

X
M
M
magazin

Das Magazin für Macromedia Anwender

MX
magazin

Ausgabe **8**/2005

Deutschland € 9,90
Österreich € 10,90
Schweiz SFR 19,90

S&S



KOSTENLOS

HEFT-CD

Macromedia-Tools*
Erstmalig verfügbar
ColdFusion MX 7
Flex 1.5
Dreamweaver MX 2004
Flash MX 2004

*Testversionen

Professionelle **Visualisierung**



Ideen, Hintergründe, Beispiele

- **Brandaktuell: Das leistet das neue ColdFusion MX 7**
- **Programmierung: XHTML und CSS richtig eingesetzt**
- **Star Trek: Hollywood-Effekte in Flash**

Flash

Automatisiertes Testen
von großen Anwendungen



Flex

Einstieg in die Grundlagen



Director

Ego-Shooter
im Eigenbau



Philipp Cielen

ColdFusion & Flash



Dateien auf CD

Die Kombination von ColdFusion und Flash zur Erstellung dynamischer Websites mit interaktiven, grafischen Benutzeroberflächen gewinnt seit einigen Jahren stetig an Popularität. Die neue Version von ColdFusion ist eine gute Möglichkeit, grundlegend auf die Möglichkeiten der Kombination von CF und Flash einzugehen. MX Magazin geht in einer neuen Artikelserie auf die verschiedenen Umsetzungsarten für solche Applikationen ein und arbeitet Vor- und Nachteile der verschiedenen Lösungen heraus.

Macromedia fördert und unterstützt die Kombination von CF und Flash nach Kräften und propagiert das Modell der „Rich Internet Applications“ als Technologie der Wahl, wenn es darum geht, moderne Webapplikationen plattformunabhängig umzusetzen. Anhand eines konkreten Anwendungsbeispiels, das sich als roter Faden durch die Artikelreihe zieht, können Sie die einzelnen Schritte zur Umsetzung nachvollziehen und am Ende selbst entscheiden, welche Lösung für Ihre eigenen Anforderungen am ehesten in Betracht zu ziehen ist.

Macromedia Flash hat in den letzten Jahren eine langsame, aber stetige Entwicklung vom reinen Design-Tool für Animationen zu einer Entwicklungsumgebung für grafische Benutzeroberflächen von Websites und sogar ganz klassische Desktop-Applikationen durchgemacht. Der Begriff *Flash* steht dabei nicht nur für die Entwicklungsumgebung, sondern gleichzeitig auch für das Format, in das die Applikationen oder Animationen verpackt werden. Macromedia hat dieses Format, das ursprünglich unter dem Namen *Shockwave Flash* bekannt wurde, schon länger komplett offen gelegt und damit auch Konkurrenzanbietern die

Gelegenheit gegeben, eigene Entwicklungsumgebungen oder Generatoren für dieses Format zu produzieren. Im Unterschied zu klassischen Open-Source-Projekten bleibt die Weiterentwicklung des Formats allein Macromedia selbst vorbehalten. Third-Party-Tools-Anbieter können also nur auf dem Code aufbauen, den Macromedia vorgibt.

Wieso Flash? Bei der Technologieauswahl stellt sich meist zuerst die Frage, wieso Flash überhaupt als Alternative klassischen, browserbasierten HTML-beziehungswise DHTML-Oberflächen vorzuziehen ist. Die Antworten auf diese Frage sind vielfältig und hängen in erster Linie mit der Art der Anforderung zusammen. Flash bietet einige Möglichkeiten, die klassisches HTML derzeit und auch in absehbarer Zukunft nicht bieten kann, beispielsweise die Integration von (maskierten) Videos mit interaktiven Elementen, Vektorgrafik- und Bitmap-Animationen, die komplette Unabhängigkeit von vorgegebenen Benutzeroberflächenelementen und die Loslösung vom Paradigma des Page Refresh als einzige Möglichkeit der Kommunikation mit einem Server. Im Allgemeinen ist Flash

vor allem dann geeignet, wenn eine intelligente, interaktive Verarbeitung oder Modifikation von Daten direkt auf dem Client stattfinden soll. Solche Clients, die mehr als nur die reine Darstellung von Daten bewerkstelligen, bezeichnet man auch als *Fat Clients* oder *Rich Clients*. Macromedia leitet von letzterem Begriff das Konzept der *Rich Internet Applications* ab.

