, nimmt auch die Server-Auslastung zu, was sich negativ auf die Performance auswirkt. Der Edge-Server verringert diese Auslastung erheblich, indem er die ausgehenden Verbindungen mehrerer Clients zu einer Verbindung mit dem Origin-Server zusammenführt. Die Kommunikation zwischen Edge- und Origin-Servern verläuft für die Clients transparent.

Der Edge-Server speichert außerdem die vom Origin-Server empfangenen aufgezeichneten Medieninhalte in einem Cache, der dann auch für andere Clients verfügbar ist, die sich mit dem Edge-Server verbinden. Durch die Cache-Speicherung statischer Inhalte wird die Auslastung des Origin-Servers weiter verringert.

Implementierungsstrategien

Ein einfaches Verfahren zur Lastenverteilung zwischen Edge-Servern besteht in der Zuordnung aller Anwender in einer bestimmten Region oder mit sonstigen gemeinsamen Merkmalen zu einem bestimmten Edge-Server. Ein Edge-Server könnte beispielsweise die Anforderungen von Clients in London zusammenführen und weiterleiten, während ein anderer die Anforderungen aus Tokio verarbeitet.

Eine Netzwerkimplementierung von Flash Media Server umfasst in der Regel mehrere Edge-Server, die entweder einzeln oder in Clustern implementiert sind. Edge-Server können auch kettenartig organisiert werden, um den Datenverkehr noch besser zu verteilen.

Zur Aktivierung der Edge- und Origin-Funktion können Sie einen beliebigen Server in Ihrem Cluster als Origin-Server und die übrigen als Edge-Server konfigurieren. In einer Edge-Origin-Konfiguration müssen alle Server-Editionen identisch sein, d. h., eine Kombination von Flash Media Streaming Server und Flash Media Interactive Server ist innerhalb eines Clusters nicht zulässig.

Die Edge-Origin-Konfiguration eignet sich gut für umfangreiche Flash Media Server-Implementierungen. Eine Einführung in das Edge-Origin-Modell von Flash Media Server finden Sie im *Konfigurations- und Administrationshandbuch*.

Edge-Server werden auch als Proxy-Server bezeichnet. In einer Flash Media Server-Umgebung können Edge- (oder Proxy-) Server auf vier Arten konfiguriert werden:

- Proxy mit automatischer Client-Erkennung
- Proxy mit automatischer Server-Erkennung (Reverse Proxy)
- Explizite URI
- Implizite URI (empfohlen)

Empfohlen wird meist die implizite URI, da diese Konfiguration am sichersten ist und die wenigste Kommunikation erfordert. Zudem kann sie die URI des Origin-Servers verbergen und lässt sich am einfachsten einrichten. Daher wird im Folgenden nur diese Konfiguration beschrieben. Weitere Informationen zu den anderen Methoden erhalten Sie im *Leitfaden zur Anwendung von Edge-Servern mit Flash Media Server*.

Konfiguration von Proxy-Servern mit impliziter URI

Die folgenden Einstellungen definieren den virtuellen Host als Proxy-Server. Mit dieser Konfiguration kann der Client eine Verbindung mit dem Edge-Server aufbauen, ohne den Proxy-Server anzugeben.

```
<Proxy>
  <Mode>remote</Mode>
  <Anonymous>>false</Anonymous>
  <CacheDir enabled="true" useAppName="true">d:\fmsCache\</Cache
  Dir>
  <LocalAddress></LocalAddress>
  <RouteTable protocol="">
```

```
<RouteEntry>edge1.fms.com:*;192.168.110.150:1935</RouteEntry>
```

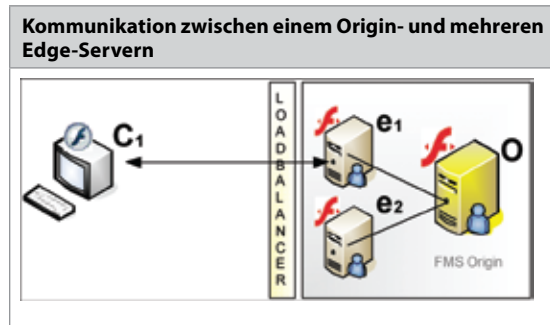
```
</RouteTable>
```

```
</Proxy>
```

Der zugehörige Verbindungs-String lautet:

```
rtmp://edge1.fms.com/ondemand/
```

Wie Sie der folgenden Abbildung entnehmen können, stellt Flash Player eine Verbindung mit dem Edge-Server her, ohne den Origin-Server mit der Kennung 192.168.110.50 anzugeben.



Edge-Server können zu Proxy-Clustern zusammengestellt werden. In der folgenden Abbildung (nächste Seite) ist Edge-Server e1 der Proxy für Edge-Server e0. Die Eigenschaft ist in der Einstellung `routeEntry` definiert. Edge-Server e0 wiederum ist der Proxy für den Flash Media Server-Origin. Diese Konfigurationsart erlaubt die Bildung von Edge-Clustern, deren Lasten geografisch verteilt sind.

