

# FileMaker® Server 11

Anpassad webbpublicering  
med XML och XSLT



© 2007-2010 FileMaker, Inc. Med ensamrätt.

FileMaker, Inc.

5201 Patrick Henry Drive

Santa Clara, Kalifornien 95054, USA

FileMaker är ett varumärke som tillhör FileMaker, Inc. och är registrerat i USA och andra länder. Filmappslogotypen är ett varumärke som tillhör FileMaker, Inc. Alla övriga varumärken tillhör respektive ägare.

FileMakers dokumentation skyddas av lagen om upphovsrätt och det är därför inte tillåtet att mångfaldiga eller distribuera detta dokument utan FileMakers skriftliga medgivande. Dokumentationen får endast användas tillsammans med ett licensierat exemplar av FileMaker-programmet.

Samtliga personer, företag, e-postadresser och URL-adresser som förekommer i exempel är fiktiva och eventuella likheter med verkliga personer, företag, e-postadresser eller URL-adresser är fullständigt oavsiktliga. En lista över medverkande visas i dokumenten. Tillskrivningar som medföljer den här programvaran. Omnämmande av tredjepartsprodukter och URL-adresser förekommer endast i informationssyfte och ska inte tolkas som förslag eller rekommendationer. FileMaker, Inc. tar inget ansvar när det gäller prestandan hos dessa produkter.

Mer information finns på webbplatsen [www.filemaker.com/se](http://www.filemaker.com/se).

Utgåva: 01

# Innehållsförteckning

Om denna handbok	9
<b>Kapitel 1</b>	
<b><i>Introduktion till anpassad webbpublicering</i></b>	
Om Web Publishing Engine	12
Hur en webbpublicerings-förfrågan behandlas	12
Anpassad webbpublicering med PHP	13
Anpassad webbpublicering med XML och XSLT	13
Jämföra PHP för XML och XSLT	14
Anledningar till att välja PHP	14
Anledningar till att välja XML och XSLT	14
<b>Kapitel 2</b>	
<b><i>Om anpassad webbpublicering med XML och XSLT</i></b>	
Skapa dynamiska webbplatser med Web Publishing Engine	15
Anpassad webbpublicering med XML	15
Anpassad webbpublicering med XSLT	16
Om utveckling av FileMaker XSLT-formatmallar	16
Nyckelfunktioner för anpassad webbpublicering med XML och XSLT	16
Krav för webbpublicering	17
Vad som krävs för att publicera en databas med anpassad webbpublicering	17
Vad webbanvändare behöver för att kunna använda en anpassad webbpubliceringslösning	17
Om att ansluta till Internet eller ett intranät	18
Så går du vidare	18
<b>Kapitel 3</b>	
<b><i>Förbereda databaser för anpassad webbpublicering</i></b>	
Aktivera anpassad webbpublicering i en databas	19
Få åtkomst till en skyddad databas	19
Skydda publicerade databaser	20
Webbserversupport för Internet-mediatyper (MIME)	21
Publicera innehållet i containerfält på webben	21
Publicera containerfältobjekt som är lagrade i en databas	21
Publicera containerfältobjekt som är lagrade som filreferenser	21
Hur webbanvändare kan visa containerfältdata	21
FileMaker-manus och anpassad webbpublicering	22
Tips och råd om manus	22
Funktionssätt i egna webbpubliceringslösningar	23
Manustriggers i egna webbpubliceringslösningar	23

## Kapitel 4

### **Introduktion till anpassad webbpublicering med XSLT**

FileMaker XSLT-formatmallar	25
Exempel på hur du kan använda FileMaker XSLT-formatmallar	25
Komma igång med anpassad webbpublicering med XSLT	26
Hur Web Publishing Engine skapar sidor baserade på XML-data och XSLT-formatmallar	26
Allmänna steg för anpassad webbpublicering med XSLT	27
Skapa XSLT-formatmallar för FileMaker med FileMaker XSLT Site Assistant	28
Innan du använder XSLT Site Assistant	28
Starta XSLT Site Assistant	29
Använda XSLT Site Assistant	29
Om formatmallar som skapas i XSLT Site Assistant	29
Använda FileMaker XSLT-formatmallar på webbplatser och i program	30
Felsöka XSLT-formatmallar	31

## Kapitel 5

### **Åtkomst till XML-data med Web Publishing Engine**

Använda anpassad webbpublicering med XML	33
Skillnader mellan Web Publishing Engine och XML-import/export i FileMaker Pro	33
Hur Web Publishing Engine genererar XML-data från en förfrågan	34
Åtkomst till XML-data från Web Publishing Engine	35
URL-syntax för XML-data och containerobjekt	35
URL-syntax för XML-data	36
URL-syntax för FileMaker-containerobjekt i XML-lösningar	37
URL-textkodning	37
Åtkomst till XML-data med Web Publishing Engine	38
Namnuttryck för FileMaker XML	38
Felkoder i FileMaker-databaser	39
Hämta dokumenttypsdefinitionerna för syntaxerna i FileMaker	39
Använda syntaxen fmresultset	39
Beskrivning av element i syntaxen fmresultset	40
Exempel på XML-data i syntaxen fmresultset	42
Använda andra FileMaker XML-syntaxer	43
Beskrivning av element i syntaxen FMPXMLRESULT	43
Exempel på XML-data i grammatikfilen FMPXMLRESULT	44
Beskrivning av element i grammatikfilen FMPXMLLAYOUT	44
Exempel på XML-data i grammatikfilen FMPXMLLAYOUT	46
Om UTF-8-kodade data	47
Efterfråga XML-data med hjälp av frågesträngar i FileMaker	47
Byta layout för ett XML-svar	49
Bearbetning av en XML-förfrågan	49
Bearbeta formatmallar på serversidan och klientsidan	50
Felsöka åtkomst till XML-dokument	51

## Kapitel 6

### **Utveckla XSLT-formatmallar för FileMaker**

Använda XSLT-formatmallar med Web Publishing Engine	53
Om referens till FileMaker-tilläggsfunktionerna för XSLT	54
Om FileMakers startlösning för XSLT	54
URL-syntax för XSLT-formatmallar för FileMaker	54
URL-syntax för FileMaker-containerobjekt i XSLT-lösningar	55
Använda frågesträngar i XSLT-formatmallar för FileMaker	56
Ange en XML-syntax för en XSLT-formatmall för FileMaker	56
Namntryck och prefix för XSLT-formatmallar för FileMaker	57
Använda statistiskt definierade frågekommandon och frågeparametrar	58
Ange textkodning för förfrågningar	59
Ange utdatametod och kodning	60
Koda XSLT-formatmallar	60
Bearbeta XSLT-förfrågningar som inte skickar frågor till FileMaker Server	60
Överföra information mellan formatmallar med hjälp av token	61
Använda FileMaker-tilläggsfunktioner och -tilläggsparametrar för XSLT	61
FileMaker-specifika XSLT-parametrar som anges av Web Publishing Engine	61
Använda frågeinformation i en förfrågan	62
Hämta klientinformation	62
Använda Web Publishing Engines grundläggande URI-parameter	63
Använda den verifierade grundläggande URI-parametern	63
Läsa in ytterligare dokument	64
Använda layoutinformation för en databas i en formatmall	65
Använda innehållsbuffering	65
Lagra information mellan förfrågningar med hjälp av Web Publishing Engine	66
Använda tilläggsfunktionerna för sessioner	67
Skicka e-postmeddelanden från Web Publishing Engine	68
Använda sidhuvudfunktionerna	70
Använda tilläggsfunktioner för cookies	71
Använda tilläggsfunktionerna för stränghantering	72
Jämföra strängar med reguljära Perl 5-uttryck	73
Söka efter värden i ett fält som är formaterat som en kryssruta	73
Använda tilläggsfunktioner för datum, tid och dag	75
Kontrollera felstatus i tilläggsfunktioner	78
Använda loggning	78
Använda bearbetning på serversidan av manusspråk	78
Definiera en tilläggsfunktion	79
Exempel på en tilläggsfunktion	79

## Kapitel 7

### **Iscensätta, testa och övervaka en webbplats**

Iscensätta en anpassad webbpubliceringsplats	83
Testa en anpassad webbpubliceringsplats	84
Exempel på formatmallar som du kan använda när du testar XML-utdata	85
Övervaka webbplatsen	86
Använda webbserveråtkomst och felloggar	86
Använda Web Publishing Engines programlogg	86
Använda felloggen för webbservern	87
Använda webbpubliceringskärnans interna åtkomstloggar	87

## Bilaga A

### **Giltiga namn i frågesträngar**

Frågekommandon och frågeparametrar	89
Riktlinjer för frågekommandon och frågeparametrar	90
FileMaker-databasen Query Strings Reference	90
Syntax för ett fullständigt fältnamn	91
Använda frågekommandon med portalfält	91
Syntax för globala fält	93
Referens för frågekommando	93
Frågekommandot –dbnames (databasnamn)	93
Frågekommandot –delete (radera post)	93
Frågekommandot –dup (duplicera post)	94
Frågekommandot –edit (redigera post)	94
Frågekommandona –find, –findall och –findany (sök efter poster)	94
–findquery (sammanfattad sökning), frågekommando	95
Frågekommandot –layoutnames (layoutnamn)	95
Frågekommandot –new (ny post)	95
–process (bearbeta XSLT-formatmallar)	96
Frågekommandot –scriptnames (manusnamn)	96
Frågekommandot –view (visa layoutinformation)	96
Referens för frågeparameter	96
Frågeparameteren –db (databasnamn)	96
–delete.related (radera portalposter), frågeparameter	97
Frågeparameteren –encoding (kodning av XSLT-förfrågan)	97
Frågeparameteren –field (namn på containerfält)	97
Frågeparameteren fältnamn (namn på annat fält än containerfält)	97
Frågeparameteren fältnamn.op (jämförelseoperator)	98
Frågeparameteren –grammar (syntax för XSLT-formatmallar)	99
Frågeparameteren –lay (layout)	99
Frågeparameteren –lay.response (växla layout vid svar)	99

Frågeparameteren –lop (logisk operator)	100
Frågeparameteren –max (maximalt antal poster)	100
Frågeparameteren –modid (ändrings-ID)	100
Frågeparameteren –query (sammanfattad sökning)	100
Frågeparameteren –recid (post-ID)	101
–relatedsets.filter (filtrera portalposter), frågeparameter	102
–relatedsets.max (begränsa portalposter), frågeparameter	102
Frågeparameteren –script (manus)	102
–script.param (skicka parameter till manus), frågeparameter	103
Frågeparameteren –script.prefind (manus före sökning)	103
–script.prefind.param (skicka parameter till manus före sökning), frågeparameter	103
Frågeparameteren –script.presort (manus före sortering)	104
–script.presort.param (skicka parameter till manus före sökning), frågeparameter	104
Frågeparameteren –skip (hoppa över poster)	104
Frågeparameteren –sortfield (sortera fält)	105
Frågeparameteren –sortorder (sorteringsordning)	105
–stylehref (Style href), frågeparameter	106
Frågeparameteren –styletype (formatmallstyp)	106
–token.[string] (överför värden mellan XSLT-formatmallar), frågeparameter	106

## Bilaga B

### ***Felkoder för anpassad webbpublicering***

Felkodsnummer för FileMaker-databaser	107
Felkodsnummer för Web Publishing Engine	114
Felkodsnummer för FileMaker-tilläggsfunktioner för XSLT	115

### ***Index***

117



# Förord

## Om denna handbok

Den här handboken riktar sig till dig som känner till hur XML och XSLT fungerar, och som kan utveckla webbplatser och skapa databaser med hjälp av FileMaker Pro<sup>®</sup>. Du bör förstå grunderna i databasdesign i FileMaker Pro, och förstå begreppen fält, relationer, layouter, portaler och containrar. Handboken innehåller följande information om anpassad webbpublicering med XML och XSLT på FileMaker Server:

- vad som krävs för att du ska kunna utveckla en anpassad webbpubliceringslösning med XML eller XSLT
- hur du publicerar databaser med XML och XSLT
- vilka webbanvändare som bör ha åtkomst till en anpassad webbpubliceringslösning
- hur du hämtar XML-data från databaser på FileMaker Server
- hur du utvecklar XSLT-formatmallar för FileMaker

**Viktigt!** Du kan hämta dokumentationen för FileMaker i PDF-format från <http://www.filemaker.com/se/support/index.html>. Uppdateringar till det här dokumentet kan också hämtas från den webbplatsen.

Dokumentationen för FileMaker Server innehåller följande information:

<b>Information om</b>	<b>Se</b>
Installera och konfigurera FileMaker Server	<i>FileMaker Server Startbok</i> <i>FileMaker Server - Hjälp</i>
Direkt webbpublicering	<i>FileMaker Handbok för Direkt webbpublicering</i>
Anpassad webbpublicering med PHP	<i>FileMaker Server Anpassad webbpublicering med PHP</i>
Använda PHP Site Assistant	Hjälp för PHP Site Assistant
Anpassad webbpublicering med XML och XSLT	<i>FileMaker Server Anpassad webbpublicering med XML och XSLT</i> (denna bok)
Använda XSLT Site Assistant	XSLT Site Assistant Hjälp
Installera och konfigurera drivrutiner för ODBC och JDBC och använd ODBJ och JDBC	<i>Handbok för FileMaker ODBC och JDBC</i>
Hur FileMaker Server Auto Update kan ladda ner den senaste versionen av ett plugin-program till FileMaker Pro-databasklienter	<i>FileMaker Server 9 Handboken Uppdatera plugin-program</i>



# Kapitel 1

## ***Introduktion till anpassad webbpublicering***

Med FileMaker Server kan du publicera din FileMaker-databas på Internet eller ett intranät på följande sätt.

**Direkt webbpublicering:** Med Direkt webbpublicering publicerar du din databas snabbt och smidigt på webben. Du behöver inte ändra dina databasfiler eller installera ytterligare program – alla som har en kompatibel webbläsare och tillgång till Internet eller ett intranät kan ansluta sig till databasen och visa, redigera, sortera och söka i poster om du ger dem behörighet.

Med Direkt webbpublicering måste värddatorn köra FileMaker Pro eller FileMaker Server. Användargränssnittet liknar FileMaker Pro-programmet. Webbsidorna och formulär som webbanvändaren samverkar med är beroende på de layouter och vyer som definierats i FileMaker Pro-databasen. Mer information finns i FileMaker Handbok för Direkt webbpublicering.

**Statisk publicering:** Om du ändrar dina data väldigt sällan eller om du inte vill att användarna ska ha en direktanslutning till din databas kan du använda dig av statisk publicering. Vid statisk publicering exporterar du data från en FileMaker Pro-databas och skapar en webbsida som du kan anpassa ytterligare med HTML. Webbsidan ändras inte när informationen i din databas ändras och användarna öppnar inte din databas. (När du använder Direkt webbpublicering uppdateras data i ett webbfönster varje gång webbläsaren skickar en begäran till FileMaker Server.) Mer information finns i *FileMaker Handbok för Direkt webbpublicering*.

**Anpassad webbpublicering:** Om du vill ha bättre kontroll över den publicerade databasens utseende och funktionalitet använder du de funktioner för anpassad webbpublicering som finns i FileMaker Server. FileMaker Server, som är värd för de publicerade databaserna, kräver inte att FileMaker Pro är installerat eller igång för att anpassad webbpublicering ska vara tillgänglig.

Med Anpassad webbpublicering kan du:

- integrera databasen med en annan webbplats
- avgöra hur användarna ska få tillgång till data
- kontrollera hur data ska visas i webbläsare.

FileMaker Server innehåller två tekniker för anpassad webbpublicering:

- Anpassad webbpublicering med PHP: Använd FileMaker Api för PHP, med ett objektorienterat PHP-gränssnitt för FileMaker Pro-databaser, för att integrera dina FileMaker-data i ett PHP-webbprogram. Du kan använda PHP Site Assistant för att skapa en komplett PHP-webbplats, eller koda PHP-webbsidor själv.
- Anpassad webbpublicering med XML och XSLT:
  - Använd XML-datapublicering för att utbyta FileMaker-data med andra webbplatser och program.
  - Använd serverbehandlade XSLT-formatmallar för att integrera eventuella delmängder med FileMaker-data med andra webbplatser, med andra programsystem och med anpassade program. Du kan använda XSLT Site Assistant för att generera XSLT-formatmallar, eller koda formatmallarna själv.

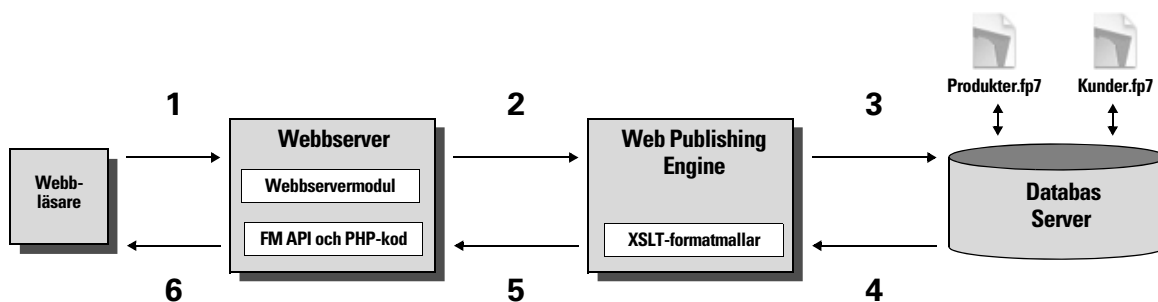
## Om Web Publishing Engine

För att stödja direkt webbpublicering och anpassad webbpublicering använder FileMaker Server en uppsättning programvarukomponenter som kallas för *FileMaker Server Web Publishing Engine*. Web Publishing Engine hanterar samspelet mellan en webbanvändares webbläsare, din webserver och FileMaker Server.

**Anpassad webbpublicering med XML och XSLT:** Web Publishing Engine fungerar som en XSLT-processor och genererar utdata som HTML, XML eller text (exempelvis vCards (elektroniska visitkort)) till webbservern, som i sin tur skickar dessa utdata till webbläsaren. Webbanvändare får åtkomst till din anpassade webbpubliceringslösning genom att klicka på en HREF-länk eller skriva en URL-adress till webbservern och en FileMaker-frågesträng. URL-adressen kan gå till XML-data eller en XSLT-formatmall. Web Publishing Engine returnerar de XML-data som angavs i frågesträngen eller resultatet från den angivna XSLT-formatmallen.

**Anpassad webbpublicering med PHP:** När en webbanvändare ansluter till din anpassade webbpubliceringslösning ansluter PHP på FileMaker Server till Web Publishing Engine och svarar genom FileMaker Api för PHP.

Använda FileMaker Servers Web Publishing Engine för anpassad webbpublicering



## Hur en webbpubliceringsförfrågan behandlas

1. En förfrågan skickas från en webbläsare eller ett program till webbservern.
2. Webbservern vidarebefordrar förfrågan genom FileMakers webservermodul till Web Publishing Engine.
3. Web Publishing Engine efterfrågar data på databasen som databasservern är värd för.
4. FileMaker Server skickar efterfrågade FileMaker-data till Web Publishing Engine.

5. Web Publishing Engine konverterar FileMaker-data för att svara på förfrågan.
  - För PHP-förfrågningar svarar Web Publishing Engine på API-förfrågan.
  - För XML-förfrågningar skickar Web Publishing Engine XML-data direkt till webbservern.
  - För XSLT-förfrågningar använder Web Publishing Engine en XSLT-formatmall som formaterar eller konverterar XML-data och genererar utdata som HTML-sidor, ett XML-dokument eller text till webbservern.
6. Webbservern skickar utdata tillbaka till webbläsaren eller programmet.

**Viktigt!** Säkerheten är mycket viktig när du publicerar information på webben. Läs igenom säkerhetsriktlinjerna i *FileMaker Pro Handboken* som kan hämtas som PDF-fil på <http://www.filemaker.com/se/support/index.html>.

## Anpassad webbpublicering med PHP

FileMaker Api för PHP ger ett objektorienterat PHP-gränssnitt för FileMaker-databaser. FileMaker Api för PHP gör att man kan få åtkomst till både uppgifter och logik som har lagrats i en FileMaker Pro-databas och publicera dem på Internet, eller exportera dem till andra program. API stöder också komplexa och sammansatta sökkommandon för att extrahera och filtrera data som har lagrats i FileMaker Pro-databaser.

PHP utformades ursprungligen som ett procedurmässigt programmeringsspråk, och har utvecklats som ett objektorienterat webbutvecklingsspråk. PHP ger programspråksfunktionalitet för att skapa praktiskt taget alla typer av logik på en webbsida. Till exempel kan du använda villkorliga logikkonstruktioner för att kontrollera sidgenerering, datarouting eller arbetsflöde. PHP ger även möjlighet till webbplatsadministration och säkerhet.

Dessutom kan du använda FileMaker PHP Site Assistant för att skapa PHP-kod som innehåller alla nödvändiga förutsättningar och funktioner för att få korrekt åtkomst till data i en FileMaker Pro-databas. PHP Site Assistant genererar en flersidig webbplats som gör att webbanvändarna kan söka i en databas, visa en lista över poster, granska poster, sortera poster, lägga till poster, redigera poster, duplicera poster, radera poster och visa en sammanfattningsrapport. FileMaker-utvecklare som har liten erfarenhet av PHP kan använda PHP Site Assistant för att generera en komplett PHP-webbplats. PHP-utvecklare som har liten erfarenhet av FileMaker kan använda PHP Site Assistant för att förstå FileMaker Api för PHP-objekt och metoder.

## Anpassad webbpublicering med XML och XSLT

FileMaker anpassad webbpublicering med XML gör att du kan skicka förfrågningar till en FileMaker Pro-databas som FileMaker Server är värd för, och visa, ändra eller hantera resulterande data. Med en HTTP-förfrågan som innehåller rätt frågekommandon och -parametrar kan du hämta FileMaker-data som ett XML-dokument. Du kan därefter exportera XML-informationen till andra program eller tillämpa en XSLT-formatmall på XML-informationen.

FileMaker anpassad webbpublicering med XSLT gör att du kan omvandla, filtrera eller formatera XML-data för webbläsare eller andra applikationer. Du kan:

- Du kan använda en XSLT-formatmall och omvandla informationen mellan FileMaker XML-syntax och en annan XML-syntax för användning i andra program eller andra databaser.
- Filtrera informationen genom att välja vilka databasfält som formatmallen ska publicera.
- Utforma informationen på en webbsida och ange hur webbanvändaren ska kunna samverka med informationen.

Web Publishing Engine använder formatmallarna för att hämta data från en FileMaker-databas varje gång en webbanvändare skickar en HTTP-förfrågan och en URL-adress som hänvisar till någon av dina XSLT-formatmallar. Web Publishing Engine transformerar och formaterar XML-informationen med hjälp av en formatmall och genererar HTML-resultatsidan som webbanvändaren kan arbeta med.

Dessutom är FileMaker Site Assistant ett program som du kan använda när du skapar grundläggande XSLT-formatmallar som kan användas som utgångspunkt för anpassad webbpublicering med XSLT. XSLT Site Assistant skapar formatmallar för sidor som söker igenom databasen, granskar en post i taget, listar posterna i databasen, sorterar poster, lägger till poster, redigerar poster, duplicerar poster, raderar poster och visar en sammanfattningsrapport.

## Jämföra PHP för XML och XSLT

Följande avsnitt ger riktlinjer för hur du avgör vilken lösning som passar bäst för din webbplats.

### Anledningar till att välja PHP

- PHP är ett mer kraftfullt, objektorienterat procedurmässigt manusspråk och är relativt enkelt att lära sig. Det finns många tillgängliga resurser för utbildning, utveckling och support.
- FileMaker Api för PHP gör att man kan få åtkomst till uppgifter och logik som har lagrats i en FileMaker Pro-databas och publicera dem på Internet, eller exportera dem till andra program.
- Med PHP kan man använda villkorslogik för att kontrollera sidkonstruktionen eller flödet.
- PHP ger programspråksfunktionalitet för att skapa många typer av logik på en webbsida.
- PHP är ett av de populäraste webbmanusspråken.
- PHP är ett språk med öppen källkod, som finns på <http://php.net>.
- PHP ger åtkomst till ett stort utbud av komponenter från tredje part som du kan integrera i dina lösningar.

**Obs!** Mer information om anpassad webbpublicering med PHP finns på *FileMaker Server Anpassad webbpublicering med PHP*.

### Anledningar till att välja XML och XSLT

- FileMaker XML frågeparametersyntax är utformat för interaktion mellan databaser, vilket underlättar lösningsutveckling.
- XML och XSLT är W3C-standarder.
- XML är ett format som kan läsas av både datorer och människor och som stöder Unicode, vilket gör att data kan kommuniceras på alla skriftliga språk.
- XML passar bra för att presentera poster, listor och trädstrukturerad data.
- Med XSLT kan du omvandla XML-utdata till strukturerade textdokument som RSS, RTF, vCard.
- Du kan använda XSLT för att omvandla XML-utdata från en syntax till en annan.
- Mallar för det enkelt att tillämpa villkorsformatering på variabla data.
- Du kan använda FMPXMLRESULT-baserade formatmallar för anpassad webbpublicering och för XML-export från FileMaker Pro-databaser.
- FileMaker Server hanterar FileMaker XSLT-formatmallsbehandling, förhindrar obehörig åtkomst till data som kan vara oskyddad med hjälp av XSLT-formatmallar på klientsidan.

# Kapitel 2

## *Om anpassad webbpublicering med XML och XSLT*

### Skapa dynamiska webbplatser med Web Publishing Engine

Med Web Publishing Engine kan du använda anpassad webbpublicering tillsammans med FileMaker Server och använda XML-datapublicering och serverbearbetade XSLT-formatmallar. Anpassad webbpublicering har flera fördelar:

- **Anpassning:** Du kan bestämma vad webbanvändarna ska kunna göra med FileMaker-data och hur informationen visas i webbläsare.
- **Datautbyte:** Genom att använda FileMaker XML kan du utbyta FileMaker-data med andra webbplatser och program.
- **Dataintegrering:** Med XSLT-formatmallar för FileMaker kan du integrera FileMaker-data för andra webbplatser, med andra programsystem och anpassade program. Du kan göra så att informationen ser ut att höra till en annan webbplats i stället för att visa hela FileMaker-layouten i webbläsaren.
- **Säkerhet:** Administratören för FileMaker Server kan separat aktivera eller inaktivera Direkt webbpublicering, XML-webbpublicering och XSLT-webbpublicering för alla databaser som körs på servern. Som ägare till en FileMaker-databas kan du styra webbanvändarnas åtkomst till Direkt webbpublicering, XML-webbpublicering och XSLT-webbpublicering för varje databas.
- **Formatmallar på serversidan:** Bearbetning av XSLT-formatmallar på serversidan förhindrar åtkomst till konfidentiell databasinformation, information som annars kan vara tillgänglig med formatmallar på klientsidan.
- **Styra och filtrera publicerade data:** Med XSLT-formatmallar kan du styra och filtrera informationen och vilken typ av databasinformation du vill publicera, vilket förhindrar obehörig användning av databasen. Du kan också dölja metadata, till exempel databas- och fältnamn.
- **Öppen standard:** Du har bättre åtkomst till verktyg, resurser och erfaren personal för anpassade webbpubliceringslösningar. Om du känner till standard-XML eller -XSLT kan du genast sätta igång att utveckla lösningar, så snart du har lärt dig ett par detaljer om anpassad webbpublicering med XML, till exempel URL-syntaxen och vilka frågeparametrar som ska användas.

### Anpassad webbpublicering med XML

Anpassad webbpublicering med XML medger att du hämtar data från FileMaker-databaser och enkelt använder dessa data i andra utdataformat. Med en HTTP-förfrågan som innehåller rätt frågekommandon och -parametrar kan du hämta FileMaker-data som ett XML-dokument. Du kan därefter använda XML-informationen i andra program eller tillämpa en XSLT-formatmall på XML-informationen. Se kapitel 5, ”Åtkomst till XML-data med Web Publishing Engine”.

## Anpassad webbpublicering med XSLT

Med anpassad webbpublicering med XSLT kan du omvandla, filtrera och formatera XML-data och använda dessa data i en webbläsare och i andra program. Du kan använda en XSLT-formatmall och omvandla informationen mellan FileMaker XML-syntax och en annan XML-syntax för användning i ett annat program eller i en annan databas. Du kan filtrera informationen genom att välja vilka databasfält som formatmallen ska publicera. Du kan formatera hur informationen visas på en webbsida och ange hur webbanvändaren ska kunna samverka med informationen. Se kapitel 4, "Introduktion till anpassad webbpublicering med XSLT".

Web Publishing Engine använder formatmallarna för att dynamiskt hämta data från en FileMaker-databas varje gång en webbanvändare skickar en HTTP-förfrågan och en URL-adress som hänvisar till någon av dina XSLT-formatmallar. Web Publishing Engine transformerar och formaterar XML-informationen med hjälp av en formatmall och genererar HTML-resultatsidan som webbanvändaren kan arbeta med.

Mer information om hur du använder anpassad webbpublicering med XML och XSLT finns på <http://www.filemaker.com/se/support/index.html>.

## Om utveckling av FileMaker XSLT-formatmallar

FileMaker Server innehåller ett verktyg för utveckling av XSLT-formatmallar: FileMaker XSLT Site Assistant är ett program du kan använda när du skapar grundläggande XSLT-formatmallar för att användas som utgångspunkt för anpassad webbpublicering med XSLT. Genom att använda Site Assistant kan du lära dig hur XSLT-formatmallar för FileMaker är uppbyggda. Därefter kan du bearbeta formatmallarna med hjälp av egna verktyg för utveckling av XSLT-formatmallar. Se "Skapa XSLT-formatmallar för FileMaker med FileMaker XSLT Site Assistant" på sidan 28.

**Obs!** FileMaker Server stöder XSLT 1.0 enligt den definition som har gjorts av World Wide Web Consortium. Alla redigeringsverktyg du använder för XSLT måste skapa filer som följer standarden XSLT 1.0.

## Nyckelfunktioner för anpassad webbpublicering med XML och XSLT

FileMaker Server Anpassad webbpublicering med XML och XSLT innehåller flera viktiga funktioner:

- Databaserna finns på FileMaker Server, så FileMaker Pro behöver inte vara igång.
- Du kan använda XSLT-formatmallsbearbetning på serversidan, vilket är säkrare än formatmallsbearbetning på klientsidan.
- Du kan använda bearbetning på serversidan av JavaScript i XSLT-formatmallar. Mer information finns i "Använda bearbetning på serversidan av manusspråk" på sidan 78.
- Du kan förhindra obehörig användning av frågekommandon och frågeparametrar med FileMaker XSLT-formatmallen genom att statistiskt definiera frågekommandona, frågeparametrarna och de värden du vill ska användas när XML-data efterfrågas. Se "Använda statistiskt definierade frågekommandon och frågeparametrar" på sidan 58.
- Precis som i FileMaker Pro är åtkomsten till data, layouter och fält baserad på användarkontots inställningar som anges i databasens behörighetsinställningar. Web Publishing Engine har också ytterligare flera säkerhetsförbättringar. Se "Skydda publicerade databaser" på sidan 20.
- Webbanvändare kan utföra komplexa manus med många steg. FileMaker stöder cirka 70 manussteg för anpassad webbpublicering. Se avsnittet "FileMaker-manus och anpassad webbpublicering" på sidan 22.