Bei allen Vorteilen, die Flash auf dem Gebiet der Webapplikationen oder Präsentationen bietet, bleiben doch einige Anwendungsgebiete für Flash weniger geeignet. Für die reine Darstellung von Informationen bleibt in den allermeisten Fällen HTML weiterhin das Mittel der Wahl, auch wenn Macromedia selbst diesen Bereich mit dem neuen Format *FlashPaper* angeht. *FlashPaper* ist ein Format zur Darstellung von Informationen und ähnelt Adobes PDF-Format. Beide eignen sich vorwiegend dazu, Daten layoutgetreu darzustellen, wie sie im Druck erscheinen würden, weniger jedoch als alternative Darstellungsform zu HTML für Informationen.

Wie wird geflasht? Flash-Webanwendungen können heute auf ganz verschiedene Arten hergestellt werden. Der

klassische und gleichzeitig flexibelste Weg führt über das Entwicklungs-Tool Flash, mit dem Layout und Bedienelemente sowohl völlig frei gestaltet als auch vorgefertigte Komponenten angepasst und erweitert werden können. Allein hier stehen dann wiederum zahlreiche Wege offen, mit einem Server wie ColdFusion zu kommunizieren. Von der einfachen Datenübergabe per URL-String über das Einlesen von Textdateien oder XML-Strings bis zu Flash Remoting und der Kommunikation über Web Services stehen hier alle Möglichkeiten offen. Von den verschiedenen Varianten des clientseitigen Kodierens dieser Datenübertragungsmethoden ganz zu schweigen.

Einige neue Frameworks bieten etwas abstraktere Methoden, Flash-basierte Webapplikationen zu erstellen – sie ermöglichen die komplett codegesteuerte Erstellung von Applikationen. Die Verwendung der ursprünglichen Flash-Entwicklungsumgebung wird hierbei nicht mehr benötigt. Beispiele sind der so genannte *Flex Presentation Server* von Macromedia, mit dessen Hilfe Flash-Frontends für verschiedene Backend-Plattformen unter Verwendung einer XML-basierenden Skriptsprache namens MXML entwickelt werden können. Einen ähnlichen Ansatz verfolgt die Open-Source-Alternative *Laszlo* von Laszlo Systems, die ebenfalls Flash-Oberflächen mithilfe XML-basierender Tags generiert.

Generell ist zu erkennen, dass die Kommunikation zwischen Front- und Backend an sich immer mehr abstrahiert wird, um Entwickler nicht mit der händischen Verpackung von Datenbündeln zu belasten. Das manuelle Serialisieren und Deserialisieren von Daten über XML oder gar URL-codierte Strings soll der Vergangenheit angehören.

Wo kommt ColdFusion ins Spiel?

Mit der neuen Version ColdFusion MX 7 bietet Macromedia auch reinen ColdFusion-Entwicklern die Möglichkeit, Flash skriptgesteuert zu generieren. Genau genommen existiert die Möglichkeit, Flash über reines Scripting zu erzeugen, bereits seit ColdFusion Version 5, das die Erstellung von Flash-Charts schon vor einigen Jahren über das Tag `<cfchart>` ermöglichte.

Über eine integrierte, Feature-begrenzte Version des Macromedia Flex

Event-Kalender:
So sieht eine fertige Benutzeroberfläche aus.