`RouteEntry` für die Cluster-Server verweist auf einen Haupt-Edge-Server.

```
<RouteTable protocol="">
```

```
<RouteEntry>edge1.fms.com:*;edge0.fms.com:*;</RouteEntry>
```

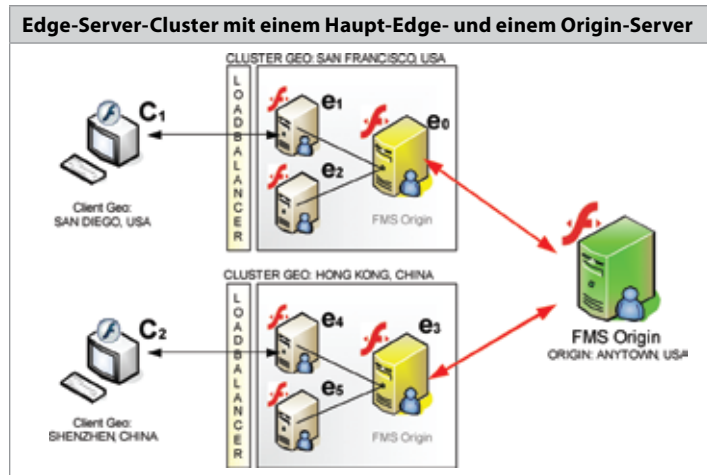
```
</RouteTable>
```

`RouteEntry` für den Haupt-Proxy (Edge-Server) im Cluster verweist zurück auf den Origin-Server.

```
<RouteTable protocol="">
```

```
<RouteEntry>edge0.fms.com:*;origin.fms.com:*;</RouteEntry>
```

```
</RouteTable>
```



Funktionen für Live-Video

Flash Media Server unterstützt die Übertragung von Live-Streams inklusive zahlreicher interaktiver Zusatzelemente. Die verbundenen Clients benötigen zur Wiedergabe lediglich Flash Player oder eine Adobe AIR-Anwendung.

Neu in Flash Media Server 3.5 ist die Möglichkeit, Server-seitige H.264-Streams aufzuzeichnen, sodass Sie interaktive HD-Medieninhalte mit DVR-Funktionalität entwickeln können. Mithilfe der Flash-IDE (Integrated Development Environment) oder Flex Builder erstellen Sie mühelos eigene Anwendungen zur Übertragung von Live-Video. Beachten Sie bitte, dass über Flash Player erfasste und übertragene Videoinhalte mit dem Sorenson Spark-Codec für Video und dem Nellymoser-Codec für Audio codiert werden müssen. Beide Formate sind weit verbreitet und bei Echtzeit-Konferenzen hoch effizient. Falls Sie jedoch die Formate VP6, H.264, AAC bzw. MP3 verwenden oder keine individuelle Codierungslösung benötigen, sollten Sie den eigenständigen Adobe Flash Media Live Encoder in Erwägung ziehen.

Flash Media Live Encoder 3

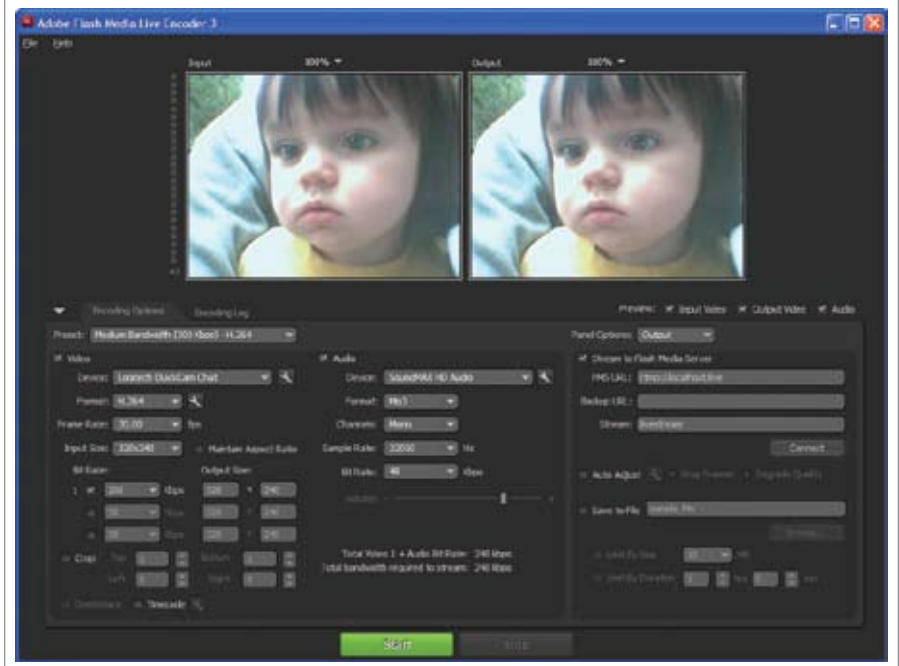
Mit Adobe Flash Media Live Encoder 3 lässt sich Live-Audio und -Video unter Windows Server, Windows XP oder Windows Vista® in Echtzeit während des Streamings zum Flash Media Server oder FVSS erfassen. Für Flash Media Live Encoder sind keine Scripting-Kenntnisse erforderlich. Die Anwendung ist sofort nach der Installation einsatzfähig und umfasst bedienfreundliche Werkzeuge für die Übertragung hochqualitativer Streams zu Sportveranstaltungen, Konzerten, Webcasts, Seminaren u. v. m. – rund um die Uhr.

Die intuitive Benutzeroberfläche (siehe folgende Abbildungen) ist mit zahlreichen Plug-and-Play-Kameras und -Mikrofonen sowie Analog-Digital-Umsetzern kompatibel. Außerdem ermöglicht sie die flexible Bereitstellung von Streaming-Inhalten mit unterschiedlichen Dateigrößen und Bit-Raten. Flash Media Live Encoder 3-Streams sind mit Flash Player 8 (oder höher) kompatibel.