- Du kan skicka ett parametervärde till ett FileMaker-manus. Mer information finns i avsnitten ”–script.param (skicka parameter till manus), frågeparameter” på sidan 103, ”–script.prefind.param (skicka parameter till manus före sökning), frågeparameter” på sidan 103 och ”–script.presort.param (skicka parameter till manus före sökning), frågeparameter” på sidan 104.
- Med den nya syntaxen fmresultset i XML får du åtkomst till fält efter namn och kan hantera relatedset (portaldata).
- Genom att använda sessionsfunktioner i XSLT-formatmallar kan du lagra en webbanvändares information och transaktioner i serverhanterade sessioner.
- Om du vill ha åtkomst till informationen i en databas måste du ange en layout. Se bilaga A, ”Giltiga namn i frågesträngar”.
- Varje webbanvändare kan ha ett unikt globalt fältvärde som finns kvar så länge som sessionen varar. Allmän information om fält för global lagring finns i FileMaker Pro Hjälp. Information om hur du använder fält för global lagring med anpassad webbpublicering finns i ”Syntax för globala fält” på sidan 93.

## Krav för webbpublicering

### Vad som krävs för att publicera en databas med anpassad webbpublicering

Om du ska publicera databaser med anpassad webbpublicering med XML eller XSLT, behöver du följande:

- en FileMaker Server-installation som innehåller
  - en webbserver, antingen Microsoft IIS (Windows) eller Apache (Mac OS).
  - FileMaker Server-databas, aktiverad för anpassad webbpublicering
  - Web Publishing Engine, installerad och konfigurerad
- en eller fler FileMaker Pro-databaser på FileMaker Server
- IP-adressen eller domännamnet på värden där webbservern körs
- en webbläsare och åtkomst till webbservern så att du kan utveckla och testa den egna webbpubliceringslösningen

Mer information finns i *FileMaker Server startbok*.

### Vad webbanvändare behöver för att kunna använda en anpassad webbpubliceringslösning

Webbanvändare som vill ha åtkomst till en anpassad webbpubliceringslösning som använder XML eller XSLT behöver följande:

- en webbläsare
- åtkomst till Internet eller ett intranät och webbservern
- IP-adressen eller domännamnet på värden där webbservern körs

Om databasen är lösenordsskyddad måste webbanvändare även uppge ett användarnamn och ett lösenord för ett databaskonto

## Om att ansluta till Internet eller ett intranät

När du publicerar databaser på Internet eller ett intranät måste värddatorn köra FileMaker Server och de databaser du vill dela måste köras och vara tillgängliga. Dessutom gäller följande:

- Publicera din databas på en dator med fast Internet- eller intranätanslutning. Det går att publicera databaser utan att ha en fast anslutning, men dessa blir bara tillgängliga för webbanvändarna när din dator är ansluten till Internet eller ett intranät.
- Värddatorn för den webbserver som är en del av FileMaker Server-driftsättningen måste ha en dedicerad *statisk* (permanent) IP-adress eller ett domännamn. Om du ansluter till Internet via en Internetleverantör (ISP) kan din IP-adress vara *dynamiskt allokerad* (dvs. olika varje gång du ansluter). Med en dynamisk IP-adress blir det svårare för webbanvändarna att hitta dina databaser. Om du inte är säker på vilken typ av anslutning du ska välja kontaktar du din Internetleverantör eller nätverksadministratör.

## Så går du vidare

Här följer några förslag för att komma igång med att utveckla egna webbpubliceringslösningar:

- Aktivera anpassad webbpublicering med hjälp av FileMaker Server Admin Console om du inte redan har gjort det. Se FileMaker Server Hjälp och *FileMaker Server Startbok*.
- I FileMaker Pro öppnar du varje FileMaker-databas som du vill publicera och ser till att databasen har rätt utökad behörighet aktiverad för anpassad webbpublicering. Se "Aktivera anpassad webbpublicering i en databas" på sidan 19.
- Information om hur du får åtkomst till data i FileMaker-databaser med XML finns i kapitel 5, "Åtkomst till XML-data med Web Publishing Engine".
- Information om hur du kommer igång och utvecklar XSLT-formatmallar för FileMaker finns i kapitel 4, "Introduktion till anpassad webbpublicering med XSLT".

# Kapitel 3

## ***Förbereda databaser för anpassad webbpublicering***

Innan du kan använda anpassad webbpublicering med en databas måste du förbereda databasen och skydda den från obehörig åtkomst.

### **Aktivera anpassad webbpublicering i en databas**

Du måste aktivera anpassad webbpublicering i varje databas som du vill publicera. Du kan för varje databas aktivera anpassad webbpublicering med XML eller anpassad webbpublicering med XSLT eller också kan du aktivera båda teknikerna i varje databas. Om du inte aktiverar båda eller en av dessa tekniker i databasen kommer användarna inte att ha åtkomst till databasen med anpassad webbpublicering, även om databasen ligger på FileMaker Server, som stöder en Web Publishing Engine.

Så här aktiverar du anpassad webbpublicering för en databas:

1. Starta FileMaker Pro och öppna databasen som du vill publicera med hjälp av ett konto som har behörighetsuppsättningen Full åtkomst. Du kan också öppna databasen med hjälp av ett konto som har inställningen Hantera utökad behörighet i behörighetsuppsättningen.
2. Tilldela den ena eller båda dessa utökade behörigheter till en eller flera behörighetsuppsättningar:
  - Om du vill tillåta anpassad webbpublicering med XML anger du följande nyckelord: `fmxml`
  - Om du vill tillåta anpassad webbpublicering med XSLT anger du följande nyckelord: `fmxmlt`Sedan FileMaker Pro 8 definieras nyckelorden `fmxml` och `fmxmlt` på fliken Utökade behörigheter.
3. Koppla de behörighetsuppsättningar som innehåller de utökade behörigheterna för anpassad webbpublicering till ett eller fler konton eller till kontot Admin eller Gäst.

**Obs!** När du definierar kontonamn och lösenord för lösningar för anpassad webbpublicering använder du utskrivbara ASCII-tecken, till exempel, a-z, A-Z och 0-9. För att göra kontonamn och lösenord säkrare inkluderar du skiljetecken, t.ex. `!"` och `%"` men inkluderar inte kolon. Information om hur du konfigurerar konton finns i FileMaker Pro Hjälpsida.

### **Få åtkomst till en skyddad databas**

När webbanvändare använder en anpassad webbpubliceringslösning för att få åtkomst till en databas kan de bli ombudda att ange kontoinformation. Om Gästkontot för databasen är inaktiverat eller inte har en behörighetsuppsättning med utökad behörighet för anpassad webbpublicering, använder Web Publishing Engine grundläggande HTTP-verifiering för verifiering av webbanvändarna. I webbanvändarens webbläsare visas dialogrutan för grundläggande HTTP-verifiering där användaren kan ange användarnamn och lösenord för ett konto som har utökad behörighet för anpassad webbpublicering.

I följande lista sammanfattas processen som sätts igång när en webbanvändare använder lösningen anpassad webbpublicering för att få åtkomst till en databas:

- Om du inte har angett något lösenord för ett konto behöver webbanvändaren bara ange kontonamnet.

- Om Gästkontot är inaktiverat måste användarna ange kontonamn och lösenord när de öppnar databasen. Kontot måste ha utökad behörighet för anpassad webbpublicering aktiverad.
- Om Gästkontot är aktiverat och har en behörighetsuppsättning som innehåller utökad behörighet för anpassad webbpublicering, kan alla webbanvändare automatiskt öppna databasen med den behörighet som Gästkontot har. Om Gästkontot har utökad behörighet för anpassad webbpublicering gäller följande:
  - Webbanvändare behöver inte ange kontonamn eller lösenord när de öppnar en fil.
  - Alla webbanvändare loggar automatiskt in med gästkontot och får den behörighet som hör till gästkontot. Du kan låta användare byta inloggningskonto från en webbläsare med manussteget Logga in igen (exempelvis om de ska växla från Gästkontot till ett annat konto som ger tillgång till fler funktioner).
  - Standardbehörigheten för gästkonton ger endast läsbehörighet. Du kan ändra standardbehörigheten, även den utökade behörigheten, för detta konto. Se FileMaker Pro Hjälp.

**Obs!** Som standardinställning kan webbanvändare inte ändra sina egna kontolösenord från en webbläsare. Du kan bygga in denna funktion för en databas med hjälp av manussteget Ändra lösenord, som gör att webbanvändarna kan ändra sina lösenord från webbläsaren. Se FileMaker Pro Hjälp.

## Skydda publicerade databaser

När du använder anpassad webbpublicering med XML eller XSLT kan du begränsa vilka som ska ha åtkomst till din publicerade databas.

- Tilldela lösenord till databaskonton som används för anpassad webbpublicering.
- Aktivera anpassad webbpublicering med XML eller XSLT endast i behörighetsuppsättningar för konton som du vill ska ha åtkomst till den publicerade databasen.
- Du kan aktivera eller inaktivera olika typer av anpassade webbpubliceringstekniker för en enskild databas genom att ange utökade behörighet.
- Aktivera eller inaktivera en typ av teknik för Anpassad webbpublicering för alla Lösningar för anpassad webbpublicering i Web Publishing Engine med FileMaker Server Admin Console. Se FileMaker Server Hjälp.
- Konfigurera webbservern för att begränsa vilka IP-adresser som ska kunna använda databasen via Web Publishing Engine. Du kan exempelvis ange att endast webbanvändare från IP-adress 192.168.100.101 ska ha tillgång till databaserna. Information om hur du begränsar åtkomst från IP-adresser finns i dokumentationen till webbservern.
- Använd SSL-kryptering (Secure Sockets Layer) för kommunikationen mellan webbservern och webbanvändarnas webbläsare. Vid SSL-kryptering konverteras informationen som utväxlas mellan webbservrar och klienter till obegriplig information med matematiska formler, så kallade chiffer. Dessa chiffer används när informationen konverteras tillbaka till begripliga data med hjälp av krypteringsnycklar. Information om hur du aktiverar och konfigurerar SSL finns i dokumentationen till webbservern.

Mer information om hur du skyddar din databas finns i *FileMaker Pro 9 Handbok* som finns som PDF-fil på <http://www.filemaker.com/se/support/index.html>.

## Webbservernsupport för Internet-mediatyper (MIME)

Webbservern fastställer stöd för de aktuella MIME-typer (Multipurpose Internet Mail Extensions) som har registrerats för Internet. Web Publishing Engine ändrar inte en webbserverns MIME-stöd. Mer information finns i dokumentationen till webbservern.

## Publicera innehållet i containerfält på webben

Innehållet i ett containerfält, till exempel en bildfil, kan antingen lagras i en FileMaker-databas eller som en filreferens med en relativ sökväg.

**Obs!** Web Publishing Engine stöder inte direktuppspelning av filmfiler. Webbanvändarna måste hämta hela filmfilen innan de kan visa filmen.

## Publicera containerfältobjekt som är lagrade i en databas

Om ett containerfält innehåller de faktiska filerna i FileMaker-databasen behöver du inte göra något med innehållet i containerfältet, under förutsättning att FileMaker Server fungerar som värd för databasfilen och den är tillgänglig på FileMaker Server. Se ”URL-syntax för FileMaker-containerobjekt i XML-lösningar” på sidan 37, och ”URL-syntax för FileMaker-containerobjekt i XSLT-lösningar” på sidan 55.

## Publicera containerfältobjekt som är lagrade som filreferenser

Om ett container-fält lagrar filreferenser istället för själva filerna ska du följa dessa steg när du vill publicera objekten i container-fälten.

**Obs!** Alla QuickTime-filmer lagras i containerfält som en referens.

Så här publicerar du containerfältobjekt som är lagrade som filreferenser:

1. Lagra containerobjektfilerna i webbmappen inuti FileMaker Pro-mappen.
2. Starta FileMaker Pro, sätt in objekten i containerfältet och välj Spara endast länk till filen.
3. Kopiera eller flytta refererade objektfiler i mappen Web till samma relativa sökväg i rotmappen i webbserverprogrammet.
  - Med IIS flyttar du filerna till: <enheten>\inetpub\wwwroot
  - Med Apache flyttar du filerna till: /Bibliotek/WebServer/Documents

**Obs!** För att containerobjekt ska kunna lagras som filreferenser måste webbservern kunna hantera MIME-typerna för de typer av filer du vill använda, till exempel filmer. Mer information finns i dokumentationen till webbservern.

## Hur webbanvändare kan visa containerfältdata

När du publicerar en databas på webben med Web Publishing Engine kan webbanvändare arbeta med data i containerfält på följande begränsade sätt:

- Webbanvändare kan inte spela upp ljud eller visa OLE-objekt i ett containerfält. I stället visas en bild.
- Webbanvändare kan inte ändra eller lägga till innehåll i ett containerfält. Webbanvändare kan inte använda containerfält för att överföra data till databasen.
- Om databasen innehåller bilder som inte är i Gif- eller Jpeg-format skapar Web Publishing Engine en temporär Jpeg-bild när bildinformationen efterfrågas av en webbläsare.

## FileMaker-manus och anpassad webbpublicering

Funktionen Hantera manus i FileMaker Pro kan automatisera uppgifter som utförs ofta eller kombinera flera uppgifter. Genom att använda FileMaker-manus tillsammans med anpassad webbpublicering kan webbanvändare utföra flera uppgifter eller en serie uppgifter.

FileMaker stöder över 75 manussteg för anpassad webbpublicering. Webbanvändare kan utföra automatiserade uppgifter om du använder manus i en frågesträng för en URL-adress eller i ett `<?xslt-cwp-query?>`-bearbetningskommando i en XSLT-formatmall. Om du vill se vilka manussteg som inte stöds väljer du kryssrutan Webbpublicering i listan Visa kompatibilitet i fönstret Ändra manus i FileMaker Pro. De manus som är nedtonade är inte webbkompatibla. Information om hur du skapar manus finns i FileMaker Pro Hjälp.

### Tips och råd om manus

Många manussteg fungerar på samma sätt på webben men det finns några som fungerar annorlunda. Se "Funktionssätt i egna webbpubliceringslösningar" på sidan 23. Testa alla manus som ska användas från en webbläsare innan du delar din databas. Logga in med olika användarkonton så att du är säker på att de fungerar för alla klienter. Kontrollera loggfilen för programmet Web Publishing Engine (`pe_application_log.txt`) för att se om du hittar några manusrelaterade fel. Mer information hittar du i "Använda Web Publishing Engines programlogg" på sidan 86.

Här följer några tips och råd:

- Använd konton och behörighet när du vill styra vilka manus en webbanvändare kan utföra. Kontrollera att manusen endast innehåller webbkompatibla manussteg och ge bara tillgång till manus som ska användas från en webbläsare.
- Var också uppmärksam på eventuella biverkningar av manus som utför kombinationer av steg som styrs via behörigheten. Om ett manus exempelvis innehåller steget Radera poster och en webbanvändare inte loggar in med ett konto som ger behörighet att radera poster, utförs inte manussteget Radera poster. Manuset kan dock fortsätta att utföras, vilket kan ge oväntade resultat.
- I fönstret Ändra manus väljer du Kör manus med fullständiga behörigheter om du vill låta manuset utföra uppgifter som du inte vill låta den enskilda användaren utföra. Du kan till exempel hindra användare från att radera poster via deras konton och behörighet, men ändå tillåta dem att utföra ett manus som raderar vissa typer av poster under vissa förutsättningar som anges i manuset.
- Om ett manus innehåller steg som inte stöds, exempelvis steg som inte är webbkompatibla, kan du använda manussteget Tillåt användaren att avbryta för att avgöra hur efterföljande steg ska hanteras.
  - Om manussteget Tillåt användaren att avbryta är aktiverat (på), stoppas manuset vid manussteg som inte stöds.
  - Om Tillåt användaren att avbryta är avstängt kommer manussteg som inte stöds att hoppas över och manuset att fortsätta.
  - Om det här manussteget inte ingår utförs manus som om funktionen vore aktiverad, så att manus stoppas vid manussteg som inte stöds.
- Vissa manus som fungerar med ett steg från en FileMaker Pro-klient kan kräva ett ytterligare steg Verkställ post/sökpost för att spara data på värddatorn. Eftersom webbanvändarna inte har någon direktanslutning till värddatorn får de inte veta när data ändras. Funktioner som till exempel villkorliga värdelistor är inte lika smidiga för webbanvändare eftersom data måste sparas på värddatorn innan effekterna märks i fältet med värdelistan.

- Manus som förändrar data bör innehålla steget Verkställ post/sökpost, eftersom vissa dataändringar inte syns i webbläsaren förrän data sparas eller ”skickas” till servern. Detta gäller flera manussteg som Klipp ut, Kopiera, Klistra in, o.s.v. Många enkelstegsåtgärder bör konverteras till manus och innehålla steget Verkställ post/sökpost. När du skapar manus som ska utföras från en webbläsare ska du inkludera steget Verkställ post/sökpost sist i manuset, så att du är säker på att alla ändringar verkställs.
- Du kan skapa villkorsstyrda manus enligt typen av klient med funktionen Get(programversion). Om det returnerade värdet innehåller ”Web Publishing Engine” vet du att den aktuella användaren använder din databas via anpassad webbpublicering. För vidare information om dessa funktioner, se FileMaker Pro Hjälptext.
- Om du använder ett manus i en XSLT-formatmall som anger eller förändrar ett läge måste du använda FileMaker Server Admin Console för att aktivera alternativet XSLT-databassessioner för Web Publishing Engine. I annat fall bevaras inte lägena mellan förfrågningar. Se FileMaker Server Hjälptext.

## Funktionssätt i egna webbpubliceringslösningar

Följande manussteg fungerar annorlunda på webben än i FileMaker Pro. Information om samtliga manussteg finns i FileMaker Pro Hjälptext.

Manussteg	Funktionssätt i egna webbpubliceringslösningar
Utför manus	Manus kan inte utföras i andra filer, såvida inte filerna ligger på FileMaker Server och anpassad webbpublicering är aktiverad i de andra filerna.
Avsluta programmet	Loggar ut webbanvändare, stänger fönster men avslutar inte webbläsaren.
Tillåt användaren att avbryta	Bestämmer hur manussteg som inte stöds ska hanteras. Välj På om manuset ska stoppas vid steg som inte stöds eller Av om sådana steg ska hoppas över. Närmare information finns i ”Tips och råd om manus” på sidan 22. Webbanvändare kan inte avbryta manus i anpassad webbpublicering, men detta alternativ gör att manussteg som inte stöds förhindrar att manuset fortsätter.
Manuell felhantering	Detta är alltid aktiverat i anpassad webbpublicering. Webbanvändarna kan inte avbryta manus i anpassad webbpublicering.
Paus/fortsätt manus	Du bör undvika detta manus, även om det stöds i anpassad webbpublicering. När ett paussteg körs, görs en paus i manuset. Det är bara manus som innehåller ett fortsätt-manussteg som kan återuppta körningen. Om manuset förblir i pausläge tills sessionen avslutas, slutförs inte manuset.
Sortera poster	Du måste spara en sorteringsordning med manussteget Sortera poster om det ska utföras vid anpassad webbpublicering.
Öppna URL	Detta manussteg har ingen effekt i en anpassad webbpubliceringslösning.
Gå till fält	Du kan inte använda Gå till fält för att aktivera ett visst fält i webbläsaren, men du kan använda manussteget tillsammans med andra manussteg för att utföra uppgifter. Du kan till exempel gå till ett fält, kopiera innehållet, gå till ett annat fält och klistra in värdet. Se till att spara posten med manussteget Verkställ post om du vill se resultatet i webbläsaren.
Verkställ post/sökpost	Skickar posten till databasen.

## Manustriggers i egna webbpubliceringslösningar

I FileMaker Pro kan både manus och användaråtgärder (som att användaren klickar i ett fält) aktivera manustrigger. I anpassad webbpublicering kan endast manus aktivera manustrigger. Om t.ex. en användare av anpassad webbpublicering klickar på ett fält som har en OnObjectEnter-manustrigger kommer utlösaren inte att aktiveras. Om ett manus gör att fokus flyttas till fältet så kommer OnObjectEnter-manustriggern att aktiveras. Mer information om dessa manustrigger finns i FileMaker Pro Hjälptext.



# Kapitel 4

## *Introduktion till anpassad webbpublicering med XSLT*

Du kan använda XSLT-formatmallar för FileMaker när du vill konvertera, filtrera eller formatera XML-data som du vill använda i en webbläsare eller i andra program. I det här kapitlet beskrivs XSLT-formatmallar i FileMaker och ett verktyg som du använder när du skapar XSLT-formatmallar - FileMaker XSLT Site Assistant. Mer detaljerad information om hur FileMaker XSLT-formatmallar är uppbyggda finns i kapitel 6, "Utveckla XSLT-formatmallar för FileMaker".

### FileMaker XSLT-formatmallar

Du kan använda XSLT-formatmallar för följande uppgifter:

- Filtrera FileMaker-data genom att välja vilka databasfält som formatmallen ska publicera.
- Dölja metadata, till exempel databas- och fältnamn.
- Utforma informationen på en webbsida och ange hur webbanvändaren ska samverka med informationen
- Mata ut informationen som HTML eller text, till exempel vCards eller kommaavgränsade värden
- Konvertera data från en FileMaker XML-syntax till en annan XML-syntax och använda informationen i en annan databas eller ett annat program, till exempel SVG.
- Integrera en delmängd FileMaker-data på andra webbplatser och med andra programsystem och anpassade program som kan vara helt annorlunda från FileMaker-databasen.
- Förhindra att obehöriga använder databasinformationen genom att ändra de publicerade fältnamnen.

**Obs!** Anpassad webbpublicering med XSLT för FileMaker Server är baserad på W3C-rekommendationen för XSLT 1.0. Information om XSLT 1.0 finns på [www.w3.org](http://www.w3.org). Tilläggsfunktionerna för XSLT i FileMaker innehåller bland annat sessionshantering, e-post och åtkomst av cookies och sidhuvuden. Information finns i "Använda FileMaker-tilläggsfunktioner och -tilläggsparametrar för XSLT" på sidan 61. Web Publishing Engine stöder inte XSL-FO (XSL Formatting Objects).

### Exempel på hur du kan använda FileMaker XSLT-formatmallar

Här är några exempel på hur du kan använda XSLT-formatmallar för FileMaker:

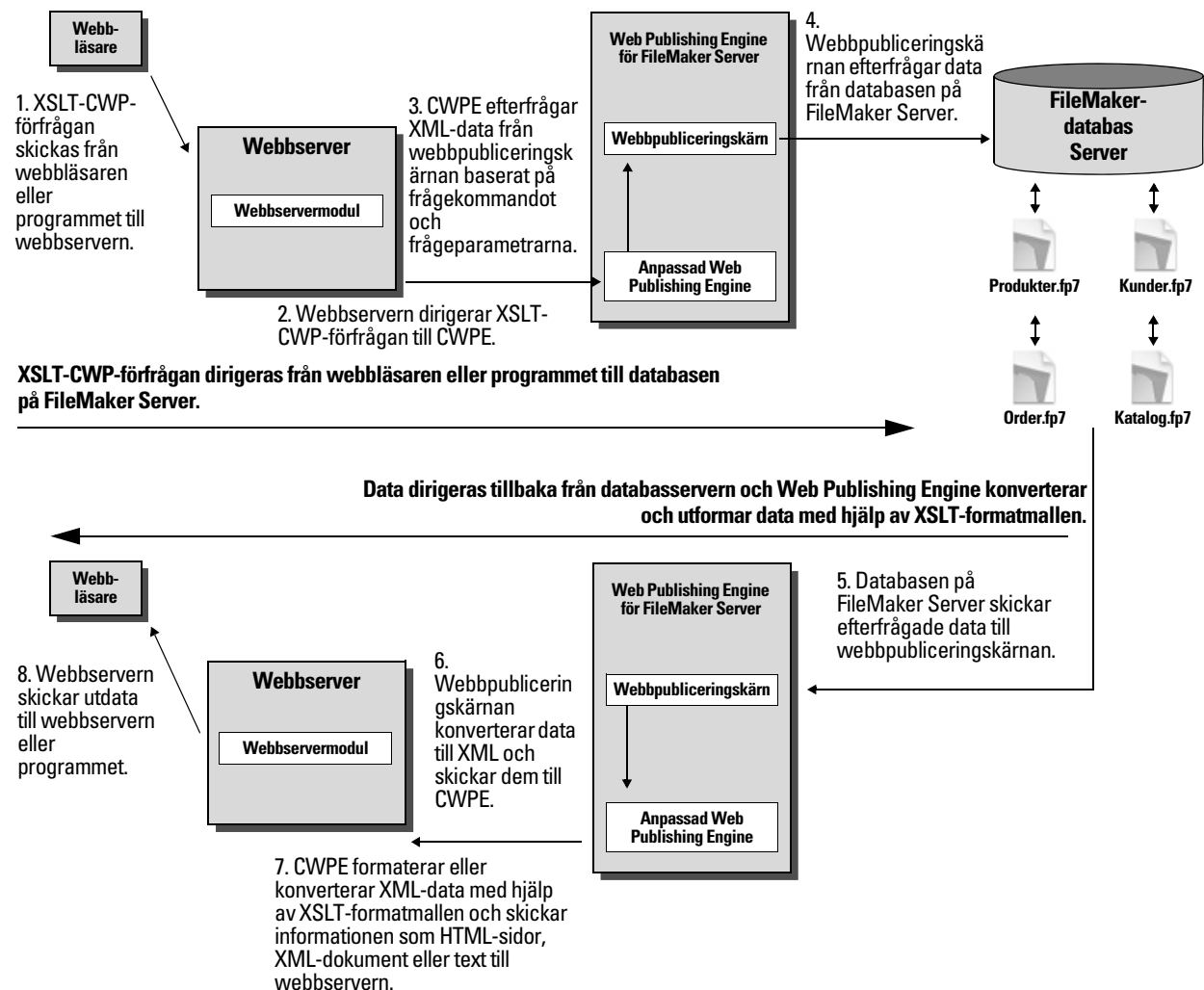
- Du kan infoga en tabell på en webbsida som webbanvändarna kan bläddra i och som innehåller en delmängd av informationen från en FileMaker-databas. Tabellen kan exempelvis innehålla personers namn och adress men inte deras telefonnummer. Om du vill förhindra obehörig åtkomst kan webbsidan innehålla generiska etiketter för informationen (exempelvis "Name") i stället för de riktiga fältnamnen i FileMaker-databasen, till exempel "first\_name".
- Du kan skapa en webbsida eller ett program som integrerar informationen från en FileMaker-portal med information från andra datakällor.
- Du kan lägga till en knapp på en webbsida som skapar ett vCard från en persons kontaktinformation i en FileMaker-databas.
- Du kan konvertera XML-data från en FileMaker-databas till en XML-syntax som kan öppnas i ett kalkylblads- eller databasprogram.

## Komma igång med anpassad webbpublicering med XSLT

Om du känner till standard-XML och standard-XSLT kan du genast börja använda Web Publishing Engine så snart du har lärt dig ett par saker om FileMaker XML- och XSLT-publicering, bland annat hur du använder FileMaker-tillägsfunktioner för XSLT, frågekommandon och frågeparametrar. XSLT Site Assistant är ett verktyg som hjälper dig att skapa formatmallar och lär dig hur de är konstruerade. Därefter kan du ytterligare förfina formatmallarna med egna XML- och XSLT-verktyg.

## Hur Web Publishing Engine skapar sidor baserade på XML-data och XSLT-formatmallar

När en XSLT-CWP-förfrågan skickas till webbservern söker Web Publishing Engine igenom FileMaker-databasen utifrån det frågekommando och de frågeparametrar som har definierats i formatmallen och URL-adressen. Därefter skickas informationen ut enligt instruktionerna i XSLT-formatmallen.



## Allmänna steg för anpassad webbpublicering med XSLT

Här följer en sammanfattning över de steg du måste utföra när du vill använda anpassad webbpublicering med XSLT:

1. Kontrollera att XSLT-publicering är aktiverat i Admin Console. Se FileMaker Server Hjälp.
2. I FileMaker Pro öppnar du varje FileMaker-databas som du vill publicera och ser till att databasen har den utökade behörigheten `fmxmlt` aktiverad för anpassad webbpublicering med XSLT. Se ”Aktivera anpassad webbpublicering i en databas” på sidan 19.

**Obs!** Se till att använda motsvarande FileMaker-databasbehörigheter när du utvecklar formatmallar som ska användas av slutanvändaren. I annat fall kanske du har tillgång till layouter och funktioner i FileMaker-databasen som inte är tillgängliga för slutanvändaren, vilket ger inkonsekventa funktioner.

3. Skapa XSLT-formatmallar som innehåller FileMaker-specifika tilläggfunktioner för XSLT, frågekommandon och frågeparametrar när du vill formatera eller konvertera XML-data från en FileMaker-databas.

Du kan använda verktyget FileMaker XSLT Site Assistant när du vill skapa en eller flera grundläggande XSLT-formatmallar som utgångspunkt för webbplatsen. Se nästa avsnitt, ”Skapa XSLT-formatmallar för FileMaker med FileMaker XSLT Site Assistant”.

Du kan också ändra XSLT-formatmallar med andra XSLT-verktyg eller utveckla formatmallar helt från början. Se kapitel 6, ”Utveckla XSLT-formatmallar för FileMaker”.

4. Kopiera eller flytta XSLT-formatmallarna i mappen `xslt-template-files`, som är en undermapp till Webbpubliceringsmappen i FileMaker Server-mappen på värd datorn där Web Publishing Engine är installerad.

Du kan också placera formatmallarna i en valfri mapp eller mapphierarki i mappen `xslt-template-files`.

5. Placera eventuella statiska filer på webbservern. Se ”Använda FileMaker XSLT-formatmallar på webbplatser och i program” på sidan 30.

6. Skapa eller ändra en webbplats eller ett program som använder XSLT-formatmallar.

Du kan till exempel använda en statisk sida som `index.html` för webbplatsen. Den dirigerar automatiskt webbanvändarna till en XSLT-formatmall eller också har den en länk till en XSLT-formatmall.

7. Se till att säkerhetsfunktionerna för platsen eller programmet fungerar.
8. Testa platsen eller programmet med XSLT-formatmallarna. Använd samma konton och behörigheter som du har definierat för webbanvändarna.
9. Informera om webbplatsen eller programmet och gör dem tillgängliga.