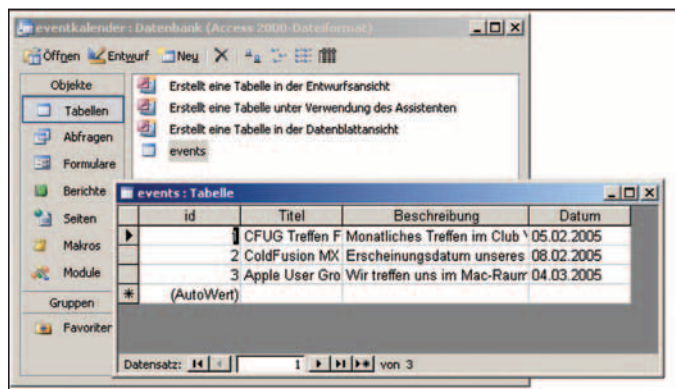
Server wird nun auch erstmals das Erstellen komplexer Benutzeroberflächen beziehungsweise Eingabeformulare ermöglicht. Die Umsetzung wird allerdings dadurch beschränkt, dass auch Flash Forms in ColdFusion zur Übermittlung von Daten an den Server noch immer explizit abgesendet werden müssen und eine direkte Interaktion der verschiedenen Benutzeroberflächenelemente nur in beschränktem Maße möglich ist. Für einfache Anwendungen kann dies allerdings durchaus ausreichend sein. Eine weitere Möglichkeit der Flash-Generierung in ColdFusion MX 7 ist das Erzeugen von FlashPaper-2-Dokumenten mithilfe des `<cfdocument>`-Tag. Das eigentliche Einsatzgebiet von ColdFusion ist jedoch die Bereitstellung einer Datenbankschnittstelle und einer serverseitigen Applikationslogik für Flash-Applikationen.

Von Front- und Backends. Seien es selbstgestrickte, datengetriebene Flash-Animation oder komplexe Flex-Applikationen: Die Trennung zwischen Frontend, der Flash-Applikation selbst, und Backend, also der Serverlogik inklusive Datenbank, sollte immer bestehen bleiben. Es gibt allerdings einige Möglichkeiten, diese Ebenen miteinander zu verschmelzen und beispielsweise vom Frontend aus direkt mit einer Datenbank zu kommunizieren, ohne den Umweg über eine Middleware wie ColdFusion, ASP, Java etc. vorzunehmen. Macromedia regt an einigen Stellen bedauerlicherweise zu einer solchen Programmierpraxis an. Aus verschiedenen Gründen der Datensicherheit, Wartbarkeit von Code, Portabilität und Performanz ist dies absolut nicht empfehlenswert!

ColdFusion bietet sich nicht nur als sichere Schnittstelle zur Datenbank an, sondern kann auch immer da zum Einsatz kommen, wo komplexe Berechnungen oder Datenumformungen auf dem Client aufgrund der schwachen Rechenleistung des Flash-Plug-ins nicht möglich oder zumindest nicht sinnvoll sind. Außerdem sollten sämtliche sicherheitsrelevanten Vorgänge ausschließlich auf dem Server stattfinden, da Sie immer davon ausgehen müssen, dass Daten auf einem Client oder auf dem Weg vom Client zum Server manipuliert werden können.

Für unsere Beispielapplikationen werden wir immer wieder auf den gleichen Backend-Code zugreifen. Dieser Code steckt verpackt in einer ColdFusion Component und bietet eine Schnittstelle zu einer Datenbank an. Durch die Verwendung von ColdFusion Components ist es möglich, gleichzeitig ColdFusion-.cfm-Seiten als Backend als auch allen möglichen anderen Plattformen als Web Service zu dienen. So können ColdFusion Components über Flash Remoting und das AMF-Format aus Flash-Applikationen direkt angesprochen werden. Alternativ hierzu lassen sich dieselben Components als Web Service aus Flash, Flex oder anderen Anwendungen heraus verwenden. Theoretisch ist es sogar möglich, wenn auch nicht wirklich empfehlenswert, das von einer ColdFusion Component generierte XML in Flash von Hand einzulesen und zu deserialisieren.

Der Event-Kalender. Die Möglichkeiten von Flash als Basis für Rich Internet Applications werden wir anhand einer



Kompakt: Eine einfache Tabelle enthält alle Daten für den Event-Kalender.

zu programmieren. Legen Sie dazu ein Verzeichnis mit dem Namen *eventkalender* unterhalb des Webroot-Verzeichnisses (z.B. *c:/cfusionmx7/wwwroot*) Ihres ColdFusion-Servers an und erstellen Sie eine neue Datei mit dem Namen *event.cfc*. Schreiben Sie in die Datei *event.cfc* den Code aus Listing 1.

ColdFusion Components stellen eine Sammlung einzelner Methoden dar. Jede Methode wird mit einem `<cffunction>`-Tag beschrieben. Hier legen wir zwei Methoden mit den Namen *getEvents* und *insertEvents* an. Diese Methoden können sowohl von normalen ColdFusion-*.cfm*-Seiten aus aufgerufen werden als auch als Web Service, sofern das Attribut *access="remote"* gesetzt ist.