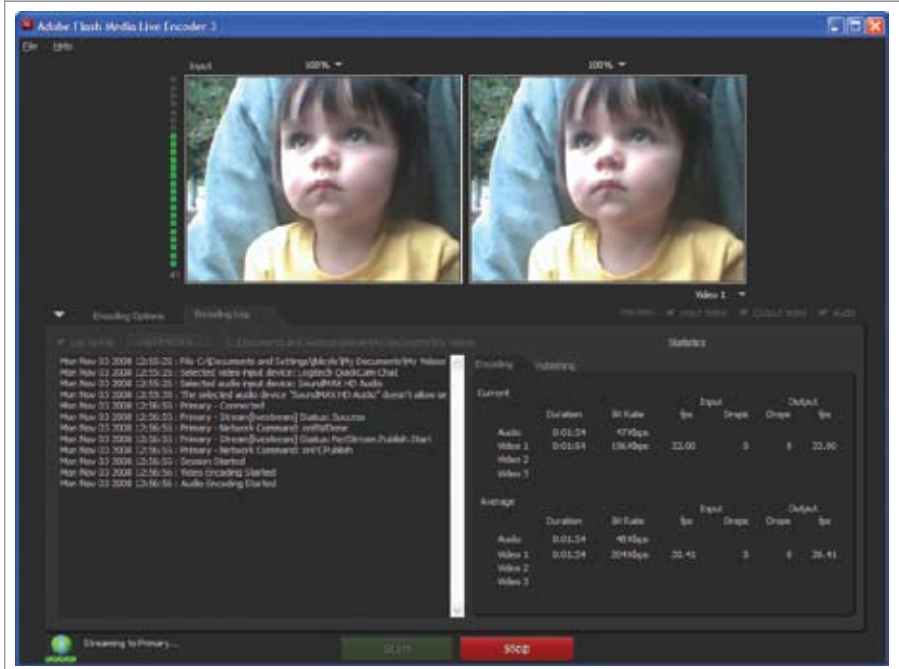
Zu den wichtigsten Neuerungen gehören:

- Gleichzeitige Codierung von bis zu drei Streams mit unterschiedlichen Bit-Raten
- Automatische Anpassung der Codierungsqualität bei schlechteren Netzwerkbedingungen durch Überspringen von Bildern oder Verringerung der Qualität
- Einfügen von Parametern wie der Bit-Rate zur Generierung aussagekräftiger Stream- und Dateibezeichnungen
- Begrenzung von Größe und Dauer der Ausgabedateien durch Codierung eines Streams in mehreren Datenblöcken
- Optimierte, intuitivere Benutzeroberfläche

Benutzeroberfläche von Flash Media Live Encoder



Protokoll einer Live-Übertragung mit Flash Media Live Encoder



Flash Media Live Encoder 3 kann auch in Ihren Streaming-Workflow eingebunden und wahlweise lokal oder per Remote-Zugriff via Microsoft Remote Desktop Connection oder Virtual Networking Computing (VNC) über die Befehlszeile gesteuert werden. Eine weitere wichtige Funktion ist der automatische Neustart nach Stromausfällen oder sonstigen technischen Problemen, der sicherstellt, dass Ihre Live-Streams jederzeit zuverlässig verfügbar sind. Falls Sie ein Gerät einsetzen, das Timecodes generiert, können Sie einen SMPTE-Timecode in den Video-Stream einbetten. Gleiches ist mit System-Timecodes möglich.

Ein signifikanter Vorteil gegenüber einer direkten Übertragung über Flash Player besteht darin, dass Sie mit Flash Media Live Encoder 3 Video neben der Übertragung mit dem VP6- und H.264-Codec Inhalte jetzt auch mit verschiedenen Bit-Raten (maximal drei gleichzeitige Streams) codieren können. Sie haben außerdem bessere Kontrolle über Audioinhalte mit den sparsamen Formaten Nellymoser, MP3 oder AAC.

Flash Media Live Encoder kann unter www.adobe.com/go/fmle kostenlos (in englischer Sprache) heruntergeladen werden.

Daten-Keyframes

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Mithilfe der Funktion für Daten-Keyframes in Flash Media Server 3.5 lassen sich Streams inklusive Metadaten versenden, wenn ein neuer Abonnent einen Stream anfordert. Die Metadaten können zudem jederzeit aktualisiert werden, indem ein neuer Daten-Keyframe über eine Server- oder Client-seitige ActionScript-Routine oder über Flash Media Live Encoder bereitgestellt wird. Alle verbundenen Clients empfangen die neuen Metadaten über die Ereignisprozedur `onMetadata`.

Ein Daten-Keyframe ist eine spezielle Datennachricht, die einem Live-Stream zugeordnet und im Speicher des Servers abgelegt werden kann. Wie andere Datennachrichten enthält ein Daten-Keyframe einen Prozedurnamen und eine Liste mit Eigenschaften, die die Daten speichern. Es können beliebig viele Daten-Keyframes eingerichtet werden. Aus Sicherheitsgründen können nur der Bereitsteller und der Server Daten-Keyframes implementieren und löschen.

Daten-Keyframes werden über die Ereignisse `NetStream.send()` oder `Stream.send()` im Server-seitigen Skript vom Client versendet. Zum Implementieren und Löschen von Daten-Keyframes werden die beiden Spezialmethoden `@setDataFrame` und `@clearDataFrame` verwendet. Das @ wurde hinzugefügt, um Konflikte mit anderen Client-definierten Methoden zu vermeiden.

Multipoint-Publishing

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 6 oder höher erforderlich)*

Die Skalierbarkeit ist ein weiterer wichtiger Aspekt bei Live-Übertragungen. Aufgrund der eingeschränkten Prozessorressourcen und Netzwerk-Bandbreite kann ein Server nur eine vorgegebene Anzahl von Abonnenten bedienen. Bei weiteren gleichzeitigen Zugriffen muss der Datenverkehr teilweise von anderen Servern übernommen werden. Multipoint-Publishing ermöglicht die Bereitstellung von Inhalten an mehrere Server mit nur einer Client-Server-Verbindung.