## Skapa XSLT-formatmallar för FileMaker med FileMaker XSLT Site Assistant

FileMaker XSLT Site Assistant är ett program du kan använda när du skapar grundläggande XSLT-formatmallar som kan användas som utgångspunkt för anpassad webbpublicering med XSLT. Genom att använda Site Assistant kan du lära dig hur XSLT-formatmallar för FileMaker är uppbyggda. Därefter kan du bearbeta formatmallarna efter behov med hjälp av dina egna verktyg för utveckling av XSLT-formatmallar eller textredigering. Du kan inte redigera eller uppdatera befintliga formatmallar med hjälp av XSLT Site Assistant, men du kan använda XSLT Site Assistant för att skapa grundformatmallar för en hel webbplats eller en enskild formatmall för att lägga till grundläggande funktioner (t ex radera poster) till en befintlig webbplats.

Du kan använda XSLT Site Assistant när du vill skapa XSLT-formatmallar för alla typer av sidor som passar att använda med FileMaker-databaser via anpassad webbpublicering. Beroende på vilka alternativ du väljer i XSLT Site Assistant kan du skapa en webbplats där användarna kan:

- granska en post åt gången
- visa en lista över alla poster i databasen
- söka i databasen och visa resultatet i en lista
- sortera poster
- lägga till poster
- redigera och duplicera poster
- radera poster
- visa en statistikrapport

Du kan också skapa en valfri startsida som är länkad till de andra genererade XSLT-formatmallssidorna.

Web Publishing Engine använder alla formatmallarna för att dynamiskt hämta data från en FileMaker-databas varje gång en webbanvändare skickar en HTTP-förfrågan och en URL-adress som hänvisar till någon av dina XSLT-formatmallar. Web Publishing Engine transformerar och formaterar XML-informationen med hjälp av en formatmall och genererar HTML-resultatsidan som webbanvändaren kan arbeta med.

**Obs!** Formatmallarna i XSLT Site Assistant konverterar XML-data från FileMaker till HTML-sidor med utgångspunkt i XML-syntaxen `fmresultset`, vilket gör formatmallarna inkompatibla med andra användningsområden för XML-data, t.ex. export av FileMaker PRO XML .

### Innan du använder XSLT Site Assistant

Innan du kan börja använda XSLT Site Assistant för att skapa XSLT-formatmallar för en databas:

- Ange den utökade behörigheten `fmxml` i databasen. När du kör XSLT Site Assistant bör du använda behörighetsuppsättningar som är likvärdiga med dem som du tilldelar webbanvändare. Se ”Aktivera anpassad webbpublicering i en databas” på sidan 19.
- Öppna databasen i komponenten databasserver och låt FileMaker Server vara värd för den. Se FileMaker Server Hjälp.
- Kontrollera att webbserverkomponenten i FileMaker Server-installationen körs.
- Kontrollera att komponenten Web Publishing Engine i FileMaker Server-installationen körs.
- Aktivera XSLT-publicering i Web Publishing Engine för användning och testning av XSLT-formatmallarna. Se FileMaker Server Hjälp.

## Starta XSLT Site Assistant

**Obs!** Om du vill använda XSLT Site Assistant måste du ha installerat Java Runtime Environment 5 eller Java Runtime Environment 6.

Så här startar du XSLT Site Assistant:

1. Öppna en webbläsare på sidan FileMaker Server Web Publishing Tools.

Gå till följande URL-adress:

`http://<server>:16000/tools`

där *<server>* är den dator där FileMaker Server finns.

2. Klicka på Verktyg för PHP Site Assistant och XSLT Site Assistant när du vill gå till sidan Verktyg för FileMaker Server-webbpublicering.

3. Klicka på Starta XSLT Site Assistant.

FileMaker Server installerar nödvändiga JAR-filer på den lokala datorn och visar en förloppsdialogruta tills proceduren är slutförd.

4. (Valfritt) När filerna är installerade kan du välja om du vill installera en ikon för XSLT Site Assistant på skrivbordet. Klicka på OK om du vill installera ikonen.

Nu kan du använda XSLT Site Assistant.

## Använda XSLT Site Assistant

Detaljerad information och stegvisa procedurer för hur du använder XSLT Site Assistant finns i hjälpen för XSLT Site Assistant. Information om hur du använder formatmallarna som skapas av XSLT Site Assistant finns i ”Använda FileMaker XSLT-formatmallar på webbplatser och i program” på sidan 30.

**Viktigt!** När du använder XSLT Site Assistant och väljer en databas som innehåller flera tabeller, måste du välja layouter som är kopplade till samma tabell, annars kommer den skapade sidan att returnera oväntade resultat. En databas kan exempelvis innehålla en produkttabell och en kundtabell. När du väljer layout för en söksida, en sida där användaren kan redigera poster och en sida där användaren kan lägga till poster, måste layouterna vara kopplade till samma tabell.

## Om formatmallar som skapas i XSLT Site Assistant

De XSLT-formatmallar som skapas av XSLT Site Assistant innehåller flera FileMaker-specifika bearbetningsinstruktioner, element och parametrar. Här följer ett par exempel på vad som ingår:

- Bearbetningsinstruktionen `<?xslt-cwp-query params="query string-fragment"?>` anger vilken XML-syntax som används och definierar namnet statistiskt på den databas du valde i XSLT Site Assistant. Se ”Använda statistiskt definierade frågekommandon och frågeparametrar” på sidan 58.

- Elementet `<xsl:param name="request-query"/>` används för åtkomst till frågeinformation i en förfrågan eller HTML-formulärdata. Du kan exempelvis använda det här elementet i formatmallarna i XSLT Site Assistant för att få åtkomst till den aktuella frågeinformationen för att fastställa den aktuella sökvägen i en grupp hittade poster och för att skapa länkar till föregående och nästa post. Se "Använda frågeinformation i en förfrågan" på sidan 62.
- Elementet `<xsl:param name="authenticated-xml-base-uri"/>`, som inte alltid ingår, används för åtkomst till den verifierade URI-basadressen i en förfrågan när fler XML-data behövs i förfrågan. Se "Använda den verifierade grundläggande URI-parametern" på sidan 63.

XSLT Site Assistant skapar även formatmallen `utilities.xsl` som definierar fel och vanliga XSLT-mallar som anropas av flera formatmallar i XSLT Site Assistant.

Mer information om andra avsnitt för formatmallar i XSLT Site Assistant finns i kapitel 6, "Utveckla XSLT-formatmallar för FileMaker".

## Använda FileMaker XSLT-formatmallar på webbplatser och i program

Oavsett om du har skapat XSLT-formatmallarna med XSLT Site Assistant eller om du har skapat helt nya formatmallar, är proceduren för hur de används på en webbsida eller i ett program med Web Publishing Engine densamma.

Så här använder du XSLT-formatmallar för FileMaker på en webbplats eller i ett program:

1. Kopiera eller flytta XSLT-formatmallarna till mappen `xslt-template-files`, som är en undermapp till mappen `Web Publishing` i mappen `FileMaker Server` på värddatorn där `Web Publishing Engine` är installerad.  
Du kan också placera formatmallarna i en valfri mapp eller mapphierarki i mappen `xslt-template-files`.
2. Om XSLT-formatmallarna använder referenser till statiska filer, till exempel statiska bilder eller HTML-filer, placerar du de statiska filerna i sin ursprungliga mapphierarki i rotmappen på webbservern. Kontrollera att den relativa sökvägen bevaras.  
Anta t.ex. att en XSLT-formatmall refererar till en bildfil med namnet `logo.jpg` med hjälp av HTML-markören ``. Filen `logo.jpg` måste finnas på följande plats på webbservern:  
`<rotmappen>/fmi/xsl/logo.jpg`
3. Om ett databascontainerfält innehåller en filreferens i stället för en faktiskt fil, måste det containerobjektet som referensen gäller finnas i webbappen för `FileMaker Pro Web` när posten skapas eller redigeras. Därefter måste fältet kopieras eller flyttas till en mapp med samma relativa placering i webbserververprogrammets rotmapp. Se "Publicera innehållet i containerfält på webben" på sidan 21.  
**Obs!** Om containerfälten innehåller de faktiska filerna i FileMaker-databasen behöver du inte göra något med innehållet i containerfältet, under förutsättning att FileMaker Server fungerar som värd för databasfilen och att filen är tillgänglig på FileMaker Server.

#### 4. Använd följande URL-syntax om du vill efterfråga och bearbeta en XSLT-formatmall:

```
<scheme>://<host>[:<port>]/fmi/xsl/<folder>/<stylesheet>.xsl[?<query string>]
```

Se ”URL-syntax för XSLT-formatmallar för FileMaker” på sidan 54.

**Obs!** För webbplatser är det en bra vana att ta med en XSLT-formatmall som en startsida som inte kräver att användarna anger en frågesträng för att få åtkomst till den. Med XSLT Site Assistant kan du skapa filen `home.xsl` som inte kräver en frågesträng eftersom den använder bearbetningsinstruktionen `<?xslt-cwp-query?>`. Om du t.ex. kopierade formatmallarna (inklusive formatmallen `home.xsl`) till mappen `my_templates`, som är en undermapp till mappen `xslt-template-files`, kan webbanvändare använda följande URL för att begära och bearbeta formatmallarna:

```
http://192.168.123.101/fmi/xsl/my_templates/home.xsl
```

**Viktigt!** Web Publishing Engine tillåter inte att webbanvändarna visar källan för XSLT-formatmallar som är installerade i mappen `xslt-template-files`. När webbanvändarna skickar en förfrågan om att bearbeta en formatmall, skickar Web Publishing Engine endast resultatet av formatmallskonverteringen till webbläsaren eller programmet.

### Felsöka XSLT-formatmallar

Om du har problem med att använda XSLT-formatmallarna bör du kontrollera följande:

- Den utökade behörighetsuppsättningen i databasen är inställd för anpassad webbpublicering med XSLT och kopplad till ett användarkonto. Se ”Aktivera anpassad webbpublicering i en databas” på sidan 19.
- Databasen öppnas i komponenten databasserver och FileMaker Server är värd för den.  
Se FileMaker Server Hjälp.
- Kontonamnet och lösenordet till databasen är korrekta.
- Kontrollera att webbserverkomponenten i FileMaker Server-installationen körs.
- Komponent Web Publishing Engine i FileMaker Server-installationen körs.
- XSLT-publicering är aktiverad i Web Publishing Engine.
  - Öppna sidan FileMaker Server tekniktest i en webbläsare:  

```
http://<server>:16000/test
```

 där `<server>` är den dator där FileMaker Server finns.
  - Klicka på länken Testa XSLT anpassad webbpublicering när du vill öppna en XSLT-sida som har åtkomst till databasen `FMServer_Sample` test.

Mer information finns i *FileMaker Server startbok* och FileMaker Server Hjälp.



# Kapitel 5

## Åtkomst till XML-data med Web Publishing Engine

Du kan visa och uppdatera FileMaker-data i formatet XML (Extensible Markup Language) genom att använda Web Publishing Engine. På samma sätt som HTML har blivit standardspråket för kommunikation på webben, har XML blivit standardspråket för strukturerat datautbyte. Många individer, organisationer och företag använder XML för överföring av produktinformation, transaktioner, lagerinformation och annan affärsinformation.

### Använda anpassad webbpublicering med XML

Om du känner till standard-XML kan du genast börja använda Web Publishing Engine, så snart du har lärt dig några saker om anpassad webbpublicering med XML, till exempel URL-syntaxen och vilka frågeparametrar som ska användas.

Genom att använda HTTP URL-förfrågningar tillsammans med FileMaker-frågekommandon och frågeparametrar kan du ställa frågor till en databas som ligger på FileMaker Server och hämta resultatet i XML-format. Du kan till exempel fråga efter alla poster som har ett visst postnummer och använda den resulterande XML-informationen på vilket sätt som helst.

Du kan också använda Web Publishing Engine XSLT-formatmallar på serversidan om du vill filtrera XML-informationen, göra om informationen till HTML eller text, exempelvis vCards, eller omvandla informationen till andra XML-syntaxer, exempelvis SVG (Scalable Vector Graphics). Se kapitel 4, "Introduktion till anpassad webbpublicering med XSLT" och kapitel 6, "Utveckla XSLT-formatmallar för FileMaker".

Allmän information om XML, ytterligare exempel med XML och länkar till XML-resurser finns på FileMakers webbplats på [www.filemaker.com/se](http://www.filemaker.com/se).

**Obs!** Web Publishing Engine skapar XML-data som är korrekt formaterade och kompatibla med XML 1.0-specifikationen. Mer information om kraven för korrekt XML-kod finns i XML-specifikationen på [www.w3.org](http://www.w3.org).

### Skillnader mellan Web Publishing Engine och XML-import/export i FileMaker Pro

Du kan använda XML-data med FileMaker-databaser med hjälp av Web Publishing Engine eller FileMaker Pro. Det finns dock viktiga skillnader mellan de två metoderna:

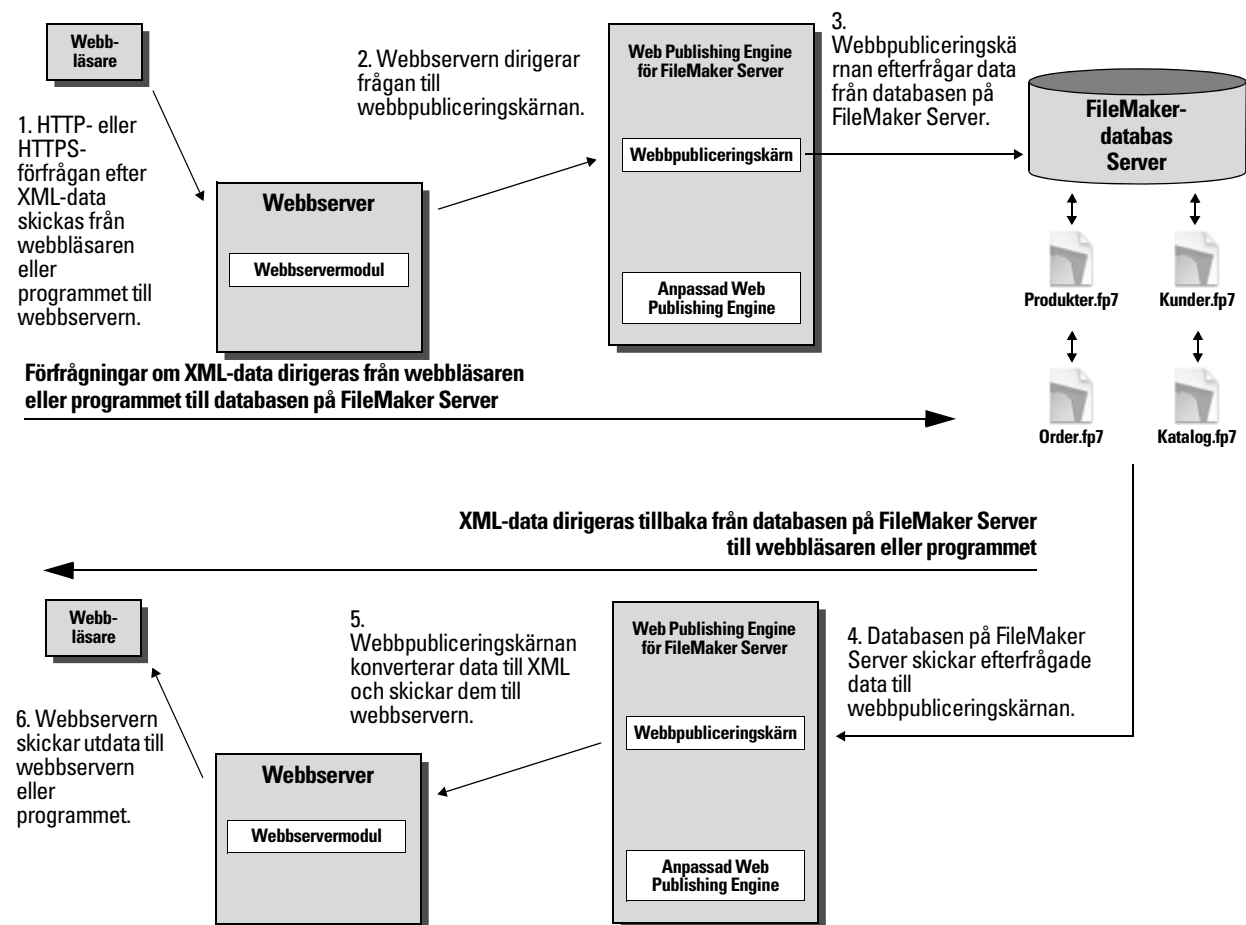
- När det gäller åtkomst av XML-data och XSLT-webbpublicering stöder Web Publishing Engine syntaxerna `fmresultset`, `FMPXMLRESULT` och `FMPXMLLAYOUT`. Vid XML-import använder FileMaker Pro syntaxen `FMPXMLRESULT` och vid export syntaxen `FMPXMLRESULT` eller `FMPDSORESET`. Se "Åtkomst till XML-data med Web Publishing Engine" på sidan 38.
- Om du vill få åtkomst till XML-data med Web Publishing Engine kan du använda en frågesträng i en URL-adress. Om du vill importera och exportera XML med FileMaker Pro använder du menykommandon och manus i FileMaker Pro.
- Web Publishing Engine är serverbaserad och kan installeras på samma värd som FileMaker Server, eller en annan. Import och export av FileMaker Pro är skrivbordsbaserad.

- Du kan dynamiskt få åtkomst till XML-data från FileMaker-databaser genom att använda URL-förfrågningar med Web Publishing Engine. XML-exportfunktionen i FileMaker Pro genererar en särskild XML-datafil.
- Att arbeta med XML-data via Web Publishing Engine är en interaktiv åtgärd. FileMaker Pro XML-import/export är en gruppåtgärd.
- Web Publishing Engine kan hämta XML-data från en FileMaker-portal, vilket inte FileMaker Pro kan.
- Web Publishing Engine kan hämta data i ett containerfält, vilket inte FileMaker Pro kan.
- Web Publishing Engine ger åtkomst i realtid till FileMaker-data via HTTP eller HTTPS, vilket inte FileMaker Pro gör.

**Obs!** Information om hur du importerar och exporterar data i XML-format med hjälp av FileMaker Pro finns i hjälpen för FileMaker Pro.

## Hur Web Publishing Engine genererar XML-data från en förfrågan

När en förfrågan om XML-data skickas till webbservern skickar Web Publishing Engine frågan till FileMaker-databasen och returnerar data som ett XML-dokument.



## Åtkomst till XML-data från Web Publishing Engine

Här nedan visas en översikt över hur du använder Web Publishing Engine för att få åtkomst till XML-data i en FileMaker-databas:

1. Kontrollera att XSLT-publicering är aktiverat i FileMaker Server Admin Console. Se FileMaker Server Hjälp.
2. I FileMaker Pro öppnar du varje FileMaker-databas som du vill publicera och ser till att databasen har den utökade behörigheten fmxml aktiverad för anpassad webbpublicering i XML. Se ”Aktivera anpassad webbpublicering i en databas” på sidan 19.

Om du vill vi kunna använda XML-data i en portal ställer du in vyn för databaslayouten till Visa en post eller Visa som lista. Om en användare eller ett manus ändrar vyn i databaslayouten till Visa som tabell är endast den första relaterade posten (den första raden i portalen) åtkomlig som XML-data.

XML-data matas ut i en ordning som motsvarar den ordning i vilken fältobjekten lades till i layouten. Om du vill att dataordningen för XML ska matcha den ordning som fälten visas i på skärmen (uppifrån och ned, vänster till höger), väljer du alla fält, grupperar dem och delar sedan upp gruppen. Den här proceduren återställer layoutordningen så att den matchar skärmordningen.

3. Skicka en HTTP- eller HTTPS-förfrågan i form av en URL-adress som anger XML-syntaxen i FileMaker, ett frågekommando och en eller fler FileMaker-frågeparametrar till Web Publishing Engine via ett HTML-formulär, en HREF-länk eller ett manus i programmet eller på webbsidan. Du kan också skriva URL-adressen i en webbläsare.

Information om hur du anger URL-adressen finns i nästa avsnitt, ”URL-syntax för XML-data och containerobjekt”. Information om frågekommandon och frågeparametrar finns i ”Efterfråga XML-data med hjälp av frågesträngar i FileMaker” på sidan 47, och bilaga A, ”Giltiga namn i frågesträngar”.

4. Web Publishing Engine använder den syntax du angav i URL-adressen för att generera XML-data som innehåller resultatet av din förfrågan, till exempel en grupp poster från databasen, och returnerar det till programmet eller webbläsaren.
5. I webbläsaren visas data (om programmet har en XML-tolk) eller så använder programmet informationen på ett sätt som du har angett.  
Om du angav en formatmall på klientsidan tillämpar webbläsartolken även formatmallsinstruktionerna. Se ”Bearbeta formatmallar på serversidan och klientsidan” på sidan 50.

## URL-syntax för XML-data och containerobjekt

I det här avsnittet visas den URL-syntax du använder för att Web Publishing Engine ska kunna använda XML-data och containerobjekt från FileMaker-databaser. URL-syntaxen för XSLT-formatmallar skiljer sig från XML. Se ”URL-syntax för XSLT-formatmallar för FileMaker” på sidan 54 och ”URL-syntax för FileMaker-containerobjekt i XSLT-lösningar” på sidan 55.

## URL-syntax för XML-data

Den URL-syntax du ska använda för att Web Publishing Engine ska få åtkomst till XML-data från FileMaker-databaser är:

```
<scheme>://<host>[:<port>]/fmi/xml/<xml_grammar>.xml[?<query string>]
```

där:

- <scheme> kan vara protokollet HTTP eller HTTPS.
- <host> är IP-adressen eller domännamnet för värden som webbservern är installerad på.
- <port> är valfritt och anger porten som webbservern använder. Om ingen port anges används standardporten för respektive protokoll (port 80 för HTTP och port 443 för HTTPS).
- <xml\_grammar> är namnet på XML-syntaxen i FileMaker. Möjliga värden är fmresultset.xml, FMPXMLRESULT.xml, FMPXMLLAYOUT.xml och FMPDSORESET.xml. Se ”Använda syntaxen fmresultset” på sidan 39 och ”Använda andra FileMaker XML-syntaxer” på sidan 43.
- <query string> är en kombination av ett frågekommando och en eller fler frågeparametrar för FileMaker XML-publicering. (Kommandot –dbnames kräver inga parametrar.) Se ”Efterfråga XML-data med hjälp av frågesträngar i FileMaker” på sidan 47, och bilaga A, ”Giltiga namn i frågesträngar”.

**Obs!** URL-syntaxen, inklusive namnen på frågekommando och parametrar, är skiftlägeskänslig, utom för delar av frågesträngen. Huvuddelen av URL-adressen anges med små bokstäver (gemener), förutom de tre syntaxnamnen i versaler: FMPXMLRESULT, FMPXMLLAYOUT och FMPDSORESET. Information om vilka regler som gäller för stora och små bokstäver i frågesträngar finns i ”Riktlinjer för frågekommandon och frågeparametrar” på sidan 90.

Här är två exempel på URL-adresser för åtkomst av XML-data via Web Publishing Engine:

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=products&-lay=sales&-findall
```

```
http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?-db=products&-lay=sales&-findall
```

## URL-syntax för FileMaker-containerobjekt i XML-lösningar

I ett XML-dokument som har skapats för en XML-lösning är den syntax som du använder för att referera till ett containerobjekt annorlunda för containerfält som innehåller själva objektet i databasen, jämfört med containerfält som innehåller en referens till objektet.

- Om ett containerfält innehåller själva objektet i databasen använder containerfältets <data>-element följande relativa URL-syntax för att referera till objektet:

```
<data>/fmi/xml/cnt/data.<extension>?<query string></data>
```

där <extension> är filnamnstillägget som identifierar objekttypen, till exempel .jpg. Med filnamnstillägget kan MIME-typen tillåta webbläsaren att identifiera rätt typ av containerdata. Information om <query string> finns i föregående avsnitt, ”URL-syntax för XML-data”.

Till exempel:

```
<data>/fmi/xml/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2</data>
```

**Obs!** I XML-informationen som har skapats för ett containerfält är värdet för frågeparametern -field ett fullständigt fältnamn. Talet inom parentes indikerar repetitionsnumret för containerfältet och skapas för både repeterade fält och icke-repeterade fält. Se ”Syntax för ett fullständigt fältnamn” på sidan 91.

Om du vill hämta containerdata från databasen använder du följande syntax:

```
<scheme>://<host>[:<port>]/fmi/xml/cnt/data.<extension>?<query string>
```

Information om <scheme>, <host> och <port> finns i föregående avsnitt, ”URL-syntax för XML-data”.

Till exempel:

```
http://www.company.com/fmi/xml/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2
```

- Om ett containerfält innehåller en filreferens i stället för själva objektet, innehåller containerfältets <data>-element en relativ sökväg till objektet. Till exempel:

```
<data>/images/logo.jpg</data>
```

**Obs!** Det containerobjekt som referensen gäller måste lagras i mappen i FileMaker Pro Web när posten skapas eller redigeras och sedan kopieras eller flyttas till en mapp med samma relativa sökväg i rotmappen på webbserverprogrammet. Se ”Publicera innehållet i containerfält på webben” på sidan 21.

- Om ett containerfält är tomt är containerfältets <data>-element också tomt.

**Obs!** Syntaxen för containerobjekt som använder XML är annorlunda än syntaxen för containerobjekt som använder XSLT. Se ”URL-syntax för FileMaker-containerobjekt i XSLT-lösningar” på sidan 55.

## URL-textkodning

URL-adresserna för åtkomst av XML-data och containerobjekt måste kodas i formatet UTF-8 (Unicode Transformation 8 Bit). Mer information finns i ”Om UTF-8-kodade data” på sidan 47.

Om du exempelvis vill sätta värdet för fältet info till fiancée, kan du använda följande URL-adress:

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=members&-lay=relationships&-recid=2
&info=fianc%C3%A9e&-edit
```

I det här URL-exemplet är %C3%A9 den URL-kodade UTF-8-motsvarigheten till tecknet é.

Mer information om URL-textkodning finns i URL-specifikationen på [www.w3.org](http://www.w3.org).

## Åtkomst till XML-data med Web Publishing Engine

Om du vill öppna XML-data via Web Publishing Engine använder du en URL-adress som innehåller namnet på den FileMaker-syntax som ska användas, ett FileMaker-frågekommando och en eller fler FileMaker-frågeparametrar. Web Publishing Engine genererar XML-data från din databas som har utformats av en av följande typer av XML-syntax:

- **fmresultset:** Denna syntax rekommenderas för Web Publishing Engine. Den är flexibel och optimerad för bearbetning av XSLT-formatmallar med enklare åtkomst till fält via namn och enklare hantering av relatedset-data (portal). Denna syntax är också på ett mer direkt sätt kopplad till terminologi och funktioner i FileMaker, till exempel globala lagringsalternativ och identifiering av statistikfält och beräkningsfält. Du kan använda denna syntax för åtkomst av XML-data och för XSLT-formatmallar. Denna syntax underlättar webbpublicering genom att den är mer ordrik än syntaxen FMPXMLRESULT. Se ”Använda syntaxen fmresultset” på sidan 39.
- **FMPXMLRESULT och FMPXMLLAYOUT:** Du kan också använda syntaxerna FMPXMLRESULT och FMPXMLLAYOUT med Web Publishing Engine när du vill använda XML-data och XSLT-formatmallar. Om du endast vill använda en formatmall för både XML-export och anpassad webbpublicering måste du använda syntaxen FMPXMLRESULT. Om du vill kunna visa värdelistor och fältinformation i layouter måste du använda syntaxen FMPXMLLAYOUT. Se ”Använda andra FileMaker XML-syntaxer” på sidan 43.
- **FMPDSORESET:** Syntaxen FMPDSORESET, som stöds i FileMaker Pro vid export av XML, kan inte användas för åtkomst av XML-data via Web Publishing Engine. Syntaxen FMPDSORESET stöds inte för XSLT-formatmallar. Information om syntaxen FMPDSORESET finns i FileMakerFMPDSORESETPro Hjälp.

Beroende på vilken syntax du anger i URL-förfrågan kommer Web Publishing Engine att skapa ett XML-dokument med hjälp av en av syntaxerna. Alla XML-dokument innehåller ett förvalt XML-namntryck för syntaxen. Se nästa avsnitt, ”Namnuttryck för FileMaker XML”. Använd någon av syntaxerna i ditt dokument eller på webbsidan när du vill visa och arbeta med FileMaker-data i XML-format.

**Obs!** XML-data som skapas i Web Publishing Engine är kodade i UTF-8-format (Unicode Transformation Format 8). Se ”Om UTF-8-kodade data” på sidan 47.

### Namntryck för FileMaker XML

Unika namnuttryck i XML särskiljer olika XML-markörer efter det program de har utformats för. Om XML-dokumentet exempelvis innehåller två <DATABASE>-element, ett för FileMaker Pro-baserade XML-data och ett annat för Oracle-baserade XML-data identifieras <DATABASE>-elementet för vart och ett av dem med hjälp av namnuttrycken.

Web Publishing Engine skapar förvalda namnuttryck för varje syntax.

För denna syntax	Används följande namnuttryck
fmresultset	xmlns="http://www.filemaker.com/xml/fmresultset"
FMPXMLRESULT	xmlns="http://www.filemaker.com/fmpxmlresult"
FMPXMLLAYOUT	xmlns="http://www.filemaker.com/fmpxmllayout"

## Felkoder i FileMaker-databaser

Web Publishing Engine genererar en felkod som motsvarar felet, i felkodselementen i början av varje XML-dokument, om det uppstod något fel när det senaste frågekommandot kördes. Värdet noll (0) returneras om det inte finns något fel.

För denna syntax	Används följande syntax
fmresultset	<error code="0"></error>
FMPXMLRESULT	<ERRORCODE>0</ERRORCODE>
FMPDSORESLT	<ERRORCODE>0</ERRORCODE>

Felkodselementet i XML-dokumentet visar fel som är relaterade till databasen och frågesträngarna. Andra typer av fel kan uppstå för XSLT-formatmallar och hanteras på ett annat sätt. Se bilaga B, "Felkoder för anpassad webbpublicering".

## Hämta dokumenttypsdefinitionerna för syntaxerna i FileMaker

Du kan hämta dokumenttypsdefinitionerna (DTD-filer) för syntaxerna i FileMaker med hjälp av en HTTP-förfrågan.

För denna syntax	Använder du följande HTTP-förfrågan
fmresultset	http://<värd>[:<port>]/fmi/xml/fmresultset.dtd
FMPXMLRESULT	http://<värd>[:<port>]/fmi/xml/FMPXMLRESULT.dtd
FMPXMLLAYOUT	http://<värd>[:<port>]/fmi/xml/FMPXMLLAYOUT.dtd
FMPDSORESLT	http://<värd>[:<port>]/fmi/xml/FMPDSORESLT.dtd?-db=<databas>&-lay=<layout>

## Använda syntaxen fmresultset

XML-elementnamnen i denna syntax använder FileMaker-terminologi och lagringen av fält separeras från fälttypen. Syntaxen innehåller även funktioner för identifiering av statistikfält, beräkningsfält och globala fält.