Die Methode *getEvents* dient zum Auslesen der Event-Daten aus der Datenbank. Die Methode *insertEvents* fügt neue Event-Daten zur Datenbank hinzu. Mithilfe des `<cfquery>`-Tag werden Datenbankinteraktionen getätigt. Das erste `<cfquery>`-Tag in der Methode *getEvents* liest alle Werte aus der Datenbank aus. Über das `<cfreturn>`-Tag werden diese Werte als Query-Objekt als Ergebnis zurückgegeben.

Im Gegensatz zur Methode *getEvents*, die völlig ohne Argumente aufgerufen wird, braucht die Methode *insertEvents* einige Daten, um ihre Funktion erfüllen zu können. Die genauen Daten, die beim Aufruf der Methode benötigt werden, legen Sie über das Tag `<cfargument>` fest.

kleinen Beispielapplikation präsentieren, die unter Einsatz verschiedener Frontend-Technologien umgesetzt wird. Als Beispiel haben wir einen Event-Kalender gewählt. Er soll Benutzer einer Website interaktiv über geplante Events informieren und über eine integrierte Administratorfunktion die Verwaltung dieser Events komfortabel ermöglichen.

In diesem ersten Teil der Serie entwickeln wir das Backend, das für die folgenden Teile als Basis der Entwicklung der verschiedenen Frontends dienen soll. Um die Artikel nachzuvollziehen, benötigen Sie keine Kenntnisse in ColdFusion: Alle notwendigen Schritte zur Einrichtung von ColdFusion als Backend für die Flash-Applikationen werden erklärt. Um jedoch die tieferen Zusammenhänge dessen zu verstehen, was im Rahmen des Artikels umgesetzt wird, bietet es sich an, an geeigneter Stelle einiges Grundlagenwissen über ColdFusion zu vertiefen. Alle Beispiele beziehen sich auf die aktuelle Version ColdFusion MX 7.

Datenbank. Die Grundlage des Event-Planers stellt eine einfache Datenbank dar, die sich in Form einer Microsoft Access-Datei auf der Heft-CD befindet. Da wir uns hauptsächlich auf Frontend-Technologie sowie den Datentransfer zwischen Flash und ColdFusion konzentrieren werden, ist die Datenbank mit einer einzigen Tabelle so einfach wie möglich gehalten. Der Transfer der Datenbank auf andere Datenbankplattformen sollte also kein Problem darstellen.

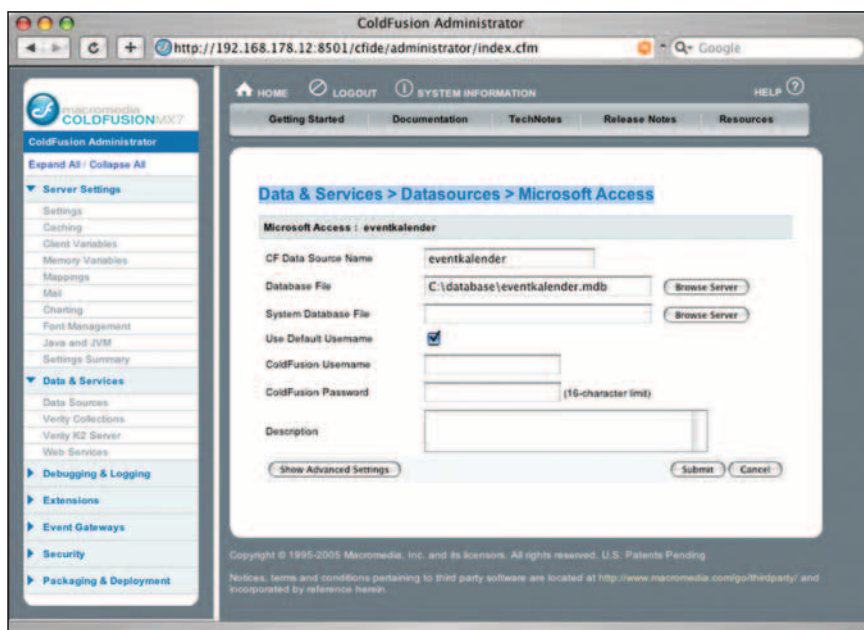
Einrichtung der Datenbank über den ColdFusion-Administrator.