Bislang waren mit der API im Server-seitigen Skript nur unidirektionale Streams von einem Remote-Server über die Objekte `Stream` und `NetConnection` möglich. Somit konnte ein Stream von einem Remote-Server wiedergegeben werden. Um Streams an einen Remote-Server zu senden bzw. festzustellen, welche Streams versuchten, auf einem anderen Server veröffentlicht zu werden, mussten alle Server direkt miteinander verbunden sein. Aufgrund des eingeschränkten Zugriffs von Server-seitigen Skripten auf CDNs war die erneute Veröffentlichung von Streams nicht möglich.

Mit den APIs von Flash Media Interactive Server 3.5 können bei Bedarf Streaming-Daten dynamisch an einen anderen Server umgeleitet werden, ohne dass dazu eine permanente Verbindung zwischen den Servern oder spezieller Server-seitiger Code erforderlich ist. Zur optionalen Anpassung sind jetzt `Publish`- und `Unpublish`-Ereignisse für Server-seitige Benachrichtigungen verfügbar.

Multipoint-Publishing schafft die dringend erforderliche Flexibilität, Anpassbarkeit und Skalierbarkeit für Anwendungen, die Live-Video in großem Maßstab streamen. Weitere Informationen finden Sie im *Entwicklerhandbuch für Adobe Flash Media Server*.

Datenschutz mit Flash Media Server 3.5

Beim elektronischen Austausch von Inhalten besteht immer das Risiko, dass sie kopiert, verändert oder unbefugt weitergegeben werden. Flash Media Server bietet mehrere Sicherheitsstufen für den Schutz Ihrer Daten und Server-Ressourcen, die für den Endanwender unauffällig, intuitiv und bequem sind.

Sicherheitsrisiken für Inhalte

Verschiedene Faktoren gefährden die Integrität von Online-Inhalten:

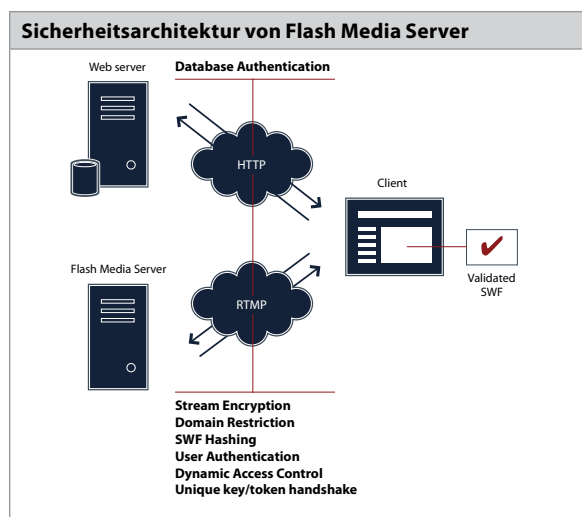
- **Auslesen des Browser-Caches** – Obwohl die Dateinamen nicht leicht lesbar sind, ist es relativ einfach, die Videodateien aus dem Browser-Cache abzurufen. Dieses Risiko besteht nur bei der progressiven Video-Bereitstellung, da Streams nie im Cache gespeichert werden.
- **Zugriff auf die Video-URI** – Video-URIs können mit kostenlosen Sniffer-Programmen leicht ermittelt werden.
- **Erneute Übertragung der SWF-Datei** – Ihre SWF-Datei kann kopiert und dann von einer anderen Domäne erneut bereitgestellt werden. SWF-Dateien lassen sich zudem decompilieren und enthüllen dabei Ihre Flash Media Server-Adresse, Anwendungen und Stream-Bezeichnungen.
- **Streamripping** – Diese Art von Stream-Angriff zählt zu den größten Sicherheitsproblemen, da sie am schwierigsten zu bekämpfen ist. Streamripping-Tools fangen den Datenstrom ab und zeichnen ihn in einer Datei auf, die dann unabhängig wiedergegeben werden kann.

Sicherheitsarchitektur von Flash Media Server

Streaming ist sicherer als der progressive Download, da Mediendateien nicht im Cache-Speicher der Festplatte abgelegt werden. Flash Media Server bietet erweiterte Sicherheitsfunktionen, die den Schutz Ihrer Inhalte erhöhen:

- Anwenderauthentifizierung mit Server-seitigem ActionScript
- Autorisierungs-Adapter
- Zugriffs-Adapter
- Verifizierung von SWF-Dateien
- Zugriffskontrolle auf Domänenebene
- Speziallösungen von CDNs (Content Delivery Network)
- Stream-Verschlüsselung mit RTMPE oder RTMPS

Zunächst wird die allgemeine Sicherheitsarchitektur von Flash Media Server erläutert, die in der folgenden Abbildung dargestellt ist. Danach werden die Schutzmaßnahmen im Einzelnen erklärt. Ungeachtet der Vertraulichkeit und Besitzverhältnisse Ihrer Inhalte sollten Sie bei der Online-Bereitstellung Sicherheitsvorkehrungen treffen. Am besten sichern Sie zuerst den Server und dann die Inhalte.



Einschränkung des Zugriffs von Domänen

Clients können sich standardmäßig von jeder Domäne oder IP-Adresse aus mit Flash Media Server verbinden. Dies stellt ein gewisses Sicherheitsrisiko dar. Sie können eine Whitelist der zulässigen Domänen (oder eine Blacklist der gesperrten Domänen) führen, um sicherzustellen, dass nur autorisierte Clients Ihre Anwendungen oder Dienste nutzen. Oder Sie fügen eine kommagetrennte Liste der Domänen und IP-Adressblöcke zu den Konfigurationsdateien `Adaptor.xml` und/oder `vHost.xml` hinzu, um die Sicherheit zu erhöhen. Auf diese Weise verhindern Sie, dass unbefugte Anwender oder Domänen auf Ihre Anwendungen oder Streams zugreifen.