Om du vill använda syntaxen fmresultset anger du följande namn på syntaxen fmresultset i URL-adressen som används när du efterfrågar XML-dokumentet från Web Publishing Engine:

fmresultset.xml

Till exempel:

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-findall

**Obs!** När du anger syntaxen fmresultset måste du använda små bokstäver (gemener).

Web Publishing Engine skapar ett XML-dokument med hjälp av syntaxen fmresultset. I XML-dokumentet refererar Web Publishing Engine till dokumenttypsdefinitionen för syntaxen fmresultset i instruktionen <!DOCTYPE> på den andra raden i dokumentet, omedelbart efter instruktionen <?xml...?>. <!DOCTYPE>-instruktionen anger vilken URL-adress du använder för att hämta DTD-filen för syntaxen fmresultset.

## Beskrivning av element i syntaxen fmresultset

Syntaxen fmresultset består huvudsakligen av <datasource>-elementet, <metadata>-elementet och <resultset>-elementet.

### <datasource>-element

I syntaxen fmresultset innehåller <datasource> -elementet tabell, layout, datumformat, tidsformat, tidsstämpelformat, totalt antal och databasattribut.

- Datumformatattributet för <datasource>-elementet anger datumformatet i XML-dokumentet.

MM/dd/yyyy

där:

- MM är det tvåsiffriga värdet för månaden (01 till 12, där 01 är januari och 12 är december)
- dd är det tvåsiffriga värdet för dagen i månaden (00 till 31)
- yyyy är det fyrsiffriga värdet för året
- Tidsformatattributet för <datasource>-elementet anger tidsformatet i XML-dokumentet:

HH:mm:ss

där:

- HH är det tvåsiffriga värdet för timmen (00 till 23) i formatet med 24 timmar)
- mm är det tvåsiffriga värdet för minuter (00 till 59)
- ss är det tvåsiffriga värdet för sekunder (00 till 59)
- Attributet för tidsstämpelattributet för <datasource>-elementet kombinerar datumformatet och tidsformatet till en tidsstämpel.

MM/dd/yyyy HH:mm:ss

**<metadata>-element**

Elementet <metadata> i syntaxen fmresultset innehåller en eller flera <fältdefinitioner> och elementet <relatedset-definition>. Var och en innehåller attribut för ett av fälten i resultatet.

Attributet <fältdefinition> anger:

- vare sig fältet är ett fält för automatisk inmatning ("yes" eller "no")
- vare sig fältet är ett fält för år angivet med fyra siffror ("yes" eller "no")
- vare sig fältet är ett globalt fält ("yes" eller "no")
- det maximala antalet repeterade värden (max-repeat, attribut)
- det maximala antalet tillåtna tecken (max characters, attribut)
- om fältet är ett ej tomt fält ("yes" eller "no")
- om fältet endast är för numeriska data ("yes" eller "no")
- result ("text", "number", "date", "time", "timestamp", eller "container")
- om fältet är ett fält för tidpunkt ("yes" eller "no")
- typ ("normal", "calculation", eller "summary")
- och fältnamnet (fullständigt vid behov)

Elementet <relatedset-definition> motsvarar en portal. Varje relaterat fält i en portal motsvaras av elementet <field-definition> som finns i <relatedset-definition>-elementet. Om det finns flera relaterade fält i en portal grupperas fältdefinitionerna för de relaterade fälten i ett enda <relatedset-definition>-element.

**<resultset>-element**

Elementet <resultset> innehåller alla <record>-element som returneras som resultat av en fråga och ett attribut för det totala antalet hittade poster. Varje <record>-element innehåller fältdata för en post i resultatet – inklusive attributen mod-id och record-id för posten och <data>-elementet som innehåller data för ett fält i posten.

Varje post i en portal motsvaras av ett <record>-element i <relatedset>-elementet. Nummerattributet för <relatedset>-elementet anger antalet poster i portalen och tabellattributet anger vilken tabell som är kopplad till portalen.

## Exempel på XML-data i syntaxen fmresultset

Här följer ett exempel på XML-data som har skapats med syntaxen fmresultset.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN" ""http://localhost:16014/fmi/xml/fmresultset.dtd">
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
  <felkod="0" />
  <product build="12/31/2012" name="FileMaker Web Publishing Engine" version="0.0.0.0" />
  <datasource database="art" date-format="MM/dd/yyyy" layout="web3" table="art" time-format="HH:mm:ss" timestamp-format="MM/dd/yyyy HH:mm:ss" total-count="12" />
  <metadata>
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Title" not-empty="no" numeric-only="no" result="text" time-of-day="no" type="normal" />
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Artist" not-empty="no" numeric-only="no" result="text" time-of-day="no" type="normal" />
    <relatedset-definition table="artlocations">
      <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="artlocations::Location" not-empty="no" numeric-only="no" result="text" time-of-day="no" type="normal" />
      <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="artlocations::Date" not-empty="no" numeric-only="no" result="date" time-of-day="no" type="normal" />
    </relatedset-definition>
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Style" not-empty="no" numeric-only="no" result="text" time-of-day="no" type="normal" />
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="length" not-empty="no" numeric-only="no" result="number" time-of-day="no" type="calculation" />
  </metadata>
  <resultset count="1" fetch-size="1">
    <record mod-id="6" record-id="4">
      <field name="Title">
        <data> Spring in Giverny 3</data>
      </field>
      <field name="Artist">
        <data>Claude Monet</data>
      </field>
      <relatedset count="0" table="artlocations" />
      <field name="Style">
        <data />
      </field>
      <field name="length">
        <data>19</data>
      </field>
    </record>
  </resultset>
</fmresultset>
```

## Använda andra FileMaker XML-syntaxer

De andra FileMaker XML-syntaxerna innehåller information om fälttyper, värdelistor och layouter. FMPXMLRESULT fungerar på samma sätt som fmresultset. Om du vill kunna visa värdelistor och fältinformation i layouter måste du använda syntaxen FMPXMMLAYOUT. Syntaxerna FMPXMLRESULT och FMPXMMLAYOUT är mer kompakta vid datautbyte.

Om du vill använda syntaxen FMPXMLRESULT anger du följande syntaxnamn i den URL-adress du använder för att efterfråga XML-dokumentet från Web Publishing Engine:

FMPXMLRESULT.xml

Till exempel:

<http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?–db=employees&–lay=family&–findall>

Om du vill använda syntaxen FMPXMMLAYOUT anger du följande syntaxnamn med frågekommandot –view i den URL-adress du använder för att efterfråga XML-dokumentet från Web Publishing Engine:

FMPXMMLAYOUT.xml

Till exempel:

<http://192.168.123.101/fmi/xml/FMPXMMLAYOUT.xml?–db=employees&–lay=family&–view>

**Obs!** När du anger syntaxerna FMPXMLRESULT och FMPXMMLAYOUT måste du skriva namnet med stora bokstäver.

I XML-dokumentet refererar Web Publishing Engine dokumenttypsdefinitionen för syntaxen fmresultset i <!DOCTYPE>-instruktionen på den andra raden i dokumentet, omedelbart efter <?xml...?>-instruktionen. <!DOCTYPE>-instruktionen anger URL-adressen du använder för att hämta DTD-filen för syntaxen.

## Beskrivning av element i syntaxen FMPXMLRESULT

I syntaxen FMPXMLRESULT innehåller elementet <DATABASE> attributen NAME, RECORDS, DATEFORMAT, LAYOUT och TIMEFORMAT.

Attributet DATEFORMAT i <DATABASE>-elementet anger datumformatet i XML-dokumentet. Attributet TIMEFORMAT i <DATABASE>-elementet anger tidsformatet i XML-dokumentet. Datum- och tidsformaten för syntaxerna FMPXMLRESULT och fmresultset är desamma. Se tabellerna i ”Beskrivning av element i syntaxen fmresultset” på sidan 40.

Elementet <METADATA> i syntaxen FMPXMLRESULT innehåller ett eller flera FIELD-element där vart och ett innehåller information för något av fälten/kolumnerna i resultatet, inklusive fältets namn så som det definieras i databasen, fälttyp, möjliga Ja/Nej-värden i tomma fält (EMPTYOK-attribut) och det maximala antalet upprepade värden (MAXREPEAT-attribut). Giltiga värden för fälttyper är TEXT, NUMBER, DATE, TIME, TIMESTAMP och CONTAINER.

Elementet <RESULTSET> innehåller alla <ROW>-element som returneras som resultat av en fråga och ett attribut för det totala antalet hittade poster. Varje <ROW>-element innehåller fält-/kolumndata för en rad i resultatet. Dessa data innehåller RECORDID och MODID för raden (se ”Frågeparametern –modid (ändrings-ID)” på sidan 100) och <COL>-elementet. Elementet <COL> innehåller data för ett fält eller en kolumn på raden där flera <DATA>-element motsvarar ett av värdena i ett repeterat fält eller ett portalfält.

## Exempel på XML-data i grammatikfilen FMPXMLRESULT

Här följer ett exempel på XML-data som har skapats med syntaxen FMPXMLRESULT.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN" ""http://localhost:16014/fmi/xml/
FMPXMLRESULT.dtd">
<FMPXMLRESULT xmlns="http://www.filemaker.com/fmpxmlresult">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="12/31/2012" NAME="FileMaker Web Publishing Engine" VERSION="0.0.0.0" />
  <DATABASE DATEFORMAT="MM/dd/yyyy" LAYOUT="web" NAME="art" RECORDS="12" TIMEFORMAT="HH:mm:ss" />
  <METADATA>
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Title" TYPE="TEXT" />
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Artist" TYPE="TEXT" />
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Image" TYPE="CONTAINER" />
  </METADATA>
  <RESULTSET FOUND="1">
    <ROW MODID="6" RECORDID="15">
      <COL>
        <DATA>Spring in Giverny 4</DATA>
      </COL>
      <COL>
        <DATA>Claude Monet</DATA>
      </COL>
      <COL>
        <DATA>/fmi/xml/cnt/data.jpg?-db=art&-lay=web&-recid=15&-field=Image(1)</DATA>
      </COL>
    </ROW>
  </RESULTSET>
</FMPXMLRESULT>
```

Ordningsföljden på <COL>-elementen motsvarar ordningsföljden på <FIELD>-elementen i <METADATA>-elementet. Exempel: Där fälten ”Title” och ”Artist” visas i <METADATA>-elementet, visas ”Village Market” och sedan ”Camille Pissarro” i samma ordning i <RESULTSET>- och <ROW>-elementen.

## Beskrivning av element i grammatikfilen FMPXMLLAYOUT

I syntaxen FMPXMLLAYOUT innehåller elementet <LAYOUT> namnet på layouten, namnet på databasen och <FIELD>-elementen för varje fält som hittas i motsvarande layout i databasen. Varje <FIELD>-element beskriver stilen på fältet och innehåller VALUELIST-attributet för alla tillhörande värdelistor i fältet.

Elementet <VALUELISTS> innehåller ett eller flera <VALUELIST>-element för alla värdelistor som hittas i layouten. Vart och ett innehåller namnet på värdelistan och ett <VALUE>-element för vart och ett av värdena i listan.

Beroende på vilka alternativ du har valt i dialogrutan **Ange fält** i värdelista i FileMaker-databasen, innehåller elementet <VALUE> ett DISPLAY-attribut som innehåller värde endast i det första fältet, endast i det andra fältet eller i båda fälten i en värdelista. Antal t.ex. att det första fältet i en värdelista lagrar konststilens ID-nummer (t.ex. "100") och det andra fältet visar konststilens tillhörande namn (t.ex. "Impressionism"). Här är en sammanfattning av innehållet i DISPLAY-attributet när de olika kombinationerna av alternativ har valts i dialogrutan **Ange fält** i värdelista:

- Om **Visa** även värden från det andra fältet inte har markerats, innehåller DISPLAY-attributet endast värdet i det första fältet från en värdelista. I följande exempel med XML-data innehåller DISPLAY-attributet endast konststilens ID-nummer:

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="100">100</VALUE>
    <VALUE DISPLAY="100">101</VALUE>
    <VALUE DISPLAY="100">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

- Om både **Visa** även värden från det andra fältet och **Visa värden endast från det andra fältet** har markerats, innehåller DISPLAY-attributet värde endast i det andra fältet. I följande exempel med XML-data innehåller DISPLAY-attributet endast konststilens namn:

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="Impressionism">100</VALUE>
    <VALUE DISPLAY="Kubism">101</VALUE>
    <VALUE DISPLAY="Abstrakt">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

- Om både **Visa** även värden från det andra fältet och **Visa värden endast från det andra fältet** har markerats, innehåller DISPLAY-attributet värden i båda fälten i en värdelista. I följande exempel med XML-data innehåller DISPLAY-attributet både konststilens ID-nummer och konststilens namn:

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="100 Impressionism">100</VALUE>
    <VALUE DISPLAY="101 Kubism">101</VALUE>
    <VALUE DISPLAY="102 Abstrakt">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

För fält för datum, tid och tidsstämpel formateras data i värdelistor med hjälp av ”fm”-formatet för fälttypen. ”fm”-formaten är MM/dd/yyyy för datum, HH:mm:ss för tid och MM/dd/yyyy HH:mm:ss för tidsstämpel. Se ”Använda tilläggsfunktioner för datum, tid och dag” på sidan 75. Om t.ex. en värdelista med födelsedagar används till en popup-meny i ett ”födelsedagsfält” i en layout och om fältet ”födelsedag” är av typen datum så kommer alla utdatavärden för den värdelistan att vara i datumformatet ”fm”.

**Obs!** Om två fält med olika fälttyper i en layout delar samma värdelistan bestämmer det första fältets typ formatet för de data som finns i värdelistan.

## Exempel på XML-data i grammatikfilen FMPXMLLAYOUT

Här följer ett exempel på XML-data som har skapats med grammatikfilen FMPXMLLAYOUT.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLLAYOUT PUBLIC "-//FMI//DTD FMPXMLLAYOUT//EN" ""http://localhost:16014/fmi/xml/
FMPXMLLAYOUT.dtd">
<FMPXMLLAYOUT xmlns="http://www.filemaker.com/fmpxmllayout">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="12/31/2012" NAME="FileMaker Web Publishing Engine" VERSION="0.0.0.0" />
  <LAYOUT DATABASE="art" NAME="web2">
    <FIELD NAME="Title">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Artist">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Image">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="artlocations::Location">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="artlocations::Date">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Style">
      <STYLE TYPE="POPUPMENU" VALUELIST="style" />
    </FIELD>
  </LAYOUT>
  <VALUELISTS>
    <VALUELIST NAME="style">
      <VALUE DISPLAY="Impressionism">100</VALUE>
      <VALUE DISPLAY="Kubism">101</VALUE>
      <VALUE DISPLAY="Abstrakt">102</VALUE>
    </VALUELIST>
  </VALUELISTS>
</FMPXMLLAYOUT>
```

## Om UTF-8-kodade data

Alla XML-data som skapas i Web Publishing Engine är kodade i UTF-8-format. I detta format komprimeras data från 16-bitars Unicode-format (standard) till 8-bitarsformatet för ASCII-tecken. XML-tolkar krävs för Unicode- och UTF-8-kodning.

UTF-8-kodning innehåller en direkt motsvarighet av värdena mellan 0 och 127 för standarduppsättningen av ASCII-tecken samt tillhandahåller flerbytekodning för Unicode-tecken med högre värden.

**Obs!** Se till att använda en webbläsare eller ett textredigeringsprogram som stöder filer i UTF-8-format.

Kodningsformatet UTF-8 innehåller följande funktioner:

- Alla ASCII-tecken är UTF-8-tecken på 1 byte. En giltig ASCII-sträng är en giltig UTF-8-sträng.
- Alla icke-ASCII-tecken (alla tecken ur det övre ASCII-området) är en del av ett flerbyte-tecken.
- Ett UTF-8-teckens första byte indikerar antalet ytterligare byte i tecknet.
- Den första byten i ett flerbyte-tecken kan lätt urskiljas från den efterföljande byten, vilket gör det enkelt att hitta starttecknet i en godtycklig position i en dataström.
- Det är enkelt att konvertera mellan UTF-8 och Unicode.
- UTF-8-kodningen är relativt kompakt och för text med en stor andel ASCII-tecken är den kompaktare än Unicode. I värsta fall är en UTF-8-sträng 50 % större än motsvarande Unicode-sträng.

## Efterfråga XML-data med hjälp av frågesträngar i FileMaker

Om du vill efterfråga XML-data från en FileMaker-databas använder du frågekommandon och frågeparametrar i en frågesträng i FileMaker. Du kan exempelvis använda frågekommandot `-findall` i följande frågesträng i en URL-adress när du vill efterfråga en lista över alla produkter i en FileMaker-databas med namnet "products":

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=products-lay=sales&-findall`

En frågesträng får endast innehålla ett frågekommando, till exempel `-new`. För de flesta frågekommandon krävs också olika frågeparametrar i frågesträngen. För alla frågekommandon förutom `-dbnames` krävs exempelvis parametern `-db`, som anger vilken databas frågan ska ställas till.

Du kan också använda frågekommandon och frågeparametrar i en URL-adress eller i en `<?xslt-cwp-query?>`-bearbetningsinstruktion i en XSLT-formatmall. Se kapitel 6, "Utveckla XSLT-formatmallar för FileMaker".

Det här avsnittet innehåller en sammanfattning av frågekommandona och frågeparametrarna i FileMaker. Mer information om hur du använder dem i en frågesträng finns i bilaga A, "Giltiga namn i frågesträngar".

**Obs!** Web Publishing Engine stöder även ett andra frågekommando (`-process`) och tre frågeparametrar som endast är definierade för att användas med XSLT-formatmallar. Se "Använda frågesträngar i XSLT-formatmallar för FileMaker" på sidan 56.

Använd det här frågekommandot	Om du vill utföra följande kommando
<code>-dbnames</code>	Hämta namnen på alla databaser som finns på servern och som är webbdelade.
<code>-delete</code>	Radera post.
<code>-dup</code>	Duplicera post.
<code>-edit</code>	Redigera post.
<code>-find</code>	Sök post(er).
<code>-findall</code>	Sök efter alla poster.

Använd det här frågekommandot	Om du vill utföra följande kommando
<code>-findany</code>	Slumpmässig sökning efter post.
<code>-findquery</code>	Utför komplex eller sammansatt sökpost.
<code>-layoutnames</code>	Hämta namnen på alla tillgängliga layouter för databaser som finns på servern och som är webbdelade.
<code>-new</code>	Lägg till ny post.
<code>-scriptnames</code>	Hämta namnen på alla tillgängliga manus för en databas som finns på servern och som är webbdelad.
<code>-view</code>	Hämta layoutinformation från en databas om syntaxen <code>FMPXMMLAYOUT</code> anges. Hämtar avsnittet <code>&lt;metadata&gt;</code> i XML-dokumentet och en tom resultatuppsättning om syntaxen <code>fmresultset</code> eller <code>FMPXMLRESULT</code> anges.

Använd följande frågeparametrar	Med följande frågekommandon
<code>-db</code> (databasnamn)	Krävs med alla frågekommandon förutom <code>-dbnames</code> och <code>-process</code> (endast XSLT-förfrågningar)
<code>-delete.related</code>	Valfritt tillsammans med <code>-edit</code>
<code>-field</code>	Krävs om du vill ange ett fält i en URL-adress för containerförfrågningar. Se ”URL-syntax för FileMaker-containerobjekt i XML-lösningar” på sidan 37.
fältnamn	Minst ett fältnamn krävs för <code>-edit</code> . Valfritt tillsammans med <code>-find</code> . Se ”Frågeparametern fältnamn (namn på annat fält än containerfält)” på sidan 97.
fältnamn.op (operator)	Valfritt tillsammans med <code>-find</code>
<code>-lay</code> (layoutnamn)	Krävs med alla frågekommandon förutom <code>-dbnames</code> , <code>-layoutnames</code> , <code>-scriptnames</code> och <code>-process</code> (endast XSLT-frågor)
<code>-lay.response</code> (växla layout för XML-svar)	Krävs med alla frågekommandon förutom <code>-dbnames</code> , <code>-layoutnames</code> , <code>-scriptnames</code> och <code>-process</code> (endast XSLT-frågor)
<code>-lop</code> (logisk operator)	Valfritt tillsammans med <code>-find</code>
<code>-max</code> (maximalt antal poster)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code>
<code>-modid</code> (ändrings-id)	Valfritt tillsammans med <code>-edit</code>
<code>-query</code>	Krävs med den sammanfattande sökposten <code>-findquery</code> .
<code>-recid</code> (post-id)	Krävs tillsammans med <code>-edit</code> , <code>-delete</code> , <code>-dup</code> . Valfritt tillsammans med <code>-find</code>
<code>-relatedsets.filter</code>	Valfritt tillsammans med <code>-find</code> , <code>-edit</code> , <code>-new</code> , <code>-dup</code> och <code>-findquery</code> .
<code>-relatedsets.max</code>	Valfritt tillsammans med <code>-find</code> , <code>-edit</code> , <code>-new</code> , <code>-dup</code> och <code>-findquery</code> .
<code>-script</code> (utför manus)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code> , <code>-findany</code> , <code>-new</code> , <code>-edit</code> , <code>-delete</code> , <code>-dup</code> , <code>-view</code>
<code>-script.param</code> (skicka ett parametervärde till det manus som anges av <code>-script</code> )	Valfritt tillsammans med <code>-script</code>
<code>-script.prefind</code> (utför manus före <code>-find</code> , <code>-findany</code> och <code>-findall</code> )	Valfritt tillsammans med <code>-find</code> , <code>-findany</code> , <code>-findall</code>
<code>-script.prefind.param</code> (skicka ett parametervärde till det manus som anges av <code>-script.prefind</code> )	Valfritt med <code>-script.prefind</code>
<code>-script.presort</code> (utför manus före sortering)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code>
<code>-script.presort.param</code> (skicka ett parametervärde till det manus som anges av <code>-script.presort</code> )	Valfritt med <code>-script.presort</code>
<code>-skip</code> (hoppa över poster)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code>

Använd följande frågeparametrar	Med följande frågekommandon
<code>-sortfield.[1-9]</code> (sortera fält)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code>
<code>-sortorder.[1-9]</code> (sorteringsordning)	Valfritt tillsammans med <code>-find</code> , <code>-findall</code>
<code>-stylehref</code> (formatmall HREF)	Valfritt med alla frågekommandon (när du vill ange en URL-adress till en formatmall för <code>-styletype</code> )
<code>-styletype</code> (formatmalltyp)	Valfritt med alla frågekommandon (när du vill ange formatmall på klientsidan)

## Byta layout för ett XML-svar

Med frågeparametern `-lay` anger du vilken layout du vill använda när du frågar efter XML-data. Ofta fungerar samma layout för bearbetning av data som förfrågningen resulterar i. Ibland kanske du vill söka efter data med en layout som innehåller fält som av säkerhetsskäl inte finns i någon annan layout som du vill använda för att visa resultatet. (Om du vill söka efter data i ett fält måste fältet placeras i den layout du angav i XML-förfrågningen.)

Om du vill visa ett XML-svar i en annan layout än den som används för bearbetning av XML-förfrågningen, kan du använda den valfria frågeparametern `-lay.response`.

Följande fråga söker till exempel efter värden som är större än 100 000 i fältet `Salary` i layouten `Budget`. Resultatet visas med layouten `ExecList`, som inte innehåller fältet `Salary`.

<http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=Budget&Salary=100000&Salary.op=gt&-find&-lay.response=ExecList>

## Bearbetning av en XML-förfrågan

Det finns flera frågeparametrar som påverkar bearbetningen av en XML-förfrågan och genereringen av ett XML-dokument.

FileMaker Server och Web Publishing Engine bearbetar en XML-fråga i följande ordning:

1. Bearbeta frågeparametern `-lay` -
2. Ange de globala fältvärden som angetts i frågan ("`?global=`"-delen av en URL-adress).
3. Bearbeta frågeparametern `-script.prefind`, om den har angetts.
4. Frågekommandon, till exempel `-find` eller `-new`.
5. Bearbeta frågeparametern `-script.presort`, om den har angetts.
6. Resultatet sorteras, om sortering angavs.
7. Bearbeta frågeparametern `-lay.response` så att den växlar till en annan layout, om detta har angetts.
8. Bearbeta frågeparametern `-script`, om den har angetts.
9. XML-dokumentet genereras.

Om något av ovanstående steg genererar en felkod, stoppar bearbetningen och alla eventuella kvarvarande steg utförs inte. Eventuella tidigare steg i förfrågningen utförs fortfarande.

Anta att en förfrågan tar bort den aktuella posten, sorterar posterna och sedan kör ett manus. Om parametern `-sortfield` anger ett fält som inte finns, tar förfrågan bort den aktuella posten och returnerar felkod 102 ("`?Field is missing`"), men manuset utförs inte.

## Bearbeta formatmallar på serversidan och klientsidan

Web Publishing Engine stöder bearbetning av XSLT-formatmallar på serversidan. Du kan också använda en frågeparameter som anger bearbetning av en formatmall på klientsidan.

Det är viktigt att förstå skillnaderna mellan de två sätten att bearbeta formatmallar samt hur säkerheten påverkas om du använder bearbetning på klientsidan. Bearbetning på serversidan är säkrare än bearbetning på klientsidan eftersom den första tekniken inte ger webbanvändare åtkomst till ofiltrerade XML-data. Med bearbetning på serversidan visas data i ett format som ägaren av informationen eller den som utformade XSLT-formatmallen bestämmer. Vid bearbetning på serversidan döljs databasnamn, fältnamn och andra implementeringsdetaljer för webbanvändaren. Du kan också använda bearbetning på serversidan för att ange statistiskt definierade frågeparametrar, vilket förhindrar obehörig användning av frågekommandon och frågeparametrar, till exempel databasnamn. Se kapitel 4, ”Introduktion till anpassad webbpublicering med XSLT” och kapitel 6, ”Utveckla XSLT-formatmallar för FileMaker”.

Om din lösning kräver bearbetning av formatmall på klientsidan kan du låta Web Publishing Engine skapa en bearbetningsinstruktion för XML-formatmallen med varje syntax, genom att inkludera parametrarna `-styletype` och `-stylehref` i FileMaker-frågesträngen. Du kan visa XML-dokumentet genom att använda överlappande formatmallar (CSS) eller XSLT-formatmallar.

- Parametern `-styletype` används för att ange värdet på typattributet (`type=text/css` eller `type=text/xsl`).
- Du använder parametern `-stylehref` när du vill ange värdet på det HREF-attribut som anger sökvägen till formatmallen med hjälp av en absolut sökväg. Till exempel: `href=/mystylesheet.css` eller `href=/stylesheets/mystylesheet.xsl`. Namnet på formatmallen kan vara vilket som helst, men det måste innehålla ändelsen `.css` eller `.xsl`.

Här följer ett exempel på en frågesträng som genererar bearbetning av formatmall på klientsidan:

```
http://localhost/fmi/xml/fmresultset.xml?-db=products-lay=sales&-findall&-styletype=text/xsl&-stylehref=/mystylesheet.xsl
```

**Obs!** Tecknet `"/` i `-stylehref=/document.xsl` i det här exemplet används eftersom formatmallen finns i rotmappen i webbserverprogrammet. Använd en URL-adress för den formatmall som använder en absolut sökväg när du vill ange dess sökväg på webbservern. Formatmallen kan även finnas på en annan webbserver.

Baserat på denna förfrågan inkluderar Web Publishing Engine följande bearbetningsinstruktion i XML-dokumentet:

```
<?xml-stylesheet type="text/xsl" href="/mystylesheet.xsl"?>
```

Kopiera eller placera formatmallen för bearbetning på klientsidan på webbservern på den plats som anges av den absoluta sökvägen i URL-adressen för HREF-attributet.

**Viktigt!** Placera inte formatmallar för bearbetning på klientsidan i mappen `xslt-template-files`, som används för bearbetning av XSLT-formatmallar på serversidan. Se ”Använda FileMaker XSLT-formatmallar på webbplatser och i program” på sidan 30.

**Obs!** En del webbläsare stöder inte bearbetning på klientsidan. Mer information finns i dokumentationen till webbläsaren.

## Felsöka åtkomst till XML-dokument

Om du har problem med att öppna XML-dokument med Web Publishing Engine kontrollerar du att:

- Den utökade behörighetsuppsättningen i databasen är inställd för XML-anpassad webbpublicering och kopplad till ett användarkonto. Se ”Aktivera anpassad webbpublicering i en databas” på sidan 19.
- Databasen öppnas i komponenten databasserver av FileMaker Server-installationen och öppnas av FileMaker Server. Se FileMaker Server Hjälp.
- Kontonamnet och lösenordet till databasen är korrekta.
- Kontrollera att webbserverkomponenten i FileMaker Server-installationen körs.
- Komponenten Web Publishing Engine i FileMaker Server-installationen körs.
- XML-publicering är aktiverad i komponenten Web Publishing Engine. Se FileMaker Server Hjälp.



# Kapitel 6

## Utveckla XSLT-formatmallar för FileMaker

Det här kapitlet innehåller information om hur XSLT-formatmallar för FileMaker är uppbyggda och hur du använder FileMaker-tilläggsfunktionerna för XSLT.

### Använda XSLT-formatmallar med Web Publishing Engine

När du utvecklar och använder XSLT-formatmallar för att efterfråga FileMaker XML-data via Web Publishing Engine bör du tänka på följande:

- Om du vill använda en XSLT-formatmall med Web Publishing Engine måste du ange namnet på XSLT-formatmallen i en URL-adress. Om du inte anger en formatmall eller om Web Publishing Engine inte kan hitta eller tolka formatmallen, visas en felsida. Se ”URL-syntax för XSLT-formatmallar för FileMaker” på sidan 54.
- Formatmallens filnamn och namnet på mappen där formatmallen lagras måste vara URL-kodad UTF-8. Om formatmallen måste vara kompatibel med äldre webbläsare begränsar du namnen till ASCII-tecken.
- Du måste ange vilken XML-syntax som ska användas, antingen som en frågeparameter i URL-adressen eller som en statiskt definierad frågeparameter i bearbetningsinstruktionen `<?xslt-cwp-query?>`. Om du inte anger en XML-syntax visas ett fel. Se ”Ange en XML-syntax för en XSLT-formatmall för FileMaker” på sidan 56.
- Du kan ange frågeparametrar som identifierar de XML-data som du vill efterfråga, antingen i URL-adressen eller som en statiskt definierad frågeparameter i bearbetningsinstruktionen `<?xslt-cwp-query?>`. Se ”URL-syntax för XSLT-formatmallar för FileMaker” på sidan 54 och ”Använda statiskt definierade frågekommandon och frågeparametrar” på sidan 58.
- Om du vill kan du ange textkodning för en XSLT-förfrågan genom att använda frågeparametern `-encoding`. Om du inte anger en kodning använder Web Publishing Engine standardinställningen för textkodning för förfrågningar. Se ”Ange textkodning för förfrågningar” på sidan 59.
- Du kan ange en utdatametod via metodattributet i elementet `<xsl:output>`. Om du inte anger en utdatametod använder Web Publishing Engine HTML. Du kan också ange utdatasidans kodning med hjälp av kodningsattributet i elementet `<xsl:output>`. Om du inte anger en kodning använder Web Publishing Engine standardinställningen för textkodning för utdatasidor. Se ”Ange utdatametod och kodning” på sidan 60.
- Du kan ange textkodning för e-postmeddelanden som skickas från Web Publishing Engine via en funktionsparameter för tilläggsfunktionen `fmxslt:send_email()`. Se ”Skicka e-postmeddelanden från Web Publishing Engine” på sidan 68.

Web Publishing Engine skapar en förfrågan genom att använda ett frågekommando och en frågeparametrar som statiskt definieras i den valfria bearbetningsinstruktionen `<?xslt-cwp-query?>`. De statiskt definierade frågekommandona och frågeparametrarna utgör den grundläggande förfrågan. Bearbetningsinstruktionen `<?xslt-cwp-query?>` är inte obligatorisk i en formatmall, men den grundläggande förfrågan får högre prioritet än matchande frågekommando eller frågeparametrar som anges i URL-frågesträngen. Därefter lägger Web Publishing Engine till eventuella frågekommandon eller frågeparametrar i den grundläggande förfrågan i URL-frågesträngen som *inte* definieras i bearbetningsinstruktionen `<?xslt-cwp-query?>`. Web Publishing Engine använder denna förfrågan för att ta fram XML-informationen och returnera den till webbläsaren eller programmet i den utdatametod du valde, eller som HTML.

## Om referens till FileMaker-tilläggsfunktionerna för XSLT

Den här versionen innehåller en FileMaker-databas som heter XSLT Reference.fp7 och som innehåller korta beskrivningar och exempel på alla FileMaker-tilläggsfunktioner för XSLT. Databasen med funktionsreferenser hittar du i följande katalog på alla datorer där FileMaker Server är installerat (huvuddator eller arbetsdator):

### Mac OS

/Library/FileMaker Server/Examples/XSLT

### Windows

<drive>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT

där <enhet> är den primära enhet där programmet startas.

## Om FileMakers startlösning för XSLT

Denna version innehåller en startlösning för FileMaker XSLT och visar ett exempel på vad du kan göra med XSLT-lösningar. XSLT-startlösningen hittar du i följande katalog på alla datorer där FileMaker Server är installerat (huvuddator eller arbetsdator):

### Mac OS

/Library/FileMaker Server/Example/XSLT/Starter Solution

### Windows

<enhet>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT\Starter Solution

där <enhet> är den primära enhet där programmet startas.

## URL-syntax för XSLT-formatmallar för FileMaker

Om du vill använda XSLT-formatmallar för FileMaker tillsammans med Web Publishing Engine använder du följande URL-syntax:

<scheme>://<host>[:<port>]/fmi/xsl/[<path>]/<stylesheet.xml>[?<query string>]

där:

- <scheme> kan vara protokollet HTTP eller HTTPS.
- <host> är IP-adressen eller domännamnet för värden som webbservern är installerad på.
- <port> är valfritt och anger porten som webbservern använder. Om ingen port anges används standardporten för respektive protokoll (port 80 för HTTP och port 443 för HTTPS).
- <path> är valfritt och anger sökvägen till undermappen i mappen xslt-template-files som innehåller XSLT-formatmallen.

- <stylesheet.xml> är filnamnet på XSLT-formatmallen.
- <query string> kan vara en kombination av ett frågekommando och en eller flera frågeparametrar för anpassad webbpublicering med XSLT. Se ”Använda frågesträngar i XSLT-formatmallar för FileMaker” på sidan 56, och bilaga A, ”Giltiga namn i frågesträngar”. Om den angivna formatmallen innehåller bearbetningsinstruktionen <?xslt-cwp-query?> får det statistiskt definierade frågekommandot och frågeparametrarna företräde framför eventuella matchande frågekommandon eller frågeparametrar i URL-frågesträngen. Se ”Använda statistiskt definierade frågekommandon och frågeparametrar” på sidan 58.

**Obs!** URL-syntaxen, inklusive namnen på frågekommando och parametrar, är skiftlägeskänslig, utom för delar av frågesträngen. Huvuddelen av URL-adressen skrivs med små bokstäver (gemener), förutom syntaxnamnen i versaler; FMPXMLRESULT och FMPXMLLAYOUT. Information om vilka regler som gäller för stora och små bokstäver i frågesträngar finns i ”Riktlinjer för frågekommandon och frågeparametrar” på sidan 90.

Här följer ett exempel på en URL-adress för användning av en FileMaker XSLT-formatmall med Web Publishing Engine:

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–db=mydatabase
&–lay=mylayout&–findall
```

## URL-syntax för FileMaker-containerobjekt i XSLT-lösningar

I ett XML-dokument som har skapats för en XSLT-lösning är den syntax som du använder för att referera till ett containerobjekt annorlunda för containerfält som innehåller själva objektet i databasen, jämfört med containerfält som innehåller en referens till objektet.