Um auf die Datenbank über ColdFusion zugreifen zu können, muss sie zunächst über den ColdFusion-Administrator eingerichtet werden. Öffnen Sie hierzu den ColdFusion-Administrator (START

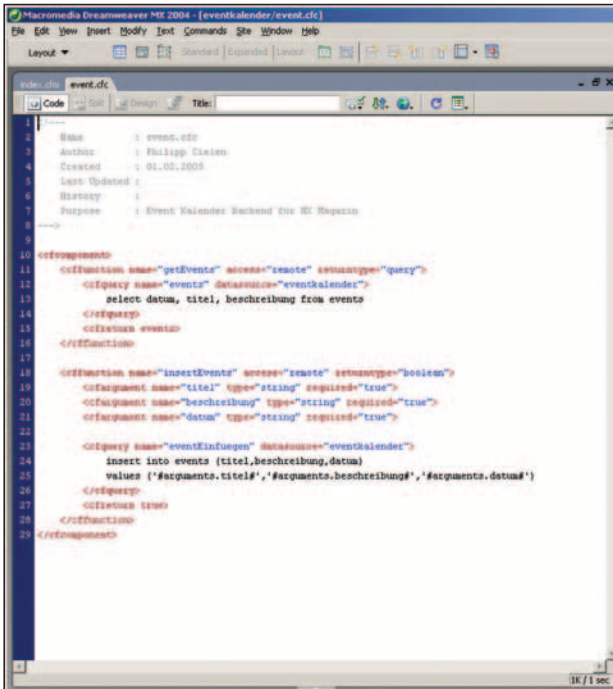
| PROGRAMME | MACROMEDIA | COLD-FUSION MX 7 | ADMINISTRATOR) und wählen Sie unter dem Punkt DATA & SERVICES/DATASOURCES Microsoft Access als Datenquelle aus. Als Name für die Datenquelle wählen Sie *eventkalender*. Kopieren Sie die Access-Datei *eventkalender.mdb* von der Heft-CD auf Ihre Festplatte und geben Sie im folgenden Dialogfenster den Pfad zu der Datei an. Die Datei ist als Datenbank verwendbar, auch wenn Sie kein Microsoft Access auf Ihrem Rechner installiert haben. Setzen Sie ein Häkchen hinter das Feld *Use Default Username*. Nach erfolgreicher Einrichtung der Datenquelle können Sie nun über das ColdFusion-Tag `<cfquery datasource="eventkalender">` auf diese Datenbank zugreifen.

ColdFusion Component event.cfc.

Nach Einrichtung der Datenbank können wir daran gehen, die ColdFusion Component für den Event-Kalender



ColdFusion-Administrator: Einrichten der Datenbank als Datenquelle.



```

1 <!--
2 Name : event.cfc
3 Author : Philipp Cielen
4 Created : 01.02.2005
5 Last Updated :
6 Status :
7 Purpose : Event Kalender Backend für MX Magazin
8 -->
9
10 <!--arguments-->
11 <cffunction name="getEvents" access="remote" returntype="query">
12     <cfquery name="events" datasource="eventkalender">
13         select datum, titel, beschreibung from events
14     </cfquery>
15     <cfreturn events>
16 </cffunction>
17
18 <cffunction name="insertEvents" access="remote" returntype="boolean">
19     <cfargument name="titel" type="string" required="true">
20     <cfargument name="beschreibung" type="string" required="true">
21     <cfargument name="datum" type="string" required="true">
22
23     <cfquery name="eventEinfuegen" datasource="eventkalender">
24         insert into events (titel,beschreibung,datum)
25         values ('#arguments.titel#','#arguments.beschreibung#','#arguments.datum#')
26     </cfquery>
27     <cfreturn true>
28 </cffunction>
29 <!--arguments-->

```

So sieht's aus:
ColdFusion
Component
event.cfc.

Adresse <http://COLDFUSIONROOT/eventkalender/event.cfc?wsdl> abrufbar.

Ein weiteres sehr nützliches Feature von ColdFusion Components ist ihre automatische Selbstdokumentation. Unter <http://COLDFUSIONROOT/eventkalender/event.cfc> können Sie auf die Beschreibung der soeben erstellten ColdFusion Component zugreifen (dazu ersetzen Sie *COLDFUSIONROOT* mit dem Pfad zu ihrem ColdFusion-Wurzelverzeichnis, beispielsweise *localhost* oder *localhost:8500*).