Server-seitiges ActionScript

Flash Media Server 3.5 verfügt über mehrere Methoden zur Anwenderauthentifizierung. In diesem Abschnitt wird die Server-seitige ActionScript-Methode zur Implementierung eines Authentifizierungsschemas zur Validierung der Client-Verbindung erläutert. Durch die Weitergabe von Variablen über die Client-seitige Methode `NetConnection` lässt sich beispielsweise eine einfache Lösung mit Benutzername/Kennwort, einem verschlüsselten Token (MD5-Hash) oder einem eindeutigen Schlüssel einrichten.

- Identitätsdaten (Benutzername und Kennwort)

```
NetConnection.connect("rtmp...", "username", "password");
```

- Verschlüsselter Token (MD5-Hash)

```
NetConnection.connect("rtmp...", 6aef79f07bc8f23c38e8979f3630f436);
```

- Eindeutiger Schlüssel

```
NetConnection.connect("rtmp...", 349jh3k4324h9.234234098);
```

Dann kann Flash Media Server mittels Webservices (SOAP), Flash Remoting, XML, HTTP POST oder einfachen Dateizugriffs den Client auf Basis der gesendeten Daten überprüfen. Dieses Authentifizierungsschema kann beliebig komplex sein. Eine schlichtere Variante wäre der Abgleich von Anmeldedaten mit einer Datenbank. Präziser wäre beispielsweise ein mit ColdFusion entwickeltes SSL-basiertes Token-System.

Zugriffs-Adapter (Plug-in)

Erweiterte Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Ein Zugriffs-Plug-in ist ein Server-Plug-in, das in C++ verfasst wird und Verbindungsanforderungen an den Server abfängt, um festzustellen, ob diese Anforderungen genehmigt, zurückgewiesen oder umgeleitet werden sollten, bevor sie die Skript-Ebene des Servers erreichen. Der Zugriffs-Adapter kann durch eigene Programmlogik zur Verarbeitung der Verbindungsanforderungen von Clients ergänzt werden. Beispielsweise können Sie nach einer Client-Anmeldung Ihre Kontodatenbank abfragen und dann nach Genehmigung der Client-Verbindung den Datensatz in der Datenbank aktualisieren.

Das Zugriffs-Plug-in kann so konfiguriert werden, dass es Anforderungen je nach Anzahl der derzeitigen Client-Verbindungen oder der derzeit genutzten Bandbreite genehmigt bzw. zurückweist. Sie können außerdem Lese- und Schreibberechtigungen für Dateien und Ordner auf dem Server einrichten, Zugriffsrechte für Audio- und Video-Bitmap-Daten definieren und Client-Eigenschaften über den Zugriffs-Adapter überprüfen.

Mit dem Zugriffs-Adapter fangen Sie die Verbindung ab, bevor sie von Flash Media Server verarbeitet wird. Aus diesem Grund können Sie nur auf Verbindungsereignisse zugreifen. Falls Sie nach dem Aufbau der Verbindung weitere Regeln anwenden möchten, müssen Sie einen Autorisierungs-Adapter konfigurieren.

Hinweis: Für jede Flash Media Interactive Server-Installation kann nur ein Zugriffs-Plug-in verwendet werden.

Autorisierungs-Adapter (Plug-in)

Funktion in Flash Media Interactive Server (Flash Player 6 oder höher erforderlich)

Die nächste Schutzebene bildet der Autorisierungs-Adapter. Über ein C++-basiertes Server-Plug-in autorisiert der Adapter den Client-Zugriff auf Server-Ereignisse. Der Adapter wird aktiv, sobald eine Verbindung hergestellt wurde, jedoch noch vor ihrer Genehmigung.

Autorisierungs-Adapter bieten folgende Funktionen:

- Autorisierung von Verbindungen zum Server
- Autorisierung der Wiedergabe oder Navigation innerhalb eines Streams
- Autorisierung der Veröffentlichung eines Streams
- Abbruch einer Verbindung zwischen Client und Server
- Aufruf einer Methode im Server-seitigen ActionScript
- Bereitstellung von Inhalten an Clients nach Standort, Abonnement und Stream-Herkunft
- Einschränkung von Zeitpunkt und Dauer des Zugriffs auf bestimmte Streams
- Zuordnung eines logischen Stream-Pfads zu einem physischen Stream-Pfad. Beispiel: Ein Client fordert den Stream „foo.flv“ an. Da er kein Premium-Abonnent dieses Dienstes ist, erhält er nur die geringerwertige Version des Inhalts „bar.flv“.

Anders als beim Zugriffs-Adapter können Sie mehrere Autorisierungs-Adapter einsetzen, die nacheinander Aktionen für das eingehende Ereignis ausführen. Beispiel: `auth1.dll` (oder `auth1.so`) autorisiert die Client-Verbindung und `auth2.dll` (oder `auth2.so`) danach die Veröffentlichung eines Streams durch den Client. Der Server wendet die Adapter in alphabetischer Reihenfolge an.

Autorisierungs-Adapter können somit einen wesentlichen Beitrag zur Stream-Sicherheit leisten und Zugriffe präzise kontrollieren. Bei Bedarf lassen sich benutzerdefinierte Funktionen für die Verwaltung von Nutzungsrechten oder zur Protokollierung hinzufügen.

Dynamische Zugriffskontrolle

Clients haben standardmäßig vollständigen Zugriff auf Streams und gemeinsame Objekte auf dem Server. Mithilfe von Server-seitigem ActionScript lässt sich der Zugriff einschränken. Sie können eine dynamische ACL (Access Control List) erstellen, in der die Berechtigungen zum Lesen, Erstellen oder Aktualisieren von gemeinsamen Objekten oder Streams genau definiert sind.