- Om ett containerfält innehåller själva objektet i databasen använder containerfältets <data>-element följande URL-syntax för att referera till objektet:

```
<data>/fmi/xsl/cnt/data.<filtillägg>?<frågesträng></data>
```

där <extension> är filnamnstillägget som identifierar objekttypen, till exempel .jpg eller .mov. Information om <query string> finns i föregående avsnitt, ”URL-syntax för XSLT-formatmallar för FileMaker”.

Till exempel:

```
<data>/fmi/xsl/cnt/data.jpg?–db=products&–lay=sales&–field=product_image(1)&–recid=2</data>
```

**Obs!** I XML-informationen som har skapats för ett containerfält är värdet för frågeparametern –field ett fullständigt fältnamn. Talet inom parentes indikerar repetitionsnumret för containerfältet och skapas för både repeterade fält och icke-repeterade fält. Se ”Syntax för ett fullständigt fältnamn” på sidan 91.

Om du vill hämta containerdata från databasen använder du följande syntax:

```
<scheme>://<host>[:<port>]/fmi/xsl/cnt/data.<extension>?<query string>
```

Information om <scheme>, <host> och <port> finns i föregående avsnitt, ”URL-syntax för XSLT-formatmallar för FileMaker”.

Till exempel:

```
http://www.company.com/fmi/xsl/cnt/data.jpg?–db=products&–lay=sales&–field=product_image(1)&–recid=2
```

- Om ett containerfält innehåller en filreferens i stället för själva objektet, innehåller containerfältets <data>-element en relativ sökväg till objektet. Om filen logo.jpg exempelvis var placerad i mappen Web i mappen FileMaker Pro skulle containerfältets <data>-element vara:

```
<data>/images/logo.jpg</data>
```

**Obs!** Det containerobjekt som referensen gäller måste lagras i mappen i FileMaker Pro Web när posten skapas eller redigeras och sedan kopieras eller flyttas till en mapp med samma relativa sökväg i rotmappen på webbserverprogrammet. Se "Publicera innehållet i containerfält på webben" på sidan 21.

- Om ett containerfält är tomt är containerfältets <data>-element också tomt.

## Använda frågesträngar i XSLT-formatmallar för FileMaker

När du använder en frågesträng i en URL-adress eller i bearbetningsinstruktionen <?xslt-cwp-query?> i en FileMaker XSLT-formatmall kan du inkludera frågekommandon och frågeparametrar som har definierats för förfrågan av XML-data från en FileMaker-databas. Se "Efterfråga XML-data med hjälp av frågesträngar i FileMaker" på sidan 47.

Du kan använda följande frågekommando och frågeparametrar som endast har definierats för användning med XSLT-formatmallar.

### Använd detta XSLT frågekommando eller parameternamn

	För att	Kommentar
-grammar3	Ange XML-syntaxen för XSLT-CWP-förfrågningar eller för XSLT-formatmallar. Se nästa avsnitt, "Ange en XML-syntax för en XSLT-formatmall för FileMaker".	Denna frågeparameter krävs för alla XSLT-förfrågningar.
-encoding	Ange textkodning för en förfrågan. Se "Ange textkodning för förfrågningar" på sidan 59.	Denna frågeparameter är valfri för alla XSLT-förfrågningar.
-process	Bearbeta en formatmall utan att begära data. Se "Bearbeta XSLT-förfrågningar som inte skickar frågor till FileMaker Server" på sidan 60.	Detta frågekommando kräver frågeparametern -grammar.
-token	Överföra värden mellan sidor utan att använda sessioner eller cookies. Se "Överföra information mellan formatmallar med hjälp av token" på sidan 61.	Denna frågeparameter är valfri för alla XSLT-förfrågningar.

## Ange en XML-syntax för en XSLT-formatmall för FileMaker

Den XML-syntax som rekommenderas med anpassad webbpublicering med XSLT är syntaxen fmresultset, som har utformats för att enkelt kunna användas med XSLT. Se "Använda syntaxen fmresultset" på sidan 39. Du kan också använda de äldre syntaxerna FMPXMLRESULT eller FMPXMLLAYOUT. Om du vill kunna visa värdelistor och fältinformation i layouter måste du använda syntaxen FMPXMLLAYOUT. Se "Använda andra FileMaker XML-syntaxer" på sidan 43. Du kan inte använda syntaxen FMPDSORESET med anpassad webbpublicering med XSLT.

Du anger syntaxen för en FileMaker XSLT-formatmall genom att använda frågeparametern -grammar i en URL-adress eller en statiskt definierad frågeparameter i bearbetningsinstruktionen <?xslt-cwp-query?>.

T.ex. i en URL-adress:

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–db=mydatabase
&–lay=mylayout&–findall
```

Du kan exempelvis skriva i en bearbetningsinstruktion:

```
<?xslt-cwp-query params="–grammar=fmresultset&–db=mydatabase&–lay=mylayout&–findall"?>
```

**Viktigt!** Om du inte anger en XML-syntax för en FileMaker XSLT-formatmall visas felet ”QUERY -ER0001”. Se bilaga B, ”Felkoder för anpassad webbpublicering”.

## Namnuttryck och prefix för XSLT-formatmallar för FileMaker

Unika namnuttryck i XSLT särskiljer olika XSLT-markörer efter det program de har utformats för. I elementet `<xsl:stylesheet>` i början av alla XSLT-formatmallar använder du namnuttrycken för FileMaker-tilläggsfunktionerna för XSLT och de specifika syntaxer du använder i formatmallen.

Om du vill använda följande	Använder du detta namnuttryck	Använd följande prefix
XML-syntaxen fmresultset	<code>xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"</code>	fmrs
Syntaxen FMPXMLRESULT	<code>xmlns:fmp="http://www.filemaker.com/fmpxmlresult"</code>	fmp
Syntaxen FMPXMLLAYOUT	<code>xmlns:fml="http://www.filemaker.com/fmpxmllayout"</code>	fml
Syntaxen för XML-frågor	<code>xmlns:fmq="http://www.filemaker.com/xml/query"</code>	fmq
För FileMaker-tilläggsfunktionerna för XSLT	<code>xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"</code>	fmxslt

Du måste också använda följande namnuttryck (obligatoriskt) i varje XSLT-formatmall:

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
```

Här följer exempel på namnuttryck:

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
  xmlns:fml="http://www.filemaker.com/fmpxmllayout"
  xmlns:fmq="http://www.filemaker.com/xml/query"
  xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"
  exclude-result-prefixes="xsl fmrs fmq fml fmxslt">
```

## Använda statistiskt definierade frågekommandon och frågeparametrar

Du kan förhindra obehörig användning av frågekommandon och frågeparametrar med XSLT-formatmallen genom att statistiskt definiera frågekommandona och frågeparametrarna du vill ska användas när XML-data efterfrågas. Om frågekommandon och frågeparametrar är statistiskt definierade i en formatmall (även om det inte behövs) gäller de före eventuella matchande frågekommandon eller frågeparametrar som en klient kanske försöker ange i URL-frågesträngen.

De formatmallar som skapas av verktyget XSLT Site Assistant är statistiskt definierade frågekommandon och frågeparametrar. FileMaker rekommenderar att du använder statistiskt definierade frågekommandon och frågeparametrar när du vill öka säkerheten i din lösning.

Om du vill definiera frågekommandon och frågeparametrar statistiskt använder du följande bearbetningsinstruktion i början av FileMaker XSLT-formatmallen:

```
<?xslt-cwp-query params="query string-fragment"?)>
```

där:

query string-fragment är en sträng som innehåller värdeparen i följande format:

```
namn=värde&namn2=värde2...
```

där:

namn är en sträng som är namnet på ett frågekommando, en frågeparameter eller ett databasfält.

värde är ett godtyckligt strängvärde i vilken längd som helst. För frågeparametrar och fältnamn använder du det specifika värde du vill definiera, till exempel "–db=products". För frågekommandon ska du inte ange tecknet ("=") eller ett värde efter kommandonamnet, till exempel –findall. Se bilaga A, "Giltiga namn i frågesträngar".

De strängar som används i fragmentet måste vara URL-kodade. Se "URL-textkodning" på sidan 37. Du måste använda samma teckenkodning som anges av kodningsattributet i markören <xsl:output>. Om ingen kodning anges använder Web Publishing Engine sin konfigurerade standardkodning.

Avgränsaren mellan de två namnparen måste vara ett ampersandtecken (&).

Anta att du använde följande bearbetningsinstruktion i formatmallen my\_stylesheet.xml:

```
<?xslt-cwp-query params="–db=products&–lay=sales&–grammar=fmresultset&productname=the%20item&–find"?)>
```

Denna bearbetningsinstruktion skulle tvinga alla förfrågningar om my\_stylesheet.xml att använda syntaxen fmresultset med produkt databasen och försäljningslayouten och göra en –find-förfrågan med fältet för produktnamn satt till värdet "the%20item".

Om en kund gjorde följande förfrågan med hjälp av my\_stylesheet.xml:

```
http://server.company.com/fmi/xsl/my_stylesheet.xml?-lay=revenue&city=London&-edit
```

skulle Web Publishing Engine bearbeta följande XML-förfrågan:

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=products&–lay=sales&productname=the%20item&city=London&–find
```

De statistiskt definierade frågekommandona och frågeparametrarna gäller före frågeparametern –lay=revenue och frågeparametern –edit som kunden angav. Eftersom fältet city inte var statistiskt definierat i bearbetningsinstruktionen inkluderar Web Publishing Engine värdet "London" i XML-förfrågan, för fältet city som kunden angav.

## Ange textkodning för förfrågningar

Web Publishing Engine utför följande steg i den ordning som anges tills kodningen i en XSLT-förfrågan identifieras:

1. Kontrollerar om charset-attributet har angetts i frågehuvudet Content-Type.
2. Kontrollerar om du angav en kodning med frågeparametern `–encoding`. Du kan ange denna parameter i en URL-adress eller som en statiskt definierad frågeparameter i bearbetningsinstruktionen `<?xslt-cwp-query?>`. Värdet i parametern `–encoding` visar vilken kodning som används för de övriga parametrarna i förfrågan. Giltiga värden för denna parameter visas i tabellen nedan. Till exempel:  

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xml?-db=products-lay=sales&-grammar=fmresultset
&-encoding=Shift_JIS&-findall
```
3. Använd standardinställningen för textkodningsalternativet begäranden och utdatasidor för Web Publishing Engine. Första gången Web Publishing Engine installeras är den förvalda textkodningsinställningen för förfrågningar UTF-8. Du kan ändra inställningarna med hjälp av Admin Console. Se FileMaker Server Hjälp.

När Web Publishing Engine har identifierat kodningen används den kodningen och inga fler försök att fastställa kodningen görs. Om charset-attributet till exempel är inställt i frågehuvudet Content-Type, använder inte Web Publishing Engine värdet i frågeparametern `–encoding`.

Den textkodning som anges via ovanstående metoder måste använda någon av följande kodningar:

Kodning	Beskrivning
US-ASCII	Den grundläggande ASCII-teckenuppsättning som vanligtvis används för vanlig text i e-post.
ISO-8859-1	Teckenuppsättningen Latin 1 används vanligtvis för webbplatser med romerska tecken och i e-postmeddelanden där tecken i det övre ASCII-området används.
ISO-8859-15	Teckenuppsättningen Latin 9, som är nästan identisk med teckenuppsättningen Latin 1 med tillägget av symbolen för euro €.
ISO-2022-JP	Den japanska ISO-kodning som vanligtvis används i japanska e-postmeddelanden.
Shift_JIS	Den japanska ISO-kodning som vanligtvis används på japanska webbsidor.
UTF-8	Åttabitskodningen av Unicode. Det blir allt vanligare att använda UTF-8 i e-postmeddelanden och på webbsidor i och med att de flesta webbläsare och e-postklienter har stöd för den kodningen. Eftersom UTF-8 stöder alla Unicode-tecken, kan kodningen hantera sidor med vilket språk som helst.

### Obs!

- Första gången du installerar Web Publishing Engine är den förvalda textkodningen för utdatasidor inställd till UTF-8. Se nästa avsnitt, ”Ange utdatametod och kodning”. För e-postmeddelanden använder Web Publishing Engine textkodningen ISO-8859-1. Du kan ändra dessa inställningar med Admin Console.
- Du kan även ange kodning för e-postmeddelanden genom att använda tilläggfunktionen `fmxslt:send_email(smtp-fält som String, meddelande som String, kodning som String)`. Se ”Skicka e-postmeddelanden från Web Publishing Engine” på sidan 68.

## Ange utdatametod och kodning

Du kan ange utdatametod och kodning för utdatasidor genom att använda metoden och kodningsattributen för elementet `<xsl:output>`. Båda dessa attribut är valfria.

Metodattributet anger utdatatyp, som kan vara html, text eller xml. Inga andra metodtyper stöds. Om du inte anger en metod använder Web Publishing Engine metoden html.

Kodningsattributet anger utdatasidornas kodning. Du kan ange en av kodningarna som visas i tabellen i föregående avsnitt. Om du inte anger en kodning använder Web Publishing Engine standardinställningen för textkodning för förfrågningar.

Till exempel:

```
<xsl:output method="html" encoding="ISO-8859-1"/>
```

Om du inte använder elementet `<xsl:output>` i en formatmall, skickas HTML-sidorna kodade efter den aktuella textkodningsinställningen för utdatasidor.

## Koda XSLT-formatmallar

Förutom kodning av förfrågningar och utdatasidor måste kodningen av XSLT-formatmallarna anges i kodningsattributet i XML-deklarationen överst i formatmallen. Du kan använda vilken som helst av textkodningarna som visas i tabellen på sidan 59.

Denna deklaration anger exempelvis UTF-8 som kodning av formatmallen:

```
<?xml-version="1.0" encoding="UTF-8"?>
```

Om du inte anger formatmallskodning kommer kodningen UTF-8 att användas.

## Bearbeta XSLT-förfrågningar som inte skickar frågor till FileMaker Server

Du kan använda frågekommandot `–process` när du vill bearbeta XSLT-förfrågningar som inte behöver några data från databasen eller om formatmallen inte kräver databasspecifik information, exempelvis poster, fältnamn eller layoutnamn. Genom att använda kommandot `–process` i dessa situationer kan du minska belastningen på FileMaker Server.

Du kan exempelvis använda kommandot `–process` när du vill:

- läsa in en formatmall som genererar en statisk sida, om ingen databasinformation behövs
- läsa in en formatmall som skapar en ny post, om formatmallen inte kräver databas- eller layoutinformation, exempelvis en värdelista
- använda en tilläggfunktion, exempelvis `fmxml:send_email()`, som inte kräver några data från databasen
- få åtkomst till information som är lagrad i en session om ingen databasinformation behövs

Kommandot `–process` returnerar ett XML-dokument som innehåller produktinformation om Web Publishing Engine.

Den enda parameter som krävs för kommandot `–process` är `–grammar` och du måste använda syntaxen `fmresultset` eller syntaxen `FMPXMLRESULT`.

Till exempel:

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–process
```

## Överföra information mellan formatmallar med hjälp av token

Du kan använda frågeparametern `-token` i en URL-adress eller som ett statiskt definierat frågekommando när du vill överföra användardefinierad information mellan formatmallar utan att använda sessioner eller cookies. Frågeparametern `-token` är valfri med alla frågekommandon.

Det användardefinierade parametervärdet kan vara vilken teckensträng som helst som är URL-kodad. Till exempel:

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset
&-token.D100=Pending&-findall
```

Se ”`-token`.[string] (överför värden mellan XSLT-formatmallar), frågeparameter” på sidan 106.

**Viktigt!** Använd inte frågeparametern `-token` för att överföra privat information.

Om du vill hämta värdet i frågeparametern `-token` använder du programsatsen `<xsl:param name="request-query" />`. Se ”Använda frågeinformation i en förfrågan” på sidan 62.

## Använda FileMaker-tilläggsfunktioner och -tilläggsparametrar för XSLT

FileMaker-tilläggsfunktionerna för XSLT är utformade för `fmxml`-namnuttrycket. Se till att ta med en deklaration av namnuttrycket `fmxml` i `<xsl:stylesheet>`-elementet i början av XSLT-formatmallen. Se ”Namnuttryck och prefix för XSLT-formatmallar för FileMaker” på sidan 57.

FileMaker-tilläggsfunktionerna för XSLT har utformats för att du ska kunna använda dem i en XSLT-formatmall genom att ange dem som ett funktionsanrop i ett Xpath-uttryck. Xpath-uttryck används som värden för attributen `select` och `test` i flera XSLT-element.

Anta att du vill kontrollera huvudet User-Agent för att se vilken webbläsare som används. När du gör detta kanske du vill använda en variabel som innehåller värdet i huvudet User-Agent:

```
<xsl:variable name="user-agent" select="fmxml:get_header('User-Agent')"/>
```

För tilläggsfunktionerna som returnerar ett värde returneras värdet i den angivna XSLT-typen. Många funktioner returnerar strängar men det är ett fåtal funktioner som returnerar en `node-set` som kan överföras.

**Obs!** I det här avsnittet beskrivs FileMaker-tilläggsfunktioner och -parametrar för XSLT. Avsnittet innehåller också några exempel. Fler exempel på varje funktion finns i referensen över FileMaker-tilläggsfunktioner för XSLT. Se ”Om referens till FileMaker-tilläggsfunktionerna för XSLT” på sidan 54.

## FileMaker-specifika XSLT-parametrar som anges av Web Publishing Engine

Vid bearbetning av en förfrågan ställer Web Publishing Engine dynamiskt in värdena för följande FileMaker-specifika XSLT-parametrar. Du kan använda värdena i dessa parametrar i formatmallen genom att använda elementet `<xsl:param>`.

FileMaker-specifik XSLT-parameter	Mer information finns i
<code>&lt;xsl:param name="request-query"/&gt;</code>	”Använda frågeinformation i en förfrågan” i nästa avsnitt.
<code>&lt;xsl:param name="client-ip"/&gt;</code> <code>&lt;xsl:param name="client-user-name"/&gt;</code> <code>&lt;xsl:param name="client-password"/&gt;</code>	”Hämta klientinformation” på sidan 62.
<code>&lt;xsl:param name="xml-base-uri"/&gt;</code>	”Använda Web Publishing Engines grundläggande URI-parameter” på sidan 63.
<code>&lt;xsl:param name="authenticated-xml-base-uri"&gt;</code>	”Använda den verifierade grundläggande URI-parametern” på sidan 63.

## Använda frågeinformation i en förfrågan

Du kan använda en XSLT-parameter när du vill använda frågeinformation i en förfrågan i en URL-adress eller i HTML-formulärdata. Du kan exempelvis använda den aktuella frågeinformationen för att visa den aktuella placeringen i en grupp hittade poster och skapa länkar till föregående och nästa post.

Följande XSLT-parameter ger åtkomst till alla frågekommandon och frågeparametrar som du använder för att efterfråga XML-data via Web Publishing Engine:

```
<xsl:param name="request-query"/>
```

Med undantag av fältnamn returnerar Web Publishing Engine alla frågekommando- och frågeparameternamn med små bokstäver. Versalerna i fältnamn bevaras.

Du läser in ett XML-dokumentfragment i förfrågningens frågeparameter med följande syntax:

```
<!DOCTYPE query [
  <!ELEMENT query (parameter)*>
  <!ATTLIST frågeåtgärd CDATA #REQUIRED>
  <!ELEMENT parameter (#PCDATA)>
  <!ATTLIST parameternamn CDATA #REQUIRED>
]
```

**Obs!** Frågeinformationen ska anges i namnuttrycket `fmq="http://www.filemaker.com/xml/query"`. Se till att ta med en deklaration av namnuttrycket `fmq` i `<xsl:stylesheet>`-elementet i början av XSLT-formatmallen. Se "Namnuttryck och prefix för XSLT-formatmallar för FileMaker" på sidan 57.

Anta att du vill visa frågekommandona och frågeparametrarna i denna förfrågan:

```
http://192.168.123.101/fmi/xsl/my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset&-token.1=abc123&-findall
```

Om du tar med uttrycket `<xsl:param name="request-query" />` före malldelen, sparar Web Publishing Engine detta XML-dokumentfragment i den parametern:

```
<query action="my_stylesheet.xml" xmlns="http://www.filemaker.com/xml/query">
  <parameter name="-db">products</parameter>
  <parameter name="-lay">sales</parameter>
  <parameter name="-grammar">fmresultset</parameter>
  <parameter name="-token.1">abc123</parameter>
  <parameter name="-findall"></parameter>
</query>
```

Du kan därefter använda parametern `request-query` för att visa värdet i en token som överfördes i en URL-adress via ett Xpath-uttryck. Till exempel:

```
$request-query/fmq:query/fmq:parameter[@name = '-token.1']
```

## Hämta klientinformation

Du kan använda följande XSLT-parametrar för FileMaker när du vill hämta information från Web Publishing Engine om en webbklients IP-adress, användarnamn och lösenord:

```
<xsl:param name="client-ip"/>
<xsl:param name="client-user-name"/>
<xsl:param name="client-password">
```

Ta med dessa parameteruttryck i XSLT-formatmallen före det översta `<xsl:template>`-elementet.

Dessa parametrar erbjuder uppgifter om webbanvändaren när en formatmall programmässigt läser in ytterligare lösenordsskyddade XML-dokument. Mer information finns i avsnittet ”Läsa in ytterligare dokument” på sidan 64. Webbanvändaren måste ange användarnamn och lösenord i dialogrutan för grundläggande HTTP-verifiering. Se ”Få åtkomst till en skyddad databas” på sidan 19.

Mer information och exempel på hur du använder dessa tre FileMaker XSLT-parametrar finns i referensen över FileMaker-tilläggfunktioner för XSLT.

## Använda Web Publishing Engines grundläggande URI-parameter

Web Publishing Engine anger den URI-parameter (URI = Uniform Resource Identifier) som ska vara värd och port där Web Publishing Engine installeras. Den grundläggande URI-parametern tillåter att förfrågningar om XML-data från FileMaker-databaser bestäms, i förhållande till Web Publishing Engines värd.

Om du vill använda Web Publishing Engines grundläggande UI inkluderar du följande uttryck i XSLT-formatmallen före det översta <xsl:template>-elementet:

```
<xsl:param name="xml-base-uri"/>
```

Du kan därefter använda den grundläggande URI-parametern för aktuell formatmall via variabeln \$xml-base-uri när du vill göra ytterligare förfrågningar om FileMaker XML-data. Du kan exempelvis använda den grundläggande URI-parametern i följande förfrågan om ytterligare XML-data:

```
<xsl:variable name="layout_information" select="document(concat($xml-base-uri,'/fmi/xml/FMPXMMLAYOUT.xml?
-db=products&-lay=sales&-view'))" />
```

## Använda den verifierade grundläggande URI-parametern

Parametern authenticated-xml-base-uri kombinerar funktionerna hos parametrarna client-user-name och client-password med parametern xml-base-uri:

```
<xsl:param name="authenticated-xml-base-uri"/>
```

Använd den här parametern när du vill läsa in ytterligare lösenordsskyddade XML-dokument som kräver samma användarnamn och lösenord som angavs i den ursprungliga förfrågan som för närvarande håller på att bearbetas. Exempel på detta finns i nästa avsnitt, ”Läsa in ytterligare dokument”.

Ta med detta parameteruttryck i XSLT-formatmallen före det översta <xsl:template>-elementet.

Om värdena för parametrarna client-user-name och client-password inte är tomma, är värdet för den verifierade parametern -authenticated-xml-base-uri:

```
http://username:password@hostname:port
```

Om värdena för parametrarna client-user-name och client-password inte är tomma, är värdet för parametern authenticated-xml-base-uri samma som värdet för parametern xml-base-uri.

## Läsa in ytterligare dokument

Om du vill läsa in ytterligare ett XML-dokument under bearbetningen av en XSLT-formatmall använder du den förvalda XSLT-funktionen `document()` med en URI till XML-dokumentet. Funktionen `document()` returnerar den begärda XML-koden som en `node-set` som kan lagras i ett `<xsl:variable>`-element.

Om du vill läsa in ett XML-dokument som innehåller data från en FileMaker-databas, använder du funktionen `document()` tillsammans med frågekommandot och frågeparametrarna. Till exempel:

```
<xsl:variable name="other-data" select="document(concat($xml-base-uri, '/fmi/xml/FMPXMMLAYOUT.xml?
-db=products&-lay=sales&-view'))" />
```

Om du vill läsa in ytterligare lösenordsskyddade XML-dokument som kräver samma användarnamn och lösenord som angavs i den ursprungliga förfrågan som för närvarande håller på att bearbetas, använder du parametern `authenticated-xml-base-uri`. Denna parameter anger samma användarnamn och lösenord som en del av den URI som överförs till funktionen `document()`.

Till exempel:

```
<xsl:variable name="other-data" select="document(concat($authenticated-xml-base-uri,
'/fmi/xml/FMPXMMLAYOUT.xml?-db=products&-lay=sales&-view'))"/>
```

Om du vill läsa in ett lösenordsskyddat XML-dokument som kräver ett annat användarnamn och lösenord än vad som angavs i den överordnade förfrågan, använder du följande syntax för att ange användarnamn och lösenord som en del av den URI överförs till funktionen `document()`:

```
http://username:password@hostname/path?querystring
```

Om du vill läsa in ett XML-dokument som inte är baserat på en FileMaker-databas, använder du funktionen `document()` utan frågekommandon och frågeparametrar. Till exempel:

```
<xsl:variable name="other-data" select="document('http://server.company.com/data.xml')" />
```

Om du använder funktionen `document()` med en relativ URL-adress försöker Web Publishing Engine att läsa in XML-dokumentet från det lokala filsystemet på den plats som är relativ till den plats där formatmallen lagras. Anta att en formatmall som är lagrad i mappen `mystylesheets` i mappen `xslt-template-files` innehåller följande `document()`-funktion med en relativ URL-adress:

```
<xsl:variable name="mydoc" select="document('mystylesheets/mydoc.xml')" />
```

Ett försök görs att läsa in filen `mydoc.xml` från mappen `mystylesheets` i mappen `xslt-template-files` i det lokala filsystemet.

**Obs!** När du använder Web Publishing Engines grundläggande URI för att läsa in ett dokument, stöder Web Publishing Engine endast HTTP. När du läser in ett dokument från en extern server stöder Web Publishing Engine både HTTP och HTTPS.

## Använda layoutinformation för en databas i en formatmall

Du kan införliva layoutinformationen från en FileMaker-databas i en formatmall genom att använda syntaxen FMPXMLLAYOUT och efterfråga informationen och sedan läsa in den till en variabel via XSLT-funktionen `document()`:

```
<xsl:variable name="layout" select="document(concat($xml-base-uri,'/fmi/xml/FMPXMLLAYOUT.xml?–view'))" />
```

Anta att du vill skapa en listruta för ett fält med namnet `Color`. Listan fylls med värdena från en värdelista med två fält med namnet `shirts` som har definierats i en layout i en FileMaker-databas. Anta att det första fältet i värdelistan med två fält lagrar ID-nummer för färg (t.ex. "100") och det andra fältet lagrar färgens associerade namn (t.ex. "Ljusgrön"). Så här kan du använda funktionen `document()` när du vill läsa in layoutinformation till en XSLT-variabel tillsammans med `DISPLAY`-attributet för att visa värdet från det andra fältet i en värdelista med två fält :

```
<xsl:variable name="layout" select="document(concat($xml-base-uri,'/fmi/xml/FMPXMLLAYOUT.xml?–db=products
&–lay=sales&–view'))" />
```

```
<select size="1">
```

```
  <xsl:attribute name="name">Color</xsl:attribute>
```

```
  <option value="">Select One...</option>
```

```
  <xsl:for-each select="$layout/fmi:FMPXMLLAYOUT/fmi:VALUELISTS/fmi:VALUELIST[@NAME = 'shirts']/
fmi:VALUE">
```

```
    <tillval ...>
```

```
      <xsl:attribute name="value"><xsl:value-of select="."/></xsl:attribute>
```

```
      <xsl:value-of select="@DISPLAY"/>
```

```
    </option>
```

```
  </xsl:for-each>
```

```
</select>
```

## Använda innehållsbuffering

När innehållsbuffering är inaktiverad skickar Web Publishing Engine resultatet av en XSLT-konvertering direkt tillbaka till klienten. Innehållsbuffering är alltid inaktiverad, såvida du inte specifikt har aktiverat den. Om du aktiverar innehållsbuffering lagrar Web Publishing Engine det konverterade innehållet tills hela konverteringen är klar.

Innehållsbuffering krävs för XSLT-formatmallar som ändrar huvuden. Eftersom huvuden skrivs före huvuddelen måste huvuddelen buffras så att den tillagda huvudinformatonen kan läggas till.

Det finns fyra tilläggsfunktioner i FileMaker som kräver XSLT-konvertering för att kunna buffras:

- `fmxslt:create_session()`: Se "Använda tilläggsfunktionerna för sessioner" på sidan 67.
- `fmxslt:set_header()`: Se "Använda sidhuvudfunktionerna" på sidan 70.
- `fmxslt:set_status_code ()`: Se "Använda sidhuvudfunktionerna" på sidan 70.
- `fmxslt:set_cookie()`: Se "Använda tilläggsfunktioner för cookies" på sidan 71.

Dessa tilläggfunktioner fungerar bara om du inkluderar följande XSLT-bearbetningsinstruktion i det översta dokumentet för förfrågan:

```
<?xslt-cwp-buffer buffer-content="true"?>
```

**Viktigt!** Om du har en grundläggande formatmall som innehåller en annan formatmall måste den grundläggande formatmallen innehålla bearbetningsinstruktionen `<?xslt-cwp-buffer?>`. Denna instruktion ignoreras om den används i en inkluderad formatmall.

En fördel med att buffra svaret med hjälp av bearbetningsinstruktionen är att Web Publishing Engine kan fastställa längden på svaret och ange huvudet Content-Length i svaret. Buffring av svar kan försämra Web Publishing Engine prestanda.

## Lagra information mellan förfrågningar med hjälp av Web Publishing Engine

Du kan följa upp och lagra olika typer av information mellan förfrågningar med hjälp av Web Publishing Engine. Med sessioner kan du skapa ett webbprogram som kan bibehålla lägen genom att använda beständiga godtyckliga informationsbitar mellan förfrågningar. Användarklientinformation som anges på en första formulärsida kan exempelvis lagras i en session och sedan användas för att mata in värden på en efterföljande sida.

Enligt standardinställningen använder Web Publishing Engine en cookie för att lagra sessions-ID. Du kan underlätta för klienter som inte tillåter cookies genom att lägga till sessions-ID i URL-adressen med hjälp av funktionen `fmxml:session_encode_url()`. Om du vill garantera kompatibiliteten i alla situationer bör du koda alla URL-adresser som är skrivna på sidan med funktionen `fmxml:session_encode_url()`. Denna funktion lägger till den semikolonsavgränsade parametern `jsessionId` i URL-adressen. Parametern fungerar som identifierare för klientens överordnade session.

I stället för att placera följande på en sida:

```
<a href="my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset&-findall">hyperlinked text</a>
```

kan du koda alla länkar på en sida enligt följande:

```
<a href="{fmxml:session_encode_url('my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset
&-findall')}">hyperlinked text</a>
```

Om klienten inte tillåter cookies innehåller sidan:

```
<a href="my_stylesheet.xml;jsessionId=<session id>?-db=products&-lay=sales&-grammar=fmresultset&-findall">
hyperlinked text</a>
```

Om Web Publishing Engine upptäcker att klienten tillåter cookies lagrar funktionen `fmxml:session_encode_url()` sessions-ID i en cookie i stället för URL-adressen.

**Obs!** Sessionsinformation finns inte kvar efter att Web Publishing Engine har startas om.

## Använda tilläggfunktionerna för sessioner

Hantera sessionsvariabler med följande tilläggfunktioner för sessioner. Du kan spara en sträng, ett tal, ett logiskt värde eller en `node-set` i ett sessionobjekt. Genom att använda en `node-set` kan du skapa en datastruktur i XML och sedan spara den mellan förfrågningar i sessionsobjektet.

Sessionstilläggfunktion	Returerad datatyp	Beskrivning
<code>fmxls:session_exists(String session-name)</code>	logisk	Kontrollerar om en session med det angivna namnet finns.
<code>fmxls:create_session(String sessionsnamn)</code>	logisk	Skapar en session med det angivna sessionsnamnet och standardtimeout, som anges via Admin Console. Se FileMaker Server Hjälp. <b>Obs!</b> Denna funktion kräver bearbetningsinstruktionen <code>&lt;?xslt-cwp-buffer?&gt;</code> . Se ”Använda innehållsbuffering” på sidan 65.
<code>fmxls:invalidate_session(String session-name)</code>	logisk	Tvingar sessionen att genast avbrytas.
<code>fmxls:set_session_timeout(String session-name, Number timeout)</code>	logisk	Anger sessionens timeout i sekunder. Standardtimeouten för sessioner anges i Admin Console.
<code>fmxls:session_encode_url(String url)</code>	sträng	Kodar en URL-adress med sessions-ID om klienten inte stöder cookies, annars returneras den inmatade URL-adressen.
<code>fmxls:set_session_object(String session-name, String name, Object value)</code>	XSLT-objekt (tal, sträng, logisk eller noduppsättning)	Sparar ett XSLT-objekt (tal, sträng, logiskt eller <code>node-set</code> ) under en session. De kan senare hämtas med funktionen <code>fmxls:get_session_object()</code> . Den här funktionen returnerar även det tidigare sparade objektet under det angivna sessionsobjektnamnet. Om inget sparades under namnet returneras ett Null-objekt. <b>Obs!</b> Tilläggfunktionen <code>set_session_object()</code> kan bara lagra strängvärden och kommer att tolka alla objekt som skickas till den som en sträng. Om objektet inte kan konverteras till en sträng, lagras inte något värde i sessionen och tilläggfunktionens felkod ställs in på 10100 (okänt sessionsfel). Om du försöker ställa in ett sessionsobjekt och använder noll eller en tom sträng kommer felkoden 10100 att visas (okänt sessionsfel). När du vill rensa bort en sessionsvariabel tar du bort den från sessionen med hjälp av funktionen <code>remove_session_object()</code> .
<code>fmxls:get_session_object(String session-name, String name)</code>	XSLT-objekt	Hämtar ett XSLT-objekt från sessionen.
<code>fmxls:remove_session_object(String session-name, String name)</code>	XSLT-objekt	Returnerar och tar sedan bort ett XSLT-objekt från sessionen.

Här är ett exempel på hur du skapar en session och sedan sparar en färg i sessionen:

```
<xsl:variable name="session">
  <xsl:choose>
    <xsl:when test="not (fmxls:session_exists(string($session-name))">
      <xsl:value-of select="fmxls:create_session(string($session-name))"/>
    </xsl:when>
    <xsl:otherwise>true</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
<xsl:variable name="favorite-color" select="fmxls:set_session_object(string($session-name), 'favorite-color', string($color))"/>
```

### Viktigt!

- Om du vill vara säker på att användarna loggar ut från en databas efter en session använder du funktionen `fmxls:invalidate_session ()` och tvingar sessionen att omedelbart avbrytas.
- Om du använder globala fält eller ett manus som anger eller förändrar ett läge måste du använda Admin Console och aktivera alternativet XSLT-databassessioner för Web Publishing Engine. I annat fall bevaras inte globala fältvärden och lägen mellan förfrågningar. Se FileMaker Server Hjälp.
- Om du växlar till en annan databas medan du använder Web Publishing Engine-sessioner kommer globala fältvärden inte att bevaras. Web Publishing Engine stänger först filen och öppnar sedan den andra filen. Som ett alternativ kan du få åtkomst till data från den andra databasfilen genom att använda en layout i den första databasfilen.

## Skicka e-postmeddelanden från Web Publishing Engine

Du kan skapa e-postmeddelanden med Web Publishing Engine, vilket är praktiskt när det gäller anpassade webblösningar. Om du vill att Web Publishing Engine ska skicka ett e-postmeddelande använder du en av följande tre `fmxls:send_email ()`-tilläggsfunktioner i en XSLT-formatmall. Du kan skicka ett eller flera separata meddelanden med dessa funktioner. Eftersom `fmxls:send_email ()`-funktionerna finns i Web Publishing Engines XSLT-formatmall på serversidan kan inte en klient använda en Web Publishing Engine för att skicka obehöriga e-postmeddelanden.

Tilläggsfunktion för e-post	Returerad datatyp	Beskrivning
<code>fmxls:send_email</code> (smtp-fält som String, meddelande som String)	logisk	Skickar ett vanligt e-postmeddelande i valfri längd från Web Publishing Engine med Web Publishing Engines standardtextkodning för e-postmeddelanden
<code>fmxls:send_email</code> (smtp-fält som String, meddelande som String, kodning som String)	logisk	Skickar ett vanligt e-postmeddelande i valfri längd med en av följande textkodningar: US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS, UTF-8. Mer information om dessa kodningar finns i "Ange textkodning för förfrågningar" på sidan 59.
<code>fmxls:send_email</code> (smtp-fält som String, xslt-fil som String, xml som Node, ta_med_bild som Boolean)	logisk	Skickar ett HTML-baserat e-postmeddelande med den kodning som är angiven i kodningsattributet i <code>&lt;xsl:output&gt;</code> -elementet i formatmallen. Om kodningsattributet inte finns med i <code>&lt;xsl:output&gt;</code> -elementet, används Web Publishing Engines standardtextkodning för e-postmeddelanden.

**Obs!**

- I var och en av de tre formerna av funktionen `fmxsst:send_email()` är parametern `smtpFields` en URL-kodad sträng i valfri längd som innehåller adressen och ämnesinformation i följande format, som är baserat på RFC 2368, URL-protokollet `mailto:`

`användarnamn@värd?namn1=värde1&namn2=värde2...`

där `användarnamn@värd` anger en mottagare. Namn-/värdeparen kan anges i vilken ordning som helst och definieras enligt följande:

- `from=username@host` (får endast förekomma en gång). Fältet `from` måste anges.
- `to=username@host`. Använd detta namn-/värdepar för ytterligare mottagare.
- `reply-to=username@host` (får endast förekomma en gång)
- `cc=användarnamn@värd`
- `bcc=användarnamn@värd`
- `subject=string` (får endast förekomma en gång)

Om fälten `from`, `reply-to` eller `subject` anges mer än gång skickas inte e-postmeddelandet, värdet `false()` returneras av funktionen och en felstatuskod uppstår.

- Web Publishing Engine kontrollerar syntaxen i alla e-postadresser. De måste vara i följande format:  
`user@host.tld` or `"quoted identifier"<user@host.tld>`  
där `tld` är en domän på översta nivån, till exempel `com` eller `net`. Om något av fälten innehåller en ogiltig e-postadress skickas inte meddelandet och en felstatuskod uppstår.
- De olika värdena för parametern `smtp-fält`, exempelvis ämne, måste vara en URL-kodad sträng. Du måste exempelvis skriva tecknet `"&"` som `"&amp;"` och blanksteg som `"%20"`. Hela strängen för parametern `smtp-fält` måste vara XML-kodad. (Se exemplet i slutet av avsnittet.)
- För var och en av dessa funktioner returneras värdet `true()` om e-postmeddelandet skickas och värdet `false()` om det inte skickas.
- För engelska e-postmeddelanden använder Web Publishing Engine den förvalda textkodningen ISO-8859-1. Du kan ändra denna inställning med Admin Console. Se FileMaker Server Hjälp.

- Funktionen `fmxml:send_email(String smtpFields, String xsltFile, Node xml, boolean includelimages)` skickar ett e-postmeddelande med XML-data som bearbetas av e-postformatmallen som du angav i den här funktionen.
  - För parametern `xslt-fil` anger du namnet på e-postformatmallen genom att ange en URL-adress som är relativ till den formatmall som bearbetas för förfrågan.
  - För parametern `xml` anger du den överordnade noden för de XML-data du vill använda med e-postformatmallen. Om du vill skicka ett e-postmeddelande med samma XML-data som visas i webbläsaren anger du bara XPath för roten till dokumentet: `"/`. I annat fall kan du använda ett annat XML-dokument genom att först läsa in det med funktionen `document()` och sedan överföra dokumentet till funktionen `fmxml:send_email()`.
  - För parametern `ta_med_bilder` anger du det logiska värdet `true()` om du vill att Web Publishing Engine ska ta med alla bilder som har angetts i HTML-koden i e-postmeddelandet som bilagor. Denna parameter innehåller både bilderna från FileMaker-databasen och bilder från andra platser. Web Publishing Engine ändrar bildens URL-adress så att den pekar på bilagorna. Om bildfilerna är många eller stora kan prestanda försämrats. Om du anger `false ()` ändrar inte Web Publishing Engine bildernas URL-adresser. Om URL-adresserna är absoluta försöker e-postklienten att läsa in bilderna från webbservern.

Här följer ett exempel på hur du använder funktionen `fmxml:send_email(smtp-fält som String, xslt-fil som String, xml som Node, ta_med_bilder som Boolean)` inuti ett XPath-uttryck, till exempelvis inuti `<xsl:if>`-elementet:

```
fmxml:send_email('tom_jones@company.com?subject=project%20status&from=john_smith@company.com
&cc=jane_doe@company.com','my_mail_template.xml',/, true())
```

Information om hur du konfigurerar Web Publishing Engine så att den ansluter till en SMTP-server finns i hjälpen till FileMaker Server.

## Använda sidhuvudfunktionerna

Du kan använda funktionen `fmxml:get_header()` om du vill läsa information från HTTP-förfrågan och svarssidhuvudena och funktionen `fmxml:set_header()` när du vill skriva information i sidhuvudena. Dessa funktioner är användbara om klienten kan använda informationen i sidhuvuden för att hämta information från webbservern, eller om du måste ange ett HTTP-huvud av andra orsaker.

Tilläggsfunktion för sidhuvud	Returerad datatyp	Beskrivning
<code>fmxml:get_header(String name)</code>	sträng	Returerar det angivna sidhuvudsvärdet
<code>fmxml:set_header(String name, String value)</code>	ogiltig	Anger det angivna sidhuvudsvärdet
<code>fmxml:set_status_code(Number status-code)</code>	ogiltig	Anger HTTP-statuskoden

### Obs!

- Namnet som används i funktionerna `fmxml:get_header()` och `fmxml:set_header()` samt värdet i funktionen `fmxml:set_header()` kan vara en sträng i valfri längd.
- Funktionen `fmxml:set_header()` och funktionen `fmxml:set_status_code()` kräver bearbetningsinstruktionen `<?xslt-cwp-buffer?>`. Se ”Använda innehållsbuffering” på sidan 65.

Följande exempel visar hur du anger värdet för sidhuvudet. Anta att du använder en formatmall för att skapa ett vCard. Ett problem uppstår när en webbläsare försöker läsa in formatmallssidan och tolkar filen .xsl som en formatmall i stället för ett vCard. Om du använder sidhuvudet som kallas Content-Disposition kan du ange att det finns en bilaga med tillägget .vcf.

```
<xsl:value-of select="fmxslt:set_header('Content-Disposition','attachment;filename=test.vcf')"/>
```

## Använda tilläggsfunktioner för cookies

Du kan använda tilläggsfunktionerna för cookies när du vill hämta eller ange cookies som är lagrade i klientens webbläsare.

Tilläggsfunktion för cookie	Returerad datatyp	Beskrivning
fmxslt:get_cookie(String name)	noduppsättning	Returnerar COOKIES node-set med det angivna cookie-namnet.
fmxslt:get_cookies()	noduppsättning	Returnerar COOKIES node-set med alla cookies tillhandahållna av klienten.
fmxslt:set_cookie(String name, String value)	ogiltig	Sparar den angivna cookien i klientens webbläsare med det angivna värdet.
fmxslt:set_cookie(String name, String value, Number expires, String path, String domain)	ogiltig	Sparar den angivna cookien i klientens webbläsare med alla tillgängliga värdena för en cookie. Parametern utgår är antalet sekunder innan cookien går ut.

### Obs!

- Funktionerna fmxslt:get\_cookie() och fmxslt:get\_cookies() returnerar ett node-set i följande struktur:
 

```
<!ELEMENT cookies (cookie)*>
      <!ATTLIST cookie xmlns CDATA #FIXED "http://www.filemaker.com/xml/cookie">
      <!ELEMENT cookie (#PCDATA)>
      <!ATTLIST cookie name CDATA #REQUIRED>
```
- Namnuttrycket för XML för cookie node-set är "http://www.filemaker.com/xml/cookie". Försäkra dig om att du deklarerar namnuttrycket och anger ett prefix för det.
- Alla parametervärden för funktionerna fmxslt:set\_cookie måste vara giltiga, annars ignorerar webbläsaren fmxslt:set\_cookie-funktionsförfrågningarna.
- Strängparametrarna kan vara hur långa som helst. Det gäller alla cookie-funktioner.
- Båda formerna av funktionen fmxslt:set\_cookie() kräver bearbetningsinstruktionen <?xslt-cwp-buffer?>. Se "Använda innehållsbuffering" på sidan 65.

### Exempel: get\_cookie

Följande exempel hämtar en cookie med namnet preferences samt dess värde:

```
<xsl:variable name="pref_cookie" select="fmxslt:get_cookie('preferences')"/>
<xsl:value-of select="concat('Cookie Name = ', $pref_cookie/fmc:cookies/fmc:cookie/@name)"/> <br/>
<xsl:value-of select="concat('Cookie Value = ', $pref_cookie/fmc:cookies/fmc:cookie)"/>
```

### Exempel: set\_cookie

Här är ett exempel på hur du anger en cookie med alla värden:

```
<xsl:variable name="storing_cookie" select="fmxslt:set_cookie ('text1', 'text2', 1800, 'my_text', 'my.company.com')"/>
```

## Använda tilläggsfunktionerna för stränghantering

Du kan ändra kodningen på strängar i vilken längd som helst med hjälp av stränghanteringsfunktionerna.

Tilläggsfunktioner för stränghantering	Returerad datatyp	Beskrivning
fmxml:break_encode(String value)	sträng	Returnerar en HTML-brytkodad sträng. Tecken som till exempel "&" (ampersandtecken) ersätts med "&amp;". Tecken för ny rad, till exempel radmatningar och vagnreturer, ersätts med  . Denna funktion fungerar endast om attributet disable-output-escaping i elementen <xsl:value-of> och <xsl:text> sätts till "yes" (disable-output-escaping="yes"). <b>Obs!</b> Om du vill inkludera en radmatning eller vagnretur i strängen som funktionen fmxml:break_encode() tillämpas på, måste du använda följande escape-tecken i strängen: "&#10;" (för radmatning) eller "&#13;" (för vagnretur). Du kan inte infoga en radmatning eller vagnretur i strängen genom att trycka på tangenten Retur i textredigeringsprogrammet.
fmxml:html_encode(String value)	sträng	Returnerar en HTML-kodad sträng. Tecken som "&" (ampersandtecken) ersätts med "&amp;".
fmxml:url_encode(String value)	sträng	Returnerar en URL-kodad sträng. URL-kodning används för att överföra tecken över Internet, speciellt för URL-adresser. Ett exempel är att "&" (ampersand) i ett URL-kodat format är %26. Om ett reserverat tecken används i din href ska du använda denna funktion för att URL-koda strängen.
fmxml:url_encode(String value, String encoding)	sträng	Returnerar en URL-kodad sträng med den teckenkodning du angav för parametern kodning, som kan vara: US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS eller UTF-8. Använd den här funktionen i situationer där du vet att en webbläsare förväntar sig en annan teckenkodning än den som används i den aktuella formatmallen. Startsidan på webbplatsen kan exempelvis visas i UTF-8, men användarna kan klicka på en länk om de vill gå till en japansk sida. Om förfrågan innehåller japanska tecken och japanska sidor använder Shift_JIS-kodning, är det bäst att koda strängen i Shift_JIS.
fmxml:url_decode(String value)	sträng	Returnerar en URL-avkodad sträng från en URL-sträng som tidigare var kodad.
fmxml:url_decode(String value, String encoding)	sträng	Returnerar en URL-avkodad sträng med den teckenkodning du angav för parametern kodning, som kan vara: US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS eller UTF-8. Använd den här funktionen i situationer där du måste ange teckenkodningen som används i en URL-kodad sträng för att strängen ska kunna avkodas på rätt sätt. Även om din webbplats använder ISO-8859-1 kan användarna skicka ett formulär med en annan teckenkodning.

## Jämföra strängar med reguljära Perl 5-uttryck

Du kan jämföra strängar med hjälp av reguljära Perl 5-uttryck genom att använda tilläggsfunktionen `fmxmlt:regex_contains()`. Genom att jämföra reguljära uttryck, som är en avancerad typ av textmatching, kan du se om en sträng matchar ett visst mönster. Syntaxen för funktionen är:

```
fmxmlt:regex_contains(indata som String, mönster som String)
```

där `indata` är en sträng och `mönster` ett reguljärt Perl 5-uttryck. Mer information om syntaxen i reguljära Perl 5-uttryck finns på [www.perldoc.com](http://www.perldoc.com). Funktionen `fmxmlt:regex_contains()` returnerar ett logiskt värde.

Den här funktionen är användbar om du vill använda en mer avancerad stränghantering än den som finns i standard-XSLT. Du kan exempelvis kontrollera om ett fältvärde innehåller ett giltigt telefonnummer eller en giltig e-postadress genom att jämföra strängen mot ett reguljärt Perl 5-uttryck.

Här är ett exempel på hur du kan använda denna funktion för att kontrollera om ett fältvärde innehåller felaktiga e-postadresser:

```
<xsl:variable name="email" select="foo@bar.com"/>
<xsl:if test="fmxmlt:regex_contains($email, '^\\w+([\\w-\\.]*\\@\\w+((-\\w+)|(\\w*))\\.([a-z]{2,3})$')">Valid Email</xsl:if>
```

Om Web Publishing Engine inte kan tolka mönstret sätts felstatus till felkoden 10311. Se ”Felkodsnummer för FileMaker-tilläggsfunktioner för XSLT” på sidan 115.

## Söka efter värden i ett fält som är formaterat som en kryssruta

Du kan använda följande tilläggsfunktion om du vill kontrollera om ett visst värde i en kryssrutans värdelista är sparad i ett fält i FileMaker-databasen:

```
fmxmlt:contains_checkbox_value(värdesträng som String, värdelistpost som String)
```

där `värdesträng` är en XPath som anger fältet och `värdelistpost` är värdet du vill söka efter.

Om det angivna värdet är sparad i fältet returnerar denna logiska funktion `true()`. Annars returneras `false()`. Du kan använda funktionen för att ta reda på om en kryssruta ska vara markerad eller ej i ett HTML-formulär.

Antag att ett fält i en FileMaker-databaslayout har följande alternativ för kryssrutor:

- Red
- Blue
- Green
- Small
- Medium
- Large

Om en användare endast valde Red, skulle fältet innehålla strängen ”Red”. Om du vill se om fältet innehåller ”Blue”, kan du använda följande funktionsanrop:

```
fmxmlt:contains_checkbox_value(<fältvärdesnod>,'Blue')
```

där `<fältvärdesnod>` är XPath till elementet `<data>` för kryssrutans fält. Funktionen returnerar ”false” i det här exemplet.

Ett vanligt sätt att använda denna funktion är att visa kryssrutans värdelista på en webbsida och markera kryssrutorna på webbsidan som är markerad i databasen. Följande HTML- och XSLT-uttryck skapar exempelvis ett antal kryssrutor för ett fält med namnet style med hjälp av en värdelista med namnet color\_size:

```
<xsl:variable name="field-value" select="fmrs:field[@name='style']/fmrs:data" />
<xsl:for-each select="$valueLists[@NAME = 'color_size']/fml:VALUE">
  <input type="checkbox">
    <xsl:attribute name="name">style</xsl:attribute>
    <xsl:attribute name="value"><xsl:value-of select="."/></xsl:attribute>
    <xsl:if test="fmxslt:contains_checkbox_value($field-value,.)">
      <xsl:attribute name="checked">checked</xsl:attribute>
    </xsl:if>
  </input><xsl:value-of select="."/><br/>
</xsl:for-each>
```

HTML- och XSLT-uttrycken i exemplet resulterar i följande kryssrutor på en webbsida, med Red och Medium markerade:

```
[x] Red
[] Blue
[] Green
[] Small
[x] Medium
[] Large
```

## Använda tilläggsfunktioner för datum, tid och dag

Du kan använda tilläggsfunktionerna om du vill visa aktuellt datum, aktuell tid eller aktuell dag och när du vill jämföra två datum, klockslag eller dagar.

Funktionerna i följande tabell använder formatet "fm" oavsett nationella inställningar. "fm"-formaten är MM/dd/yyyy för datum, HH:mm:ss för tid och MM/dd/yyyy HH:mm:ss för tidsstämpel.

Om du vill arrangera om utdatavärden till ett annat format du föredrar, använder du beräkningsfunktioner eller JavaScript.

Tilläggsfunktion för datum, tid och dag	Returerad datatyp	Beskrivning
fmxls:get_date()	sträng	Returnerar det aktuella datumet i "fm"-format.
fmxls:get_date(String format)	sträng	Returnerar det aktuella datumet i det format du anger. Ange strängen "short", "long" eller "fm" för parametern format.
fmxls:get_time()	sträng	Returnerar den aktuella tiden i "fm"-format.
fmxls:get_time(String format)	sträng	Returnerar det aktuella klockslaget i det format du anger. Ange strängen "short", "long" eller "fm" för parametern format.
fmxls:get_day()	sträng	Returnerar den aktuella dagen i kort format.
fmxls:get_day(String format)	sträng	Returnerar den aktuella dagen i det format du anger. Ange strängen "short" eller "long" för parametern format.
fmxls:get_fm_date_format()	sträng	Returnerar formateringssträngen för "fm"-datumformatet: "MM/dd/yyyy"
fmxls:get_short_date_format()	sträng	Returnerar formateringssträngen för det korta datumformatet: "M/d/yy"
fmxls:get_long_date_format()	sträng	Returnerar formateringssträngen för det långa datumformatet: "MMM d, yyyy"
fmxls:get_fm_time_format()	sträng	Returnerar formateringssträngen för "fm"-tidsformatet: "HH:mm:ss"
fmxls:get_fm_timestamp_format()	sträng	Returnerar formateringssträngen för "fm"-tidstämpelformatet: "MM/dd/yyyy HH:mm:ss"
fmxls:get_short_time_format()	sträng	Returnerar formateringssträngen för det korta tidsformatet: "h:mm a"
fmxls:get_long_time_format()	sträng	Returnerar formateringssträngen för det långa tidsformatet: "h:mm:ss a z"
fmxls:get_short_day_format()	sträng	Returnerar formateringssträngen för det korta dagformatet: "EEE"
fmxls:get_long_day_format()	sträng	Returnerar formateringssträngen för det långa dagformatet: "EEEE"
fmxls:compare_date(String date1, String date2)	nummer	Denna funktion jämför två datumvärden. Den returnerar ett negativt tal om date1 är före date2. Den returnerar ett positivt tal om date1 är efter date2. Den returnerar 0 om date1 är identiskt med date2. Båda datumen måste anges i "fm"-datumformatet.
fmxls:compare_time(String time1, String time2)	nummer	Denna funktion jämför två tidsvärden. Den returnerar ett negativt tal om time1 är före time2. Den returnerar ett positivt tal om time1 är efter time2. Den returnerar 0 om time1 är identiskt med time2. Båda klockslagen måste anges i "fm"-tidsformatet.
fmxls:compare_day(String day1, String day2)	nummer	Denna funktion jämför två dagsvärden. Den returnerar ett negativt tal om day1 är före day2. Den returnerar ett positivt tal om day1 är efter day2. Den returnerar 0 om day1 är identiskt med day2. Båda dagarna måste anges i det korta dagformatet.

Funktionerna i följande tabell använder anpassade datumformateringssträngar som anger ett datum- och tidsformat. Se nästa avsnitt, "Datum- och tidsformatsträngar".

Tilläggsfunktion för datum, tid och dag	Returerad datatyp	Beskrivning
fmxls: get_datetime(String dateFormat)	sträng	Returerar aktuellt datum och klockslag med datum- och tidsformatsträngarna.
fmxls: convert_datetime(String oldFormat, String newFormat, String date)	sträng	Konverterar angivet datumet i angivet oldFormat till strängen enligt angivet newFormat. Strängarna oldFormat och newFormat måste anges med datum- och tidsformatsträngarna.
fmxls: compare_datetime(String dateFormat1, String dateFormat2, String date1, String date2)	nummer	Den här funktionen jämför date1 och date2 genom att avkoda datumerna enligt respektive datumformat. Den returnerar ett negativt tal om date1 är före date2. Den returnerar ett positivt tal om date1 är efter date2. Den returnerar 0 om date1 är identiskt med date2. Strängarna dateFormat1 och dateFormat2 måste anges med datum- och tidsformatsträngarna.

### Datum- och tidsformatsträngar

Datum- och tidsformaten anges med strängar för datum- och tidsmönster. Inom dessa strängar tolkas bokstäver utan citattecken, från A till Z och från a till ö som mönsterbokstäver som motsvarar komponenterna i en datum- eller tidssträng.

Följande mönsterbokstäver är definierade (alla andra tecken mellan A och Z och mellan a och z är reserverade):

Brev	Datum- eller tidskomponent	Visning	Exempel
G	Erabeteckning	Text	AD
y	År	År	1996; 96
M	Månad under år	Månad	Juli; Jul; 07
w	Vecka under år	Tal	27
W	Vecka i månad	Tal	2
D	Dag under år	Tal	189
d	Dag i månad	Tal	10
F	Veckodag i månad	Tal	2
E	Dag i vecka	Text	Tuesday; Tue
a	Am/pm-markör	Text	EM
H	Timme under dag (0-23)	Tal	0
k	Timme under dag (1-24)	Tal	24
K	Timme under am/pm (0-11)	Tal	0
h	Timme under am/pm (1-12)	Tal	12
m	Minut under timme	Tal	30
s	Sekund under minut	Tal	55
S	Millisekund	Tal	978
z	Tidzon	Allmän tidzon	Pacific Standard Time; PST; GMT-08:00
Z	Tidzon	RFC 822-tidzon	-0800

Mönsterbokstäver upprepas vanligtvis, eftersom deras antal motsvarar vad som ska visas:

- **Text:** Om antalet mönsterbokstäver vid formatering är fyra eller mer, används den fullständiga formen, annars används en kortform eller en förkortning. När det gäller tolkning är båda formerna godkända, oavsett antal mönsterbokstäver.
- **Tal:** För formatering är antalet mönsterbokstäver det minsta antalet siffror och kortare tal fylls ut med nollor till detta antal. När det gäller tolkning ignoreras antalet mönsterbokstäver såvida det inte behövs för att två närliggande fält ska kunna avgränsas.
- **År:** Om antalet mönsterbokstäver vid formatering är två, förkortas året till två siffror, annars tolkas det som ett tal.

När det gäller tolkning tolkas året ordagrant om antalet mönsterbokstäver är mer än 2, oavsett antalet siffror. Om du använder mönstret "MM/dd/yyyy", "01/11/12" tolkas detta som den 11 januari 12 e.Kr.

För tolkning med ett förkortat mönster för året ("y" eller "yy") måste det förkortade året tolkas i förhållande till ett århundrade, vilket sker genom att datumen justeras att vara inom 80 år före och 20 år efter den tid då förekomsten av datumformatet skapas. Om du exempelvis använder mönstret "MM/dd/yy" och en datumformatsförekomst som skapades den 1 januari 1997, tolkas strängen "01/11/12" som den 11 januari 2012 medan strängen "05/04/64" skulle tolkas som den 4 maj 1964. Vid tolkning, kommer endast strängar som innehåller två siffror att tolkas till standardårhundradet. Andra numeriska strängar, till exempel en sträng med en, tre eller flera siffror eller en sträng med två siffror som inte enbart är siffror (exempelvis "-1") kommer att tolkas ordagrant. Det innebär att "01/02/3" och "01/02/003" tolkas (med samma mönster) som den 2 januari 3 AD. På samma sätt tolkas "01/02/-3" som den 2 januari 4 BC.

- **Månad:** Om antalet mönsterbokstäver är tre eller fler tolkas månaden som text, annars tolkas den som ett tal.
- **Allmän tidzon:** Tidzoner tolkas som text om de har namn. För tidzoner som motsvarar ett GMT-förskjutningsvärde, används följande syntax:
  - GMT-förskjutningstidzon. *GMT tecken timmar:minuter*
  - Tecken + eller -
  - Timmar. *Siffra* eller *siffra siffra*
  - Minuter. *Siffra siffra*
  - Siffror. Något av följande: 0 1 2 3 4 5 6 7 8 9

*Timmar* måste vara mellan 0 och 23 och *minuter* måste vara mellan 00 och 59. Formatet är landsberoende och siffrorna måste tas från Basic Latin-blocket i Unicode-standarden.

För tolkning accepteras även RFC 822-tidzoner.

- **RFC 822-tidzon:** För formatering används RFC 822:s fyrsiffriga tidzonsformat:
  - RFC822-tidzon. *Tecken Tvåsiffrorstimmar minuter*
  - Tvåsiffrorstimmar. *Siffra siffra*

*Tvåsiffrorstimmar* måste vara mellan 00 och 23. Andra definitioner är samma som för allmänna tidzoner.

För tolkning accepteras även allmänna tidzoner.

Följande exempel visar hur datum- och tidsmönster tolkas med lokala inställningar för USA. Datum och klockslag är 2001-07-04 12:08:56 lokal tid, USA Pacific Time.

Datum- och tidsmönster	Resultat
"yyyy.MM.dd G 'at' HH:mm:ss z"	2001.07.04 AD kl. 12:08:56 PDT
"EEE, MMM d, 'yy"	on, jul 4, '01
"h:mm a"	12:08 PM
"hh 'o' 'clock' a, zzzz"	12 o'clock PM, Pacific Daylight Time
"K:mm a, z"	0:08 PM, PDT
"yyyyy.MMMMM.dd GGG hh:mm aaa"	02001.juli.04 AD 12:08 PM
"EEE, d MMM yyyy HH:mm:ss Z"	on, 4 jul 2001 12:08:56 -0700
"yyMMddHHmmssZ"	010704120856-0700

Copyright 2003 Sun Microsystems, Inc. Omtryckt med tillåtelse.

## Kontrollera felstatus i tilläggsfunktioner

Du kan använda följande tilläggsfunktion i en XSLT-formatmall om du vill visa den aktuella felstatusen i den senast anropade tilläggsfunktionen för FileMaker XSLT och hantera fel som uppstår under bearbetningen av sidorna:

```
fmxslt:check_error_status()
```

När funktionen `fmxslt:check_error_status()` anropas returnerar Web Publishing Engine det aktuella felkodsvärdet för den mest aktuella anropade funktionen som typen Number och återställer sedan felstatusen till 0 ("Inget fel"). Information om felkodsvärden finns i "Felkodsnummer för FileMaker-tilläggsfunktioner för XSLT" på sidan 115.

## Använda loggning

Du kan använda standard-XSLT-elementet `<xsl:message>` när du vill skriva loggposter i Web Publishing Engines programloggfil. Se "Använda Web Publishing Engines programlogg" på sidan 86.

## Använda bearbetning på serversidan av manusspråk

Den underliggande XSLT-transformeraren som ingår i Web Publishing Engine stöder bearbetning på serversidan för manusspråk. Som ett resultat kan du använda JavaScript när du vill utveckla dina egna tilläggsfunktioner som kan anropas direkt från en XSLT-formatmall.

Två Java-bibliotek har installerats för att aktivera de här funktionerna:

- `bsf.jar` – Detta bibliotek medger att XSLT-omformaren ansluter till manusspråk.
- `js.jsr` – Detta bibliotek är en fullständig JavaScript-implementation från Mozilla-projektet.

Med dessa bibliotek kan du skapa egna tilläggsfunktioner inuti koden för XSLT-formatmallen. Dessa tilläggsfunktioner kan implementera all manuslogik och kan vara lättare att hantera än om du skulle lita på XSLT och XPath för att utföra logiska funktioner.

Du kan hitta mer detaljerad information om stöd för tillägg i XSLT-omformaren på webbplatsen för Apache Xalan Extensions:

<http://xml.apache.org/xalan-j/extensions.html>

## Definiera en tilläggsfunktion

Så här definierar du en tilläggsfunktion inuti din formatmall:

### 1. Definiera namnuttrycket för tillägget.

Lägg till namnuttrycket `xalan` för att instruera XSLT-omformaren att stödja tilläggskomponenter, erbjuda namn för dina egna namnuttryck för tilläggsfunktioner. Följande exempel använder `fmp-ex` som tilläggsfunktion för namnuttrycksprefixet.

```
<xsl:stylesheet version="1.0"
xmlns:xsl=http://www.w3.org/1999/XSL/Transform
xmlns:xalan=http://xml.apache.org/xslt
xmlns:fmp-ex="ext1"
exclude-result-prefixes="xsl xalan fmp-ex">
```

### 2. Definiera tilläggskomponenten och tilläggsfunktionerna med den kod som faktiskt implementerar din tilläggsfunktion.

```
<xalan:component prefix="fmp-ex" functions="getValueColor">
<xalan:script lang="javascript">
  function getValueColor(value) {
    if (value > 0)
      return ("#009900");
    else
      return ("#CC0000");
  }
</xalan:script>
</xalan:component>
```

Det här exemplet returnerar ett färgvärde baserat på ett indatavärde. Om indatavärdet är större än 0, returneras färgen grön ("#009900"). I annat fall, om värdet är lägre än 0, returneras färgen röd ("#CC0000").

**Obs!** Elementet `<xalan:component>` måste vara underordnat elementet `<xsl:stylesheet>`.

### 3. Använd tilläggsfunktionen inuti formatmallen.

Följande exempel visar hur du anropar en tilläggsfunktion med hjälp av ett XPath-uttryck.

Det här första exemplet skulle ställa in färgen på teckensnittet till grönt ("#009900").

```
<font color="{fmp-ex:getValueColor(50)}">Värdet är 50</font>
```

Det här andra exemplet skulle ställa in färgen på teckensnittet till rött ("#CC0000").

```
<font color="{fmp-ex:getValueColor(-500)}">Värdet är -500</font>
```

## Exempel på en tilläggsfunktion

Den enkla JavaScript-funktionen som användes i ovanstående process kunde ha implementerats med hjälp av ett `<xsl:choose>`-uttryck. Den verkliga styrkan i att använda ett manustillägg är att du kan skapa en funktion som inte kan implementeras bara i XSLT eller XPath.

Låt oss t.ex. säga att du skapar en intranätportal för ditt företag och du vill inkludera information om aktuellt aktiepris på den portalsidan. Det finns tillgängliga XML-feeds för aktier tillgängliga men normalt kräver de kommersiella licenser för att du ska kunna använda dem. Du kan emellertid hämta aktiedata i formatet CSV-dokument (Comma Separated Values) från Yahoo!:'s webbplats. XPath-funktionen `document()` kan importera innehåll från XML-källor men du behöver konvertera CSV-innehållet till XML. En lösning är att använda JavaScript och hämta information om CSV-aktiepriserna, tolka filen och extrahera data.

Denna URL visar syntaxen för att hämta ett aktiepris från Yahoo!:'s webbplats som en CSV-fil:

```
http://quote.yahoo.com/d/quotes.csv?s=<ticker>&f=l1gh&e=.csv
```

där <ticker> representerar aktiesymbolen för den aktie du vill hämta data för.

De data som returneras är tre tal separerade med kommatecken, t.ex.:

```
31.79,31.17,32.12
```

där det första värdet är sista försäljningspris, det andra värdet är dagens lägsta och det tredje värdet är dagens högsta.

Nedanstående exempel visar en XSLT-tilläggsfunktion i JavaScript vilken hämtar ett aktuellt aktiepris från Yahoo!:'s webbplats och gör det tillgängligt via en XPath-funktion:

```
<?xml-version="1.0" encoding="UTF-8"?>
<xsl:stylesheet
  exclude-result-prefixes="xsl fmxslt fmrs xalan fmp-ex"
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
  xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"
  xmlns:xalan="http://xml.apache.org/xslt"
  xmlns:fmp-ex="ext1"
>

<?xslt-cwp-query params="-grammar=fmresultset&-process" ?>
<xsl:output method="html"/>
<xalan:component prefix="fmp-ex" functions="include get_quote" >
<xalan:script lang="javascript">
  function include(url) {
    var dest = new java.net.URL(url);
    var dis = new java.io.DataInputStream(dest.openStream());
    var res = "";
    while ((line = dis.readLine()) != null)
    {
      res += line + java.lang.System.getProperty("line.separator");
    }
    dis.close();
    return res;
  }
  function get_quote(ticker) {
    url = "http://quote.yahoo.com/d/quotes.csv?s="+
```

```

        "+ticker+"&f=l1gh&e=.csv";
    csv_file = include(url);
    var str_tokenizer = new java.util.StringTokenizer(csv_file, ',');
    // den första token i det senaste försäljningspriset
    var last = str_tokenizer.nextToken();
    return last;
}
</xalan:script>
</xalan:component>

<xsl:template match="/fmrs:fmresultset">
    <html>
        <huvuddel>
            <font size="2" face="verdana, arial">
                Aktiepris för Apple: <xsl:value-of select="fmp-ex:get_quote('AAPL')"/>
            </font>
        </body>
    </html>
</xsl:template>

</xsl:stylesheet>

```

När Web Publishing Engine bearbetar den här formatmallen kräver den aktieinformation från Yahoo!:'s webbplats. Funktionen `get_quote()` tolkar aktiedata och returnerar data till formatmallen. Dessa omformade utdata visas i webbläsaren.



# Kapitel 7

## *Iscensätta, testa och övervaka en webbplats*

I detta kapitel finns instruktioner för att iscensätta och testa en webbplats med anpassad webbpublicering innan den körs i en produktionsmiljö. Instruktioner finns även för att använda loggfiler för att övervaka sidan vid testning och efter den har körts igång.

### Iscensätta en anpassad webbpubliceringsplats

Innan du kan testa din webbplats ordentligt måste du kopiera och flytta de filer som behövs till korrekta platser på servern eller servrarna där du ska iscensätta webbplatsen.

Så här iscensätter du och förbereder du webbplatsen för testning:

1. Genomför alla steg som anges i kapitel 3, ”Förbereda databaser för anpassad webbpublicering”.
2. Kontrollera att XSLT och XML har aktiverats och konfigurerats på rätt sätt i FileMaker Server Admin Console.

**Obs!** Anvisningar finns i hjälpen för FileMaker Server.

3. Kontrollera att webbservern och Web Publishing Engine är i gång.

4. Kopiera eller flytta XSLT-formatmallarna till den dator där Web Publishing Engine finns.

Kopiera eller flytta XSLT-formatmallarna till följande katalog på den dator där Web Publishing Engine finns.

- Apache (Mac OS): /Library/FileMaker Server/Web Publishing/xslt-template-files
  - IIS (Windows): <enhet>\Program Files\FileMaker\FileMaker Server\Web Publishing\xslt-template-files
- där <enhet> är den primära enhet där programmet startas.

**Obs!** Du kan också placera formatmallarna i en valfri mapp eller mapphierarki i mappen xslt-template-files.

5. Kopiera eller flytta alla hänvisade containerobjekt till webbserverdatorn.

Om ett containerfält i en databas lagrar en filreferens istället för en verklig fil måste det containerobjektet som referens leder till lagras i FileMaker Pro-katalogen när posten skapas eller redigeras. Om du vill iscensätta din webbplats måste du kopiera eller flytta de hänvisade containrarna till en mapp med samma relativa position i rotkatalogen på webbserverprogramvaran.

**Obs!** Om containerfälten innehåller de faktiska filerna i FileMaker-databasen behöver du inte göra något med innehållet i containerfältet, under förutsättning att FileMaker Server fungerar som värd för databasfilen och att filen är tillgänglig på FileMaker Server.

6. Använd följande URL-syntax för att begära och bearbeta en XSLT-formatmall och generera den resulterande HTML-koden:

```
<scheme>://<host>[:<port>]/fmi/xsl/<path>/<stylesheet>.xsl[?<query string>]
```

där:

- <scheme> är http- eller HTTPS-protokollet.
- <host> är IP-adressen eller domännamnet för värddatorn som webbservern är installerad på.
- <port> är valfritt och anger porten som webbservern övervakar. Om ingen port anges används standardporten för respektive protokoll (port 80 för HTTP och port 443 för HTTPS).
- <path> är valfritt och anger sökvägen till undermappen i mappen xslt-template-files som innehåller XSLT-formatmallen.
- <stylesheet> är namnet på formatmallen med filnamnstillägget .xsl.
- <query string> kan vara en kombination av ett frågekommando och en eller flera frågeparametrar för anpassad webbpublicering med XSLT.

Om den angivna formatmallen innehåller bearbetningsinstruktionen <?xslt-cwp-query ?> får det statiskt definierade frågekommandot och parametrarna företräde framför eventuella matchande frågekommandon eller parametrar i URL-frågesträngen. Om du refererar till formatmallen home.xsl, som genereras av XSLT Site Assistant, behöver du inte lägga till någon frågesträng. Mer information om frågesträngar finns i bilaga A, "Giltiga namn i frågesträngar".

URL-syntaxen, inklusive namnen på frågekommandon och parametrar, är skiftlägeskänslig, utom för delar av frågesträngen. I större delen av URL-adressen används gemener. Om du t ex kopierade formatmallarna (inklusive formatmallen home.xsl) till mappen my\_templates, som är en undermapp till mappen xslt-template-files, kan du använda följande URL för att begära och bearbeta formatmallarna:

```
http://192.168.123.101/fmi/xsl/my_templates/home.xsl
```

**Obs!** Web Publishing Engine tillåter inte att webbanvändarna visar källan för XSLT-formatmallar som är installerade i mappen xslt-template-files. När en webbanvändare skickar en begäran om bearbetning av en formatmall skickar Web Publishing Engine bara HTML-sidorna som är resultatet av XSLT Site Assistants formatmallar till webbläsaren.

## Testa en anpassad webbpubliceringsplats

Innan du meddelar användarna att din anpassade webbpubliceringsplats är tillgänglig ska du kontrollera att den ser ut och fungerar som du har tänkt dig.

- Testa funktioner som sökning och att lägga till, radera och sortera poster med olika konton och behörighet.
- Kontrollera att behörighetsinställningarna fungerar som väntat genom att logga in med olika konton. Kontrollera att inga obehöriga användare kan komma åt och ändra dina data.
- Kontrollera alla manus så att du vet att resultatet blir det väntade. Information om hur du skapar manus som fungerar väl på webben finns i "FileMaker-manus och anpassad webbpublicering" på sidan 22.
- Testa webbplatsen med olika operativsystem och webbläsare.

**Obs!** Om du inte har någon nätverksanslutning och har installerat FileMaker Server på en enda dator – med webbserver, Web Publishing Engine och databasserver på en dator – kan du testa din anpassade webbpubliceringsplats genom att använda `http://localhost/` eller `http://127.0.0.1/` i URL-adressen. Information om URL-syntaxen finns i ”URL-syntax för XML-data och containerobjekt” på sidan 35, och ”URL-syntax för XSLT-formatmallar för FileMaker” på sidan 54.

## Exempel på formatmallar som du kan använda när du testar XML-utdata

Här är två exempel på XSLT-formatmallar som du kan använda när du vill testa XML-utdata.

- Följande formatmall skickar efterfrågade XML-data utan att göra någon konvertering. Denna formatmall är bra när du vill visa den faktiska XML-information Web Publishing Engine använder.

```
<?xml-version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
  <xsl:output method="xml"/>
  <xsl:template match="/">
    <xsl:copy-of select="."/>
  </xsl:template>
</xsl:stylesheet>
```

- När du avslutar en formatmall kan du använda följande exempel på en HTML `<textarea>`-markör när du vill visa det XML-källdokument som hämtades via formatmallen i ett textbläddringsfält. På samma sida kan du jämföra det konverterade XSLT-resultatet med XML-källdokumentet före konverteringen.

```
<?xml-version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
  <xsl:output method="html"/>
<html>
  <huvuddel>
  <xsl:template match="/fmrs:fmresultset">
    <textarea rows="20" cols="100">
      <xsl:copy-of select="."/>
    </textarea><br/>
  </xsl:template>
  </body>
</html>
</xsl:stylesheet>
```

## Övervaka webbplatsen

Du kan använda följande typer av loggfiler när du vill övervaka den anpassade webbpubliceringsplatsen och hämta information om de webbanvändare som besöker webbplatsen:

- Webbserveråtkomst och felloggar
- Programlogg för Web Publishing Engine
- Fellogg för webbservermodulen
- Webbpubliceringskärnans interna åtkomstloggar

### Använda webbserveråtkomst och felloggar

Apache (Mac OS): Webbservern Apache genererar en åtkomstloggfil och en felloggfil. Åtkomstloggfilen på Apache, som är i formatet W3C Common Logfile Format som standard, är en post över alla inkommande HTTP-förfrågningar till webbservern. Felloggen på Apache är en lista med problem som rör bearbetningen av HTTP-förfrågningar. Mer information om dessa loggfiler finns i dokumentationen för Apache-webbservern.

IIS (Windows): Webbservern Microsoft IIS genererar en åtkomstloggfil och visar felen i Loggboken (Windows) i stället för att skriva dem till en loggfil. Åtkomstloggfilen, som är i formatet W3C Extended Log File Format som standard, är en lista med alla inkommande HTTP-förfrågningar till webbservern. Du kan även använda formatet W3C Common Logfile Format för åtkomstloggen. Mer information finns i dokumentationen till Microsoft IIS-webbservern.

Information om formaten W3C Common Logfile Format och W3C Extended Log File Format finns på webbplatsen [www.w3.org](http://www.w3.org).

### Använda Web Publishing Engines programlogg

Som standard genererar Web Publishing Engine en programlogg som är en post över fel, manus och användarens loggningsinformation.

- Fellogginformationen beskriver eventuella ovanliga fel som har uppstått i Web Publishing Engine. Vanliga felmeddelanden som rapporteras till webbanvändaren, t.ex. om att en viss databas inte är öppen, registreras inte i felloggfilen.
- Manuslogginformationen beskriver eventuella fel som har uppstått när webbanvändarna utförde manus. Den kan exempelvis innehålla en lista över manussteg som ignoreras om de inte är webbkompatibla.
- Användarens loggmeddelanden innehåller meddelanden som genereras av XSLT-elementet `<xsl:message>` i XSLT-formatmallar. Varje gång en webbanvändare använder XSLT-formatmallen kommer den information du tar med i ett `<xsl:message>`-element att registreras i programloggfilen. Se kapitel 6, "Utveckla XSLT-formatmallar för FileMaker".

Programloggfilen heter `pe_application_log.txt` och finns i mappen Logs i mappen FileMaker Server på värden för Web Publishing Engine.

Filen `pe_application_log.txt` skapas om något av följande loggningsalternativ är aktiverat i Web Publishing Engine:

Loggningsalternativ aktiverat	Information registrerad i <code>pe_application_log.txt</code>
Felloggning	Alla ovanliga fel som har inträffat på Web Publishing Engine. Vanliga felmeddelanden som rapporteras till webbanvändaren, t.ex. om att en viss databas inte är öppen, registreras inte i felloggfilen.
Manusloggning	Alla fel som genereras när webbanvändarna verkställer manus. Den kan exempelvis innehålla en lista över manussteg som ignoreras om de inte är webbkompatibla.
Användarloggning	Meddelanden som skapas när webbanvändare använder din lösning för anpassad webbpublicering.

Som standard är alla tre alternativ aktiverade. Information om hur du ställer in dessa alternativ med hjälp av Admin Console finns i hjälpen till FileMaker Server .

**Obs!** Posterna i programloggen tas inte bort automatiskt och med tiden kan filen därför bli mycket stor. Du kan spara utrymme på hårddisken genom att arkivera programloggfilen med jämna mellanrum.

## Använda felloggen för webbservern

Om webbservern inte kan ansluta till Web Publishing Engine skapar webbservermodulen en loggfil över eventuella fel som uppstår vid bearbetningen. Filen heter `web_server_module_log.txt` och finns i mappen Logs i mappen för FileMaker Server på datorn som är webbservervärd.

## Använda webbpubliceringskärnans interna åtkomstloggar

Programkomponenten för webbpubliceringskärnan som tillhör Web Publishing Engine skapar två interna åtkomstloggfiler som en registrering varje gång webbpubliceringskärnan används.

- Åtkomstloggen `wpc_access_log.txt` innehåller alla förfrågningar från slutanvändare om att skapa XML-kod och att använda direkt webbpublicering i FileMaker Server direkt webbpublicering. Dessa förfrågningar dirigeras från webbservern direkt till webbpubliceringskärnan.
- Åtkomstloggen `pe_internal_access_log.txt` visar alla interna XML-förfrågningar som XSLT-CWP-programkomponenten i Web Publishing Engine skapar vid bearbetning av XSLT-förfrågningar. Dessa förfrågningar skickas internt i Web Publishing Engine från XSLT-CWP-programkomponenten till webbpubliceringskärnan.

Dessa loggfiler finns i mappen Logs i mappen för FileMaker Server på värden för Web Publishing Engine.

De interna åtkomstloggarna skapas om alternativet Åtkomstloggning är aktiverat i Web Publishing Engine. Alternativet är enligt standardinställningen aktiverat. Information om hur du ställer in alternativet Åtkomstloggning med hjälp av Admin Console finns i hjälpen till FileMaker Server.



# Bilaga A

## Giltiga namn i frågesträngar

Den här bilagan innehåller giltiga namn på frågekommandon och frågeparametrar som du kan använda i en frågesträng för att hämta FileMaker-data med hjälp av Web Publishing Engine.

### Frågekommandon och frågeparametrar

Listan nedan innehåller namnen på alla frågekommandon och frågeparametrar:

#### Frågekommandon

- dbnames (Se sidan 93.)
- delete (Se sidan 93.)
- dup (Se sidan 94.)
- edit (Se sidan 94.)
- find, –findall, –findany (Se sidan 94.)
- findquery (Se sidan 95.)
- layoutnames (Se sidan 95.)
- new (Se sidan 95.)
- process (endast XSLT) (Se sidan 96.)
- scriptnames (Se sidan 96.)
- view (Se sidan 96.)

#### Frågeparametrar

- db (Se sidan 96.)
- encoding (endast XSLT) (Se sidan 97.)
- field (Se sidan 97.)
- fältnamn (Se sidan 97.)
- fältnamn.op (Se sidan 98.)
- grammar (endast XSLT) (Se sidan 99.)
- lay (Se sidan 99.)
- lay.response (Se sidan 99.)
- lop (Se sidan 100.)
- max (Se sidan 100.)
- modid (Se sidan 100.)
- query (Se sidan 100.)
- recid (Se sidan 101.)
- relatedsets.filter (Se sidan 102.)
- relatedsets.max (Se sidan 102.)
- script (Se sidan 102.)
- script.param (Se sidan 103.)
- script.prefind (Se sidan 103.)
- script.prefind.param (Se sidan 103.)
- script.presort (Se sidan 104.)
- script.presort.param (Se sidan 104.)
- skip (Se sidan 104.)
- sortfield.[1-9] (Se sidan 105.)
- sortorder.[1-9] (Se sidan 105.)
- stylehref (Se sidan 106.)
- styletype (Se sidan 106.)
- token.[string] (endast XSLT) (Se sidan 106.)

**Viktigt!** Parametern –lay för att ange en layout krävs med alla query-kommandon utom –dbnames, –layoutnames, –scriptnames och –process (endast XSLT-frågor).

## Riktlinjer för frågekommandon och frågeparametrar

Tänk på följande när du använder frågekommandon och frågeparametrar i en frågesträng:

- En frågesträng måste innehålla ett frågekommando, varken fler eller färre. En frågesträng kan till exempel innehålla `-new` om du vill lägga till en ny post men den kan inte innehålla `-new` och `-edit` i samma frågesträng.
- För de flesta frågekommandon krävs diverse frågeparametrar i frågesträngen. För alla frågekommandon förutom `-dbnames` och `-process` krävs parametern `-db`, som anger vilken databas frågan ska ställas till. En tabell över vilka parametrar som krävs finns i "Efterfråga XML-data med hjälp av frågesträngar i FileMaker" på sidan 47.
- För frågeparametrar och fältnamn anger du det specifika värde du vill använda, till exempel `-db=employees`. För frågekommandon ska du inte ange tecknet "=" eller ett värde efter kommandonamnet, till exempel `-findall`.
- Frågekommandon och frågeparametrar måste skrivas med små bokstäver (gemener), till exempel `-delete` och `-lay`.
- Databasnamn, layoutnamn och fältnamn som används i frågesträngar är inte skiftlägeskänsliga. Du kan till exempel skriva `-lay=mylayout` när du vill ange `MyLayout`.

**Obs!** Fältnamn och databasnamn som används i XSLT-uttryck utanför frågesträngar är skiftlägeskänsliga och måste skrivas exakt likadant som de riktiga namnen som används i databasen. I följande uttryck:

```
<xsl:value-of select="fmrs:field[@name='LastName']"/>
```

måste exempelvis fältreferensen "LastName" skrivas exakt likadant som fältet LastName i databasen.

- Fältnamn kan innehålla punkter, med följande undantag:
  - Punkten kan inte följas av en siffra. Exemplet `myfield.9` är ett ogiltigt fältnamn.
  - Punkten kan inte följas av textsträngen `op` (de två bokstäverna "op"). Exemplet `myfield.op` är ett ogiltigt fältnamn.
  - Punkten kan inte följas av textsträngen `global` (ordet "global"). Exemplet `myfield.global` är ett ogiltigt fältnamn.

Fältnamn som innehåller något av dessa undantag kan inte nås via XML eller XSLT med hjälp av en http-fråga. Dessa ord är reserverade för post-ID enligt beskrivningen i avsnittet, "Syntax för ett fullständigt fältnamn," nedan.

- För kommandot `-find` är värdet i ett fält inte skiftlägeskänsligt. Du kan exempelvis använda `Field1=Blue` eller `Field1=blue`. För kommandona `-new` och `-edit` bevaras de stora och små bokstäver du använder i värdet för ett fält i databasen, på samma sätt som du skrev i frågesträngen. Exempel, `LastName=Doe`.

## FileMaker-databasen Query Strings Reference

FileMaker Server-skivan innehåller en FileMaker-databas som heter Query Strings Reference.fp7. Den innehåller korta beskrivningar och exempel på alla FileMaker-frågekommandon och -frågeparametrar. Du hittar den i följande katalog på alla datorer där FileMaker Server är installerat (huvuddator eller arbetsdator):

Mac:

```
/Library/FileMaker Server/Examples/XSLT
```

Windows:

```
<drive>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT
```

Där: <enhet> är den primära enhet där programmet startas.

## Syntax för ett fullständigt fältnamn

Ett fullständigt fältnamn identifierar en exakt förekomst av ett fält. Eftersom fält med vanliga namn kan baseras på olika tabeller, måste du använda fullständiga namn för att undvika fel.

Syntax för ett fullständigt fältnamn är:

```
tabellnamn::fältnamn(repetitionsnummer).post-id
```

där:

- tabellnamn är namnet på tabellen som innehåller fältet. Tabellnamnet krävs bara om fältet inte finns i den underliggande tabellen i den layout som anges i frågesträngen.
- fältnamn(repetitionsnummer) är värdet i ett repeterat fält och krävs endast för repeterade fält. Repetitionsnumret börjar med 1. Exempelvis avser field-name(2) det andra värdet i det repeterade fältet. Om du inte anger ett repetitionsnummer för ett repeterat fält används det första värdet i det repeterade fältet. Repetitionsnumret krävs för frågekommandona `-new` och `-edit` när de hanterar repeterade fält, men inte för kommandot `-find`.
- post-id är postens ID och krävs endast om du vill lägga till eller redigera poster i portalfält med hjälp av en frågesträng. Se avsnitten "Lägga till poster i en portal," och "Redigera poster i en portal" post-id krävs för frågekommandona `-new` och `-edit` när de hanterar portalfält, men krävs inte för kommandot `-find`.

**Obs!** För att fält ska kunna användas måste de vara placerade i den layout du har angett i frågesträngen.

## Använda frågekommandon med portalfält

I följande avsnitt finns beskrivningar om hur frågekommandon fungerar med portalfält.

### Lägga till poster i en portal

Om du vill lägga till en ny post i en portal samtidigt som du lägger till en överordnad post, använder du frågekommandot `-new` och gör följande i frågesträngen för förfrågan:

- Använd det fullständiga fältnamnet för det relaterade portalfältet.
- Ange 0 som post-ID efter namnet på det relaterade portalfältet.
- Ange minst ett av fälten för den överordnade posten innan du anger det relaterade portalfältet.
- Ange data för det matchande fältet (nyckelfältet) i den överordnade posten.

Följande URL-adresser lägger exempelvis till en ny överordnad anställningspost för John Doe och en ny relaterad post för Jane i portalen på samma gång. Namnet på den relaterade tabellen är Dependents och namnet på det relaterade fältet i portalen är Names. I det matchande fältet, ID, sparas ett anställnings-ID.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&FirstName=John&LastName=Doe
&ID=9756&Dependents::Names.0=Jane&-new
```

**Obs!** Du kan enbart lägga till en relaterad post till en portal per anrop.

## Redigera poster i en portal

Om du vill redigera en eller flera poster i en portal använder du kommandot `-edit` och ett post-ID för att ange den överordnade post som innehåller de portalposter du vill redigera. Ange den portalpost du vill redigera genom att ange dess post-ID i ett fullständigt fältnamn. Du kan bestämma ett post-ID från postens ID-attribut i elementet `<record>` i elementet `<relatedset>` i XML-informationen. Se ”Använda syntaxen `fmresultset`” på sidan 39.

Följande URL-adress redigerar exempelvis en post i en portal där den överordnade posten har post-ID 1001. `Dependents` är namnet på den relaterade tabellen, `Names` är namnet på det relaterade fältet i portalen och `2` i `Names.2` är post-ID för en portalpost.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=1001
&Dependents::Names.2=Kevin&-edit
```

Här är ett exempel på hur du kan använda en förfrågan för att redigera flera portalposter via den överordnade posten:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=1001
&Dependents::Names.2=Kevin&Dependents::Names.5=Susan&-edit
```

Du kan också använda kommandot `-edit` och ange `0` som portalpostens ID när du vill lägga till en ny relaterad post i portalen för en befintlig överordnad post. Till exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=1001
&Dependents::Names.0=Timothy&-edit
```

## Radera portalposter

Om du vill radera portalposter använder du parametern `-delete.related` tillsammans med kommandot `-edit` i stället för att använda kommandot `-delete`.

Följande URL-adress tar t.ex. bort posten 1001 från tabellen `employees`:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=1001&-delete
```

Följande URL-adress tar bort en portalpost med post-ID 3 från den relaterade tabellen som kallas `Dependents`, med det överordnade ID-numret 1001.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=1001
&-delete.related=Dependents.3&-redigera
```

Mer information finns i ”`-delete.related` (radera portalposter), frågeparameter” på sidan 97.

## fråga portalfält

I en lösning som har många relaterade poster kan sökning och sortering av portalposter vara tidskrävande. För att begränsa antalet poster och rader som visas i ett sökresultat kan du använda parametrarna `-relatedsets.filter` och `-relatedsets.max` med sökposter. Mer information finns i ”`-relatedsets.filter` (filtrera portalposter), frågeparameter” på sidan 102 och ”`-relatedsets.max` (begränsa portalposter), frågeparameter” på sidan 102.

## Syntax för globala fält

Syntaxen för globala fält är:

tabellnamn::fältnamn(repetitionsnummer).global

där global anger att ett fält använder global lagring. Information om tabellnamn och fältnamn(repetitionsnummer) finns i ”Syntax för ett fullständigt fältnamn” på sidan 91. Information om fält som använder global lagring finns i FileMaker Pro Hjälp.

Du måste använda syntaxen .global om du vill identifiera ett globalt fält i en frågesträng. Web Publishing Engine anger parametervärden för globala fält innan frågekommandot utförs eller några andra parametervärden ställs in i frågesträngen. För direkta XML-förfrågningar och förfrågningar som görs via XSLT-formatmallar som inte använder sessioner, slutar de globala värdena att gälla så snart förfrågan är klar. För förfrågningar via en XSLT-formatmall som använder sessioner, finns de globala värdena kvar under den sessions längd som anges i formatmallen, eller tills värdena ändras med en annan förfrågan.

Om du inte identifierar ett globalt fält i en frågesträng med hjälp av syntaxen .global, utvärderar Web Publishing Engine det globala fältet tillsammans med återstoden av frågesträngen, utan att först ange det globala fältvärdet.

Till exempel:

[http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments  
&Country.global=USA&-recid=1&-edit](http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&Country.global=USA&-recid=1&-edit)

**Viktigt!** Om du använder ett globalt fält i en XSLT-formatmall måste du använda Admin Console för att kunna aktivera alternativet XSLT-databassessioner för Web Publishing Engine. I annat fall bevaras inte värdena i de globala fälten mellan förfrågningar. Se FileMaker Server Hjälp.

## Referens för frågekommando

Det här avsnittet innehåller information om de frågekommandon som är tillgängliga för XML- och XSLT-förfrågningar.

**Obs!** Endast för XSLT-förfrågningar: alla frågekommandon som följer kräver frågeparametern –grammar.

### Frågekommandot –dbnames (databasnamn)

Hämtar namnen på alla databaser som FileMaker Server är värd för och som är aktiverade för anpassad webbpublicering med XML eller XSLT.

**Obligatoriska frågeparametrar:** (inga)

**Exempel:**

Om du vill hämta databasnamnen:

<http://192.168.123.101/fmi/xml/fmresultset.xml?-dbnames>

### Frågekommandot –delete (radera post)

Tar bort den post som anges med parametern –recid.

**Obligatoriska frågeparametrar:** –db, –lay, –recid

**Valfri frågeparameter:** –script

**Exempel:**

Så här raderar du en post:

<http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=4&-delete>

### Frågekommandot **-dup (duplicera post)**

Duplicerar den post som anges av `-recid`

**Obligatoriska frågeparametrar:** `-db`, `-lay`, `-recid`

**Valfri frågeparameter:** `-script`

#### Exempel:

Om du vill duplicera den angivna posten:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=14&-dup`

### Frågekommandot **-edit (redigera post)**

Uppdaterar posten som anges med parametern `-recid` och fyller fälten med innehållet i ett fältnamn/värdepar. Parametern `-recid` anger vilken post som ska ändras.

**Obligatoriska frågeparametrar:** `-db`, `-lay`, `-recid`, ett eller flera fältnamn

**Valfri frågeparameter:** `-modid`, `-script`

**Obs!** Information om hur du redigerar poster i en portal finns i ”Redigera poster i en portal” på sidan 92.

#### Exempel:

Så här ändrar du en post:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=13&Country=USA&-edit`

### Frågekommandona **-find**, **-findall** och **-findany (sök efter poster)**

Skickar en sökförfrågan med definierade villkor.

**Obligatoriska frågeparametrar:** `-db`, `-lay`

**Valfria frågeparametrar:** `-recid`, `-lop`, `-op`, `-max`, `-skip`, `-sortorder`, `-sortfield`, `-script`, `-script.prefind`, `-script.presort`, fältnamn

#### Exempel:

Om du vill söka efter en post med hjälp av ett fältnamn:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&Country=USA&-find`

**Obs!** Att ange att fältnamn flera gånger i en enda fråga stöds inte. FileMaker Server tolkar alla värdena men använder endast det senast tolkade.

Om du vill söka efter en post med hjälp av post-ID:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-recid=427&-find`

Om du vill söka efter alla poster i databasen använder du `-findall`:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-findall`

Om du vill söka efter en post slumpmässigt använder du `-findany`:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-findany`

## –findquery (sammanfattad sökning), frågekommando

Skickar en sökfråga med hjälp av flera sökposter och fråga för att utelämna poster.

**Obligatoriska frågeparametrar:** –db, –lay, –query

**Valfria frågeparametrar:** –max, –skip, –sortorder, –sortfield, –script, –script.prefind, –script.presort

### Exempel:

Sök poster med katter eller hundar som inte heter ”Fluffy.”

```
http://host/fmi/xml/fmresultset.xml?–db=vetclinic&–lay=animals&–query=(q1);(q2);!(q3)
&–q1=typeofanimal&–q1.value=Cat&–q2=typeofanimal&–q2.value=Dog&–q3=name&–q3.value=Fluffy&–findquery
```

### Använd kommandot –findquery för sammansatta sökningar

Ett –findquery-uttryck består av fyra delar, i följande ordning:

- Parametern –query.
- Frågedeklarationerna som består av deklARATIONEN med frågeidentifieraren och frågebearbetningar.
- Sökfältet och värdedefinitioner för varje frågeidentifierare.
- Kommandot –findquery i slutet av det fullständiga uttrycket.

För mer information om att använda parametern –query, se ”Frågeparametern –query (sammanfattad sökning)” på sidan 100.

## Frågekommandot –layoutnames (layoutnamn)

Hämtar namnen på alla tillgängliga layouter för en angiven databas som FileMaker Server är värd för och som är aktiverad för anpassad webbpublicering med XML eller XSLT.

**Obligatoriska frågeparametrar:** –db

### Exempel:

Om du vill hämta namnen på tillgängliga layouter:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–layoutnames
```

## Frågekommandot –new (ny post)

Skapar en ny post och fyller den med innehållet i ett fältnamn/värdepar.

**Obligatoriska frågeparametrar:** –db, –lay

**Valfri frågeparameter:** ett eller flera fältnamn, –script

**Obs!** Information om hur du inkluderar nya data för en portal finns i ”Lägga till poster i en portal” på sidan 91.

### Exempel:

Så här lägger du till en ny post:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&Country=Australia&–new
```

### **–process (bearbeta XSLT-formatmallar)**

Bearbetar en XSLT-formatmall utan att begära data från databasen. Detta frågekommando kan endast användas med XSLT-formatmallar.

**Obligatorisk frågeparameter:** –syntax. Du måste använda syntaxen fmresultset eller FMPXMLRESULT.

**Exempel:**

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–process`

Se ”Bearbeta XSLT-förfrågningar som inte skickar frågor till FileMaker Server” på sidan 60.

### **Frågekommandot –scriptnames (manusnamn)**

Hämtar namnen på alla tillgängliga manus för en angiven databas som FileMaker Server är värd för och som är aktiverad för anpassad webbpublicering med XML eller XSLT.

**Obligatoriska frågeparametrar:** –db

**Exempel:**

Om du vill hämta namnen på alla manus:

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–scriptnames`

### **Frågekommandot –view (visa layoutinformation)**

Om syntaxen FMPXMLLAYOUT har angivits hämtas layoutinformation från en databas. Den visas sedan i syntaxen FMPXMLLAYOUT. Om du anger datasyntaxen (fmresultset eller FMPXMLRESULT) hämtas metadataavsnittet i XML-dokumentet samt en tom resultatuppsättning.

**Obligatoriska frågeparametrar:** –db, –lay

**Valfri frågeparameter:** –script

**Exempel:**

Om du vill hämta layoutinformation:

`http://192.168.123.101/fmi/xml/FMPXMLLAYOUT.xml?–db=employees&–lay=departments&–view`

Om du vill hämta metadatainformation:

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–view`

## **Referens för frågeparameter**

Det här avsnittet innehåller information om de frågeparametrar som är tillgängliga för XML- och XSLT-förfrågningar. Information om parametrar som endast är tillgängliga för XSLT-förfrågningar finns i ”Använda frågesträngar i XSLT-formatmallar för FileMaker” på sidan 56.

### **Frågeparametern –db (databasnamn)**

Anger i vilken databas frågekommandot ska användas.

**Värdet är:** namnet på databasen, förutom eventuell filändelse.

**Obs!** När du anger namnet på databasen för parametern –db i en frågesträng, ska du inte ta med filnamnsändelsen. Det riktiga filnamnet på databasen kan innehålla en filändelse, men filändelser är inte tillåtna som värden för parametern –db.

**Krävs för:** Alla frågekommandon förutom –dbnames och –process

**Exempel:**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–findall`

## –delete.related (radera portalposter), frågeparameter

Raderar en post från ett portalfält.

**Kan användas med:** frågekommandot –edit

**Kräver:** Ett relaterat tabellnamn och ett post-ID

### Exempel:

Följande exempel tar bort en portalpost med post-ID 20 från den relaterade tabellen som kallas jobtable med det överordnade ID-numret 7.

`http://host/fmi/xml/fmresultset.xml?–db=career&–lay=applications&–recid=7&–delete.related=jobtable.20&–edit`

## Frågeparametern –encoding (kodning av XSLT-förfrågan)

Anger textkodning för en XSLT-förfrågan. Detta frågekommando kan endast användas tillsammans med anpassad webbpublicering med XSLT-förfrågningar.

**Värdet är:** US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift\_JIS eller UTF-8

**Kan användas med:** alla frågekommandon i en XSLT-förfrågan.

### Exempel:

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–db=employees&–lay=departments&–grammar=fmresultset&–encoding=Shift_JIS&–findall`

Se ”Ange textkodning för förfrågningar” på sidan 59.

## Frågeparametern –field (namn på containerfält)

Anger namnet på ett containerfält.

**Krävs för:** förfrågningar om data i ett containerfält.

Se ”URL-syntax för FileMaker-containerobjekt i XML-lösningar” på sidan 37, och ”URL-syntax för FileMaker-containerobjekt i XSLT-lösningar” på sidan 55.

## Frågeparametern fältnamn (namn på annat fält än containerfält)

Fältnamn styr villkoren för frågekommandot –find. Du kan även använda fältnamn när du vill ändra innehållet i en post. Om du måste ange ett värde för ett annat fält än ett containerfält för ett frågekommando eller en frågeparameter, skriver du fältnamnet utan bindestreck (–) som namndelen i ett namn/värdepar.

**Namnet är:** namnet på fältet i FileMaker-databasen. Om fältet inte finns i den underliggande tabellen i den layout som anges i frågesträngen, måste du ange det fullständiga fältnamnet. Fältnamn kan innehålla punkter, med följande undantag:

- Punkten kan inte följas av en siffra. Exemplet myfield.9 är ett ogiltigt fältnamn.
- Punkten kan inte följas av textsträngen op (de två bokstäverna ”op”). Exemplet myfield.op är ett ogiltigt fältnamn.
- Punkten kan inte följas av textsträngen global (ordet ”global”). Exemplet myfield.global är ett ogiltigt fältnamn.

Fältnamn som innehåller något av dessa undantag kan inte nås via XML eller XSLT med hjälp av en HTTP-fråga. Dessa ord är reserverade för post-ID enligt beskrivningen i avsnittet, ”Syntax för ett fullständigt fältnamn” på sidan 91.

**Värdet är:** För frågekommandona `-new` och `-edit` anger du det värde du vill spara i fältet i den aktuella posten. För `-find`-frågekommandon anger du det värde du vill söka efter i fältet. När du anger värdet för ett datumfält, ett tidsfält eller ett tidsstämpelfält måste du ange värdet enligt formatet ”fm” för den fälttypen. ”fm”-formaten är MM/dd/yyyy för datum, HH:mm:ss för tid och MM/dd/yyyy HH:mm:ss för tidsstämpel.

**Krävs för:** frågekommandot `-edit`

**Kan användas med:** Frågekommandona `-new` och `-find`.

**Exempel:**

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-op=eq&FirstName=Sam  
&-max=1&-find`

**Obs!** Att ange att fältnamn flera gånger i en enda fråga stöds inte. FileMaker Server tolkar alla värdena men använder endast det senast tolkade.

### Frågeparametern `fältnamn.op` (jämförelseoperator)

Anger vilken jämförelseoperator som ska användas på det fältnamn som föregår operatorm. Jämförelseoperatorerna används med frågekommandot `-find`.

**Värdet är:** den operator du vill använda. Standardoperatorm är ”begins with” (börjar med). Följande giltiga operatormer finns:

Nyckelord	FileMaker Pro, motsvarande operator
eq	= ord
cn	*ord*
bw	ord*
ew	*ord
gt	> ord
gte	>= ord
lt	< ord
lte	<= ord
neq	uteslut, ord

**Kan användas med:** frågekommandot `-find`

**Kräver:** Ett fältnamn och ett värde

Syntax för en jämförelseoperator är:

`tabellnamn::fältnamn=värde&tabellnamn::fältnamn.op=op-symbol`

där:

- `tabellnamn` är tabellen som innehåller fältet. Det krävs bara om fältet inte finns i källtabellen i den layout som anges i frågesträngen.
- `op-symbol` är ett av nyckelorden i föregående tabell, till exempel `cn`.

**Exempel:**

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&name=Tim&name.op=cn&-find`

Du kan använda valfri sökoperator i FileMaker Pro genom att ange nyckelordet **bw**. Om du exempelvis vill söka efter ett intervall med värden med intervalloperatorn (...), anger du nyckelordet **bw** och skriver sedan tecknen "...” före sökvillkoret.

**Exempel:**

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&IDnum=915...925&IDnum.op=bw
&-find
```

Mer information om vilka operatörer du kan använda när du söker efter text finns i FileMaker Pro Hjälp.

### Frågeparametern **-grammar** (syntax för XSLT-formatmallar)

Anger vilken syntax som ska användas för XSLT-formatmallar. Detta frågekommando kan endast användas tillsammans med anpassad webbpublicering med XSLT-förfrågningar.

**Värdet är:** fmresultset, FMPXMLRESULT eller FMPXMLLAYOUT

**Krävs för:** alla XSLT-förfrågningar.

**Exempel:**

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?-grammar=fmresultset&-db=mydatabase
&-lay=mylayout&-findall
```

Se ”Ange en XML-syntax för en XSLT-formatmall för FileMaker” på sidan 56.

### Frågeparametern **-lay** (layout)

Anger den databaslayout du vill använda.

**Värdet är:** namnet på layouten.

**Krävs för:** Alla frågekommandon utom **-dbnames**, **-layoutnamn**, **-manusnamn** och **-process** (endast XSLT-förfrågningar).

**Exempel:**

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-view
```

### Frågeparametern **-lay.response** (växla layout vid svar)

Anger att FileMaker Server ska använda den layout som anges med parametern **-lay** vid bearbetning av en förfrågan och att layouten ska växla till den layout som anges med parametern **-lay.response** när XML-svaret bearbetas.

Om du inte inkluderar parametern **-lay.response** används den layout som anges av parametern **-lay** när både frågan och svaret bearbetas.

Du kan använda parametern **-lay.response** för XML-förfrågningar eller i XSLT-formatmallsförfrågningar.

**Värdet är:** namnet på layouten.

**Kan användas med:** alla frågekommandon förutom **-dbnames**, **-layoutnames**, **-scriptnames** och **-process** (endast XSLT-frågor).

**Exempel:**

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=Budget&Salary=100000&Salary.op=gt&-find
&-lay.response=ExecList
```

## Frågeparametern **-lop** (logisk operator)

Anger hur sökvillkoren i frågekommandot `-find` kombineras i en "and"-sökning eller en "or"-sökning.

**Värdet är:** and eller or (som måste skrivas med små bokstäver (gemener)). Om frågeparametern `-lop` inte inkluderas, använder frågekommandot `-find` värdet and.

**Kan användas med:** frågekommandot `-find`

**Obs!** Stöds inte av frågekommandot `-findquery`.

### Exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&Last+Name=Smith
&Birthdate=2/5/1972&-lop=and&-find
```

## Frågeparametern **-max** (maximalt antal poster)

Anger det maximala antalet poster som ska returneras.

**Värdet är:** ett tal. Du kan också använda värdet all om du vill att alla poster ska returneras. Värdet all måste skrivas med små bokstäver (gemener). Om du inte anger `-max` returneras alla poster.

**Kan användas med:** frågekommandot `-find` eller `-findall` och `-findquery`

### Exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-max=10&-findall
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-max=all&-findall
```

## Frågeparametern **-modid** (ändrings-ID)

Ändrings-ID är en inkrementell räknare som anger den aktuella versionen av en post. Genom att ange ett ändrings-ID tillsammans med frågekommandot `-edit` kan du vara säker på att du redigerar den aktuella versionen av en post. Om det ändrings-ID du anger inte matchar aktuellt ändrings-ID i databasen, kan frågekommandot `-edit` inte användas. I stället returneras en felkod.

**Värdet är:** ett ändrings-ID, som är en unik identifierare för den aktuella versionen av en post i en FileMaker-databas.

**Kan användas med:** frågekommandot `-edit`

**Kräver:** parametern `-recid`

### Exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=22&-modid=6
&last_name=Jones&-edit
```

## Frågeparametern **-query** (sammanfattad sökning)

Anger frågenamn och sökkriterium för en sammansatt sökpost. Se "findquery (sammanfattad sökning), frågekommando" på sidan 95.

**Värdet är:** Ett frågeuttryck.

**Krävs för:** frågekommando `-findquery`

Syntaxen för en sammansatt sökpost är:

```
-query=<request-declarations><request-definitions>&-findquery
```

Där:

<request-declarations> är två eller flera frågedeklarationer.

- Varje frågedeklaration är sammansatt av en eller flera identifierare, separerade av kommatecken och omgivna av parentestecken. En frågeidentifierare är bokstaven ”q” följt av ett nummer. Till exempel: q1
- Omgiven av parenteser, fungerar flera frågor som logiska sökningar med AND vilka begränsar sökresultatet. I exemplet (q1);(q2) returneras poster som matchar q1 och q2.
- I FileMaker Pro kan varje fråga vara antingen en sökfråga eller en uteslutningsfråga. En sökfråga lägger till de matchande posterna i sökresultatet. En uteslutningsfråga tar bort de matchande posterna från sökresultatet. Standardinställningen är sökfråga. För en uteslutningsfråga placerar du ett utropstecken (!) framför det första parentestecknet.

Till exempel: (q1)!(q2)

I detta exempel är q1 en sökfråga. q2 är en uteslutningsfråga eftersom den föregås av ett utropstecken.

- Frågor separeras av semikolon. Flera sökfrågor fungerar som logiska OR-sökningar som utökar sökresultatet. I exemplet (q1);(q2) returneras poster som matchar q1 eller q2. Uteslutningsfrågor fungerar inte som logiska OR-sökningar eftersom uteslutningsfrågor tar bort poster från sökresultatet.
- Frågor utförs i den ordning de anges. Sökresultatet inkluderar resultatet av hela den sammansatta sökposten.

<request-definitions> är en frågedefinition för varje frågedeklaration. Varje frågedefinition består av ett sökfält och värdedefinitioner. Ett minustecken (–) startar sökdefinitionen.

Syntax:

–<query-id>=<fieldname>&–<query-id>.value=<value>

Till exempel:

–q1=typeofanimal&–q1.value=Cat

–q2=name&–q2.value=Fluffy

### Exempel:

Sök poster med gråa katter som inte heter ”Fluffy.”

[http://host/fmi/xml/fmresultset.xml?–db=petclinic&–lay=Patients&–query=\(q1, q2\)!\(q3\)&–q1=typeofanimal&–q1.value=Cat&–q2=color&–q2.value=Gray&–q3=name&–q3.value=Fluffy&–findquery](http://host/fmi/xml/fmresultset.xml?–db=petclinic&–lay=Patients&–query=(q1, q2)!(q3)&–q1=typeofanimal&–q1.value=Cat&–q2=color&–q2.value=Gray&–q3=name&–q3.value=Fluffy&–findquery)

## Frågeparametern –recid (post-ID)

Anger posten du vill bearbeta. Används mest av frågekommandona –edit och –delete. Används av visningskommandot –för att hämta data från en relaterad värdelista i syntaxen FMPXMLLAYOUT.

**Värdet är:** ett post-ID, som är unikt för en viss post i en FileMaker-databas.

**Krävs för:** Frågekommandona –edit, –delete och –dup

**Kan användas med:** Kommandona –find query och –view

### Exempel 1:

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–recid=22&–delete>

### Exempel 2:

<http://localhost/fmi/xml/FMPXMLLAYOUT.xml?–db=test&–lay=empty&–view&–recid=9>

### –relatedsets.filter (filtrera portalposter), frågeparameter

Anger om raderna som blir resultatet ska filtreras för frågor som använder portalfält.

**Värdet är:** layout eller none.

- Standardvärdet är none om den här parametern inte har angetts.
- Om frågan anger värdet layout respekteras inställningarna som har angetts i dialogrutan FileMaker Pro Portalkonfiguration. Posterna sorteras baserat på den sortering som har angetts i dialogrutan Portalkonfiguration, med postuppsättningen filtrerad för att starta med den först angivna raden.
- Om inställningen Visa vertikal rullningslist har aktiverats i dialogrutan Portalkonfiguration kan du använda alternativet –relatedsets.max när du vill ange maximalt antal rader som ska returneras som svar på frågan.
- Om inställningen Visa vertikal rullningslist är inaktiverad kommer inställningen i dialogrutan Antal rader i dialogrutan i Portalkonfiguration i FileMaker Pro att avgöra det maximala antalet rader som ska visas.
- Om –relatedsets.filter har angetts till none kommer Web Publishing Engine att returnera alla rader i portalen och portalrader som inte för försorterade.

**Kan användas med:** –find, –edit, –new, –dup och –findquery.

#### Exempel:

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=none&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&relatedsets.filter=layout
&–relatedsets.max=all&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=layout
&–relatedsets.max=10&–findany
```

### –relatedsets.max (begränsa portalposter), frågeparameter

Anger maximalt antal rader som returneras i resultatet för den här frågan.

**Värdet är:** ett heltal eller alla.

- Om du anger ett heltal returnerar Web Publishing Engine antalet rader efter att den första raden har returnerats.
- Om frågan anger alla returnerar Web Publishing Engine alla relaterade poster.
- Om frågan inte anger parametern –relatedsets.max bestäms antalet rader av det värde som anges i parametern –relatedsets.filter. Se ”–relatedsets.filter (filtrera portalposter), frågeparameter” på sidan 102.

**Kan användas med:** –find, –edit, –new, –dup och –findquery.

#### Exempel:

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&relatedsets.filter=layout
&–relatedsets.max=all&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=layout
&–relatedsets.max=10&–findany
```

### Frågeparametern –script (manus)

Anger vilket FileMaker-manus som ska utföras efter att frågekommandot och sorteringen har utförts. Se ”Bearbetning av en XML-förfrågan” på sidan 49.

**Värdet är:** manusets namn.

**Kan användas med:** alla frågekommandon förutom –dbnames, –layoutnames, –process och –scriptnames

#### Exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–script=myscript&–findall
```

### –script.param (skicka parameter till manus), frågeparameter

Skickar en parameter till det FileMaker-manus som anges av –script

**Värdet är:** En enda textparameter.

- Om du vill skicka flera parametrar kan du skapa en sträng som avgränsar parametrarna och låta ditt manus tolka de enskilda parametrarna. Om du t.ex. skickar ”param1|param2|param3” som en lista med tecknet ”|” URL-kodat som denna: param1%7Cparam2%7Cparam3
- För att textparameteren ska behandlas som ett värde som inte är text kan ditt manus konvertera textvärdet. Om du t.ex. vill konvertera textvärdet till ett tal kan ditt manus inkludera följande:  
GetAsNumber(Get(ScriptParam))
- Om din fråga innehåller –script.param utan –script kommer –script.param att ignoreras.
- Om din fråga innehåller flera än en –script.param kommer Web Publishing Engine att använda det sista värde den tolkar.

**Kan användas med:** –script

**Exempel:**

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script=myscript  
&-script.param=Smith%7CChatterjee%7CSu&-findall

### Frågeparameteren –script.prefind (manus före sökning)

Anger vilket FileMaker-manus som ska köras före sökning och sortering av poster (om så har angetts) under bearbetningen av frågekommandot –find.

**Värdet är:** manusets namn.

**Kan användas med:** alla frågekommandon förutom –dbnames, –layoutnames, –process och –scriptnames

**Exempel:**

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script.prefind=myscript&-findall

### –script.prefind.param (skicka parameter till manus före sökning), frågeparameter

Skickar en parameter till det FileMaker-manus som anges av –script.prefind

**Värdet är:** En enda textparameter.

- Om du vill skicka flera parametrar kan du skapa en sträng som avgränsar parametrarna och låta ditt manus tolka de enskilda parametrarna. Om du t.ex. skickar ”param1|param2|param3” som en lista med tecknet ”|” URL-kodat som denna: param1%7Cparam2%7Cparam3
- För att textparameteren ska behandlas som ett värde som inte är text kan ditt manus konvertera textvärdet. Om du t.ex. vill konvertera textvärdet till ett tal kan ditt manus inkludera följande:  
GetAsNumber(Get(ScriptParam))
- Om din fråga innehåller –script.prefind.param utan –script.prefind ignoreras –script.prefind.param.
- Om din fråga innehåller flera än en –script.prefind.param kommer Web Publishing Engine att använda det sista värde den tolkar.

**Kan användas med:** –script.prefind

**Exempel:**

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script.prefind=myscript  
&-script.prefind.param=payroll&-findall

## Frågeparametern `-script.presort` (manus före sortering)

Anger vilket FileMaker-manus som ska köras efter sökning av poster (om så har angetts) och före sortering av poster under bearbetningen av frågekommandot `-find`.

**Kan användas med:** alla frågekommandon förutom `-dbnames`, `-layoutnames`, `-process` och `-scriptnames`

### Exempel:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script.presort=myscript&-sortfield.1=dept&-sortfield.2=rating&-findall`

## `-script.presort.param` (skicka parameter till manus före sökning), frågeparameter

Skickar en parameter till det FileMaker-manus som anges av `-script.presort`

**Värdet är:** En enda textparameter.

- Om du vill skicka flera parametrar kan du skapa en sträng som avgränsar parametrarna och låta ditt manus tolka de enskilda parametrarna. Om du t.ex. skickar ”param1|param2|param3” som en lista med tecknet ”|” URL-kodat som denna: `param1%7Cparam2%7Cparam3`
- För att textparametern ska behandlas som ett värde som inte är text kan ditt manus konvertera textvärdet. Om du t.ex. vill konvertera textvärdet till ett tal kan ditt manus inkludera följande:  
`GetAsNumber(Get(ScriptParam))`
- Om din fråga innehåller `-script.presort.param` utan `-script.presort` ignoreras `-script.presort.param`.
- Om din fråga innehåller flera än en `-script.presort.param` kommer Web Publishing Engine att använda det sista värde den tolkar.

**Kan användas med:** `-script.presort`

### Exempel:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script.presort=myscript&-script.presort.param=18%7C65&-sortfield.1=dept&-sortfield.2=rating&-findall`

## Frågeparametern `-skip` (hoppa över poster)

Anger hur många poster som ska hoppas över i sökresultatet.

**Värdet är:** ett tal. Om värdet är större än antalet poster i sökresultatet visas inga poster. Standardvärdet är 0.

**Kan användas med:** frågekommandot `-find`

I följande exempel hoppas de 10 första posterna i sökresultatet över och posterna 11 till och med 15 returneras.

### Exempel:

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-skip=10&-max=5&-findall`

## Frågeparametern –sortfield (sortera