Sie haben nun die Grundlagen gelegt, um weiterführende Programmierungen vorzunehmen. Im nächsten Teil des Artikels erfahren Sie, wie Sie die soeben erstellte ColdFusion Component von einem Flash-Frontend aus ansprechen.

Das Attribut *required* gibt an, ob der zu übergebende Wert zwingend benötigt wird, das Attribut *type* den erwarteten Datentyp. Über das Tag `<cfquery>` werden in dieser Methode die übergebenen Werte für Titel, Beschreibung und

Datum in die Datenbank geschrieben. Als Rückgabewert wird zur Bestätigung der boolsche Wert *true* zurückgegeben.

Diese beiden Methoden bilden das Grundgerüst für die darauf basierenden Flash-Frontends. Typisch für ColdFusion ist, dass sich mit sehr wenig Codezeilen und einfachem, logischem Code sehr komplexe Funktionalitäten abbilden lassen. Der für die Verwendung als Web Service benötigte XML-Code wird von ColdFusion automatisch generiert. Das benötigte WSDL-File zur Beschreibung des Web Service ist unter der HTTP-

Listing 1

```

<cfcomponent>
  <cffunction name="getEvents" access="remote"
    returntype="query">
    <cfquery name="events" datasource="
      eventkalender">
      select datum, titel, beschreibung from events
    </cfquery>
    <cfreturn events>
  </cffunction>

  <cffunction name="insertEvents"
    access="remote" returntype="boolean">
    <cfargument name="titel" type="string"
      required="true">
    <cfargument name="beschreibung"
      type="string" required="true">
    <cfargument name="datum" type="string"
      required="true">

    <cfquery name="eventEinfuegen"
      datasource="eventkalender">
      insert into events (titel,beschreibung,datum)
      values ('#arguments.titel#','#arguments.
        beschreibung#','#arguments.datum#')
    </cfquery>
    <cfreturn true>
  </cffunction>
</cfcomponent>

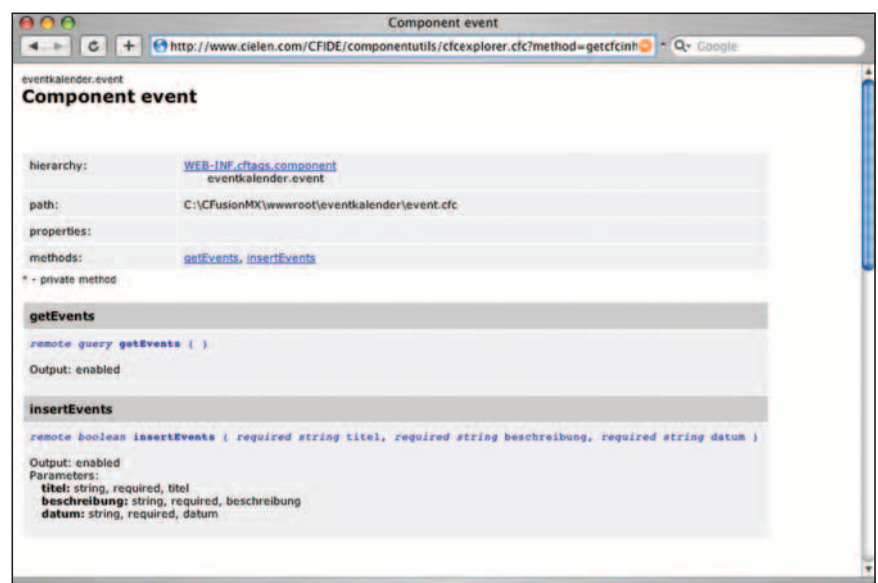
```

Autor Philipp Cielen

Philipp Cielen ist Geschäftsführer der Agentur cielen.com, die sich auf Anwendungsentwicklung im Bereich Macromedia-Technologien spezialisiert hat. Philipp ist Co-Autor des deutschsprachigen Buchs „ColdFusion MX“, Macromedia-zertifizierter Entwickler und leitet die CFUG Frankfurt.



philipp@cielen.com
www.cielen.com/blog/
www.coldflash.de



Automatisch: ColdFusion generiert eine Dokumentation zu jeder ColdFusion Component.