In Server-seitigem ActionScript wird jeder Client, der eine Verbindung zum Server herstellt, einem Client-Objekt mit den Eigenschaften `readAccess` und `writeAccess` zugewiesen. Diese Eigenschaften akzeptieren mehrere kommagetrennte Werte. Wenn Sie diese Werte beim Genehmigen der Client-Verbindung definieren, können Sie festlegen, auf welche Streams und gemeinsamen Objekte ein Client zugreifen kann.

Verschlüsselung von Streaming-Inhalten

Flash Media Server 3.5 bietet zwei Optionen zur Verschlüsselung von Streams: SSL und RTMPE.

SSL

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 8 oder höher erforderlich)*

In früheren Versionen von Flash Media Server war die Verschlüsselung von Streaming-Inhalten bei der Bereitstellung über SSL mittels RTMPS verfügbar. Dies ist auch in Flash Media Server 3.5 möglich. Für die Implementierung ist ein extern ausgestelltes Zertifikat und die Konfiguration der Server-Seite erforderlich. Flash Media Server 3.5 bietet jetzt ein einfacheres Verfahren zur Verschlüsselung: RTMPE.

RTMPE

*Funktion in Flash Media Streaming Server und Flash Media Interactive Server
(Flash Player 9,0,115,0 oder höher erforderlich)*

Verschlüsseltes RTMP (RTMPE) ist in Flash Media Server standardmäßig aktiviert und ermöglicht das Versenden von Streams über eine verschlüsselte Verbindung ohne Zertifikatverwaltung. Die sichere 128-Bit-Verschlüsselung durch RTMPE wird nur von Flash Player 9 (und höher), der aktualisierten FLVPlayback-Komponente und den `NetConnection`-Klassen unterstützt. Sowohl SSL als auch RTMPE können getunnelt werden, um eine Verbindung über die Firewall eines Netzwerks hinweg zu ermöglichen. Die Verschlüsselung mit RTMPE wird empfohlen, weil sie leichter zu implementieren und wesentlich schneller als SSL ist.

Die Implementierung von Stream-Verschlüsselung in Ihren Anwendungen ist einfach. Geben Sie beim Verbindungsaufbau mit Ihrer Anwendung das Protokoll vor.

- SSL

```
NetConnection.connect("rtmps://yourFMSServer.com");
```

- Getunnelte SSL-Verbindung

```
NetConnection.connect("rtmpts://yourFMSServer.com");
```

- Enhanced RTMP

```
NetConnection.connect("rtmpe://yourFMSServer.com");
```

- Getunnelte Verbindung per Enhanced RTMP

```
NetConnection.connect("rtmpte://yourFMSServer.com");
```

Den optimalen Schutz für Ihre Inhalte bietet eine Kombination aus RTMPE und SWF-Verifizierung, wobei das standardmäßige RTMP bei Verwendung von RTMPE oder RTMPTE deaktiviert sein sollte.

Schutz vor Streamripping

Streamripping-Technologien sind ein heikles Sicherheitsproblem, da sie den direkten Zugriff auf und die Aufzeichnung der Daten in einem Stream erlauben.

Die unbefugte Aufzeichnung lässt sich mittels Stream-Verschlüsselung und SWF-Verifizierung verhindern. Bislang konnte nur SSL ausreichend Schutz bieten, allerdings war die Lösung für die meisten Anwendungen zu langsam. Der neue Flash Media Server 3.5 verwendet jetzt das wesentlich effizientere und einfacher zu implementierende RTMPE.

Ein weiteres Verfahren zum Schutz vor Streamripping ist der Einsatz intelligenter Funktionen beim Datenaustausch zwischen Server und Client. Zusätzlicher Code im Video-Player kann beispielsweise erfordern, dass die SWF-Datei zur Verifizierung einer vom Server gesendeten Zeichenfolge auf eine Anforderung von Flash Media Server antwortet. Auf diese Weise wird der Datenfluss zur Streamripping-Technologie unterbrochen; da sie nicht über die nötigen Daten verfügt, wird der Zugriff auf den Stream abgelehnt.

DRM (Digital Rights Management)

Erweiterte Funktion in Flash Media Streaming Server und Flash Media Interactive Server (Flash Player 6 oder höher erforderlich; Flash Player 9,0,115,0 oder höher für RTMPE/SWF-Verifizierung erforderlich)

DRM-Lösungen (Digital Rights Management) dienen sowohl der Verschlüsselung als auch der Zugriffskontrolle. Zur Bereitstellung von Video stehen zwei Verfahren zur Verfügung: Streaming oder Download. Beim Streaming mit Flash Media Server wird automatisch die Sicherheit Ihrer Inhalte erhöht.

Bei Flash Media Server erfolgt die Verschlüsselung in Echtzeit mit RTMPS (SSL) bzw. bei Flash Media Server 3.5 mit RTMPE. On-Demand-Inhalte lassen sich auch mit Flash Media Rights Management Server verschlüsseln. Flash Media Server kann diese verschlüsselten Inhalte an Adobe AIR-Anwendungen wie Adobe Media Player streamen. Die Bereitstellung verschlüsselter FLV- und F4V-Dateien erfolgt mit den RTMP-basierten Protokollen RTMPPT und RTMPE oder mit HTTP. Zur Zusammenarbeit von Flash Media Server und Flash Media Rights Management Server ist keine spezielle Konfiguration erforderlich.

Zur Steuerung des Zugriffs auf Streaming-Inhalte führt Flash Media Server eine SWF-Verifizierung durch. Bei Flash Media Interactive Server ist dieses Verfahren aufgrund der neuen Plug-in-Architektur sowie der Server-seitigen Anwendungsebene wesentlich leistungsfähiger. Mit Webservices (SOAP), Flash Remoting und XML können Sie ein effektives Kontrollsystem mit sicheren Tokens erstellen.

Dies sind die grundlegenden Prinzipien von DRM für Streaming-Inhalte. Bei Downloads setzen Sie zusätzlich Flash Media Rights Management Server für die Bereitstellung signierter und verschlüsselter Medieninhalte ein. Weitere Informationen über die Verwaltung von Nutzungsrechten und Flash Media Rights Management Server erhalten Sie unter www.adobe.com/go/fmrms_de.

CDNs

Eine einfache Lösung, Streaming-Inhalte zusätzlich zu schützen, ist der Flash Video Streaming Service, den Sie über einen CDN-Partner von Adobe in Anspruch nehmen können. Viele FVSS-Partner bieten Optionen zur Einschränkung des Zugriffs sowie eigene Lösungen für sicheres Video-Streaming an.

Weitere Informationen über den Schutz von Inhalten über ein CDN erhalten Sie unter www.adobe.com/go/fvss_de.

Glossar

Adobe AIR: Das plattformübergreifende Werkzeug ermöglicht es Web-Entwicklern, ihre Kenntnisse in HTML, Ajax, Adobe Flash und Adobe Flex auch beim Erstellen von Rich-Internet-Anwendungen für den Desktop zu nutzen.

Adobe Media Player: Ein Media-Player für den Desktop, der die Vorteile von klassischem Fernsehen und Web-Video vereint. Hochwertige Inhalte werden online und offline konsistent wiedergegeben. Zudem lässt sich der Player an die Anforderungen individueller Geschäftsmodelle anpassen.

Bandbreite: Die Bandbreite beschreibt den Durchsatz eines Servers oder Clients und wird meist in Megabit pro Sekunde (Mbps) oder Kilobit pro Sekunde (kbps) gemessen. Ein Ethernet-Kabel hat in der Regel eine Bandbreite von 100 Mbps, eine drahtlose Verbindung 54 Mbps. Die Bandbreite von Server und Client bestimmt das Volumen an Videodaten, die empfangen oder gesendet werden können.

CDN (Content Delivery Network): CDN-Anbieter stellen Streaming-Dienste und Bandbreite zur Verfügung, damit Kunden keine eigenen Server einrichten und installieren müssen.

Client: Rechner des Verbrauchers, der über Flash Player oder eine Adobe AIR-Anwendung eine Verbindung zu Flash Media Server aufbaut.

Codec: Codecs sind die Formate, in die eine Video- oder Audiodatei codiert wird. Flash verwendet die Video-Codecs Sorenson Spark, On2 VP6-S, On2 VP6-E und H.264 sowie die Audio-Codecs Nellymoser, MP3 und AAC. Der Begriff „Codec“ setzt sich aus „Code“ und „Decode“ zusammen. Die Decodierfunktionen des Codecs müssen im Player integriert sein, damit Video mit dem zugehörigen Codec abgespielt werden kann.

DRM (Digital Rights Management): Mit DRM codiertes Video ist gegen die unbefugte Weitergabe und Diebstahl geschützt.

Encoder: Encoder komprimieren oder transcodieren Video in ein anderes Format.

Enhanced RTMP: RTMPE ist die nächste Generation des RTMP-Protokolls (Real Time Messaging Protocol) und bietet mehr Sicherheit und Leistung als die Vorgängerversionen.

Flash Lite 3: Der Flash Player für mobile Endgeräte unterstützt die Codecs VP6 und Spark sowie RTMP-Verbindungen zu Flash Media Server.

Flash Media Live Encoder: Die kostenlose, auf Windows XP basierende Desktop-Anwendung verbindet sich mit Flash Media Server und ermöglicht das Streamen von Live-Video und -Audio an Flash Player.

Flash Media Server Solution Provider-Programm: Das Partnerprogramm wurde ins Leben gerufen, um eine starke Anbieter- und Entwicklergemeinde rund um Flash Video und Flash Media Server aufzubauen.

Flash Video Streaming Service (FVSS): In Zusammenarbeit mit führenden CDN-Partnern bietet Adobe gehostete Dienste für die Bereitstellung von On-Demand-Video für Flash Player über hochleistungsfähige, zuverlässige Netzwerke an.

Flex: Flex ist ein plattformübergreifendes Open-Source-Framework für die Erstellung von Rich-Internet-Anwendungen, die in allen gängigen Browsern und auf allen Plattformen konsistent ausgeführt werden.

Inhalt: Video- oder Audiodaten, die von Flash Media Server gestreamt werden.

Live: Flash Media Live Encoder und Flash Player unterstützen die Live-Übertragung von Flash-Streams.

On2 VP6: Der Video-Codec ermöglicht die hochwertige, effiziente Wiedergabe im Vollbildmodus (erhältlich seit Flash Player 8). VP6-S ist eine vereinfachte Version, die ideal für die Bereitstellung hochwertiger Videoinhalte an ältere Rechner geeignet ist (erhältlich seit Flash Player 9). Mit dem VP6-E-Codec, der ursprünglich mit Flash Player 8 verfügbar war, erhalten Sie eine bessere Qualität, allerdings erfordert die Wiedergabe eine höhere Prozessorleistung.

Publishing-Point: Mit Publishing-Point wird ein Verzeichnis auf Flash Media Server bezeichnet, in das Kunden Video- und Audioinhalte platzieren können und von dem aus Live-Video veröffentlicht wird.

Puffer: Menge an Videodaten, die im RAM des Clients gespeichert werden. Je größer der Pufferspeicher, desto reibungsloser die Wiedergabe. Pufferdaten werden nie auf die Festplatte geschrieben.

Quality of Service (QoS): QoS beschreibt die Qualität des Wiedergabeerlebnisses aus der Sicht des Verbrauchers.

Real Time Message Protocol (RTMP): RTMP ist Adobes proprietäres Protokoll für die Kommunikation zwischen Flash Player-Clients und Flash Media Server.

Sorenson Spark: Dieser Codec war ursprünglich im Lieferumfang von Flash Player 6 und 7 enthalten. Ein Encoder für diesen Codec ist im Flash Player integriert. Dies ermöglicht die Übertragung von Webcam-Daten und die Archivierung von Inhalten, wenn Flash Media Server eingesetzt wird.

Transcodierung: Transcodierung ist der Fachbegriff für die Konvertierung von Videodateien. Bei der Transcodierung können Sie in der Regel den Codec ändern. Jedes Mal, wenn eine Datei transcodiert wird, führt dies zu Qualitätsverlusten.

Verbindung: Clients nutzen Verbindungen, um Video-Streams zu empfangen. Wenn mehrere Clients auf einmal auf Streaming-Inhalte zugreifen, spricht man von gleichzeitigen Verbindungen.

VOD (Video on demand): Bereitstellung von vorab aufgezeichneten Video-Streams auf Abruf.

Online-Ressourcen

Flash On™ - Galerie

www.adobe.com/go/flashon (Englisch)

HD-Videos mit Adobe Flash

www.adobe.com/de/products/hdvideo

Adobe HD - Galerie

www.adobe.com/products/hdvideo/hdgallery (Englisch)

Erläuterungen zu den Unterschieden zwischen progressivem Download und Video-Streaming

www.adobe.com/de/devnet/flash/articles/video_guide_02.html

www.adobe.com/de/products/hdvideo/supported_technologies/streaming.html

DRM und Schutz digitaler Medieninhalte mit Flash Media Server

www.adobe.com/devnet/flashmediaserver/articles/digital_media_protection.html (Englisch)

www.adobe.com/de/products/hdvideo/supported_technologies/content_protection.html

Optimierung der Performance von Flash Media Server 2 für Live-Webcasts unter Linux

www.adobe.com/devnet/flashmediaserver/articles/performance_tuning_webcasts.html

(Englisch)

Flash Player-Unterstützung für HD-Inhalte in H.264 (Video) und AAC (Audio)

www.adobe.com/devnet/flashplayer/articles/hd_video_flash_player.html (Englisch)

AAC-HE 2

www.adobe.com/de/products/hdvideo/supported_technologies/heaacv2.html

H.264

www.adobe.com/de/products/hdvideo/supported_technologies/h264.html

Streaming durch Firewalls und Proxys aus der Perspektive des Clients

www.adobe.com/devnet/flashcom/articles/firewall_streaming.html (Englisch)

Adobe Flash Media Server 3.5

www.adobe.com/go/fms_de

Adobe Flash Media Encoding Server

www.adobe.com/go/fmes_de

Adobe Flash Media Rights Management Server

www.adobe.com/go/fmrms_de

Adobe Flash Media Live Encoder

www.adobe.com/de/products/flashmediaserver/flashmediaencoder

Adobe FVSS-Partner (Flash Video Streaming Service)

www.adobe.com/go/fvss_de

Adobe Flash Media Solution Provider-Programm

www.adobe.com/de/products/flashmediaserver/partners

Häufig gestellte Fragen zum Flash Player 9 Update

http://labs.adobe.com/wiki/index.php/Flash_Player:9:Update:H.264 (Englisch)

Flash Media Server-Communitys

FMSGuru

www.fmsguru.com (Englisch)

FlashComGuru

www.flashcomguru.com (Englisch)

FlashConnections

www.flashconnections.com (Englisch)

Flash Video | Optimizations and Tools

<http://flashvideo.progettosingia.com> (Englisch)

Quellen

Leitfaden zur Bereitstellung von Medieninhalten per Streaming

www.adobe.com/products/aftereffects/pdfs/AdobeStr.pdf (Englisch)

Adobe-Einführung in Digitalvideo: HD-Video verstehen und einsetzen

www.adobe.com/designcenter/productionstudio/articles/prslip_hdprimer/prslip_hdprimer.pdf
(Englisch)

Schutzmaßnahmen für Videoinhalte in Flash Media Server

www.adobe.com/devnet/flashmediaserver/articles/protecting_video_fms.html (Englisch)

Über die Autorin

Lisa Larson-Kelley ist Entwicklerin, Dozentin, Beraterin und Expertin für elektronische Medien. Sie ist Mitverfasserin des Buches *Flash Video for Professionals* (Wiley, 2007) und Autorin des FlashConnections-Blogs (www.flashconnections.com).



Adobe Systems GmbH

Georg-Brauchle-Ring 58 • D-80992 München

Adobe Systems (Schweiz) GmbH

World Trade Center • Leutschenbachstrasse 95 CH-8050 Zürich

www.adobe.de, www.adobe.at, www.adobe.ch, www.adobe.com

Adobe, das Adobe-Logo, ActionScript, Adobe AIR, ColdFusion, Flash, Flash Lite, „Flash On.“, Flex, Flex Builder, Macromedia und XMP sind Marken oder eingetragene Marken von Adobe Systems Incorporated in den USA und/oder anderen Ländern. Mac ist eine eingetragene Marke von Apple Inc. in den USA und anderen Ländern. HP ist eine eingetragene Marke der Hewlett-Packard Company. Intel, Intel Xeon und Pentium sind Marken der Intel Corporation in den USA und anderen Ländern. Linux ist eine eingetragene Marke von Linus Torvalds in den USA und anderen Ländern. Microsoft, Windows, Windows Server und Windows Vista sind Marken oder eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern. Red Hat ist eine Marke oder eingetragene Marke von Red Hat, Inc. in den USA und anderen Ländern. Alle anderen Marken sind Eigentum der jeweiligen Inhaber.
Adobe übernimmt keine Haftung für eventuelle Druckfehler.

© 2009 Adobe Systems Incorporated. Alle Rechte vorbehalten. Printed in Germany.
95011937 1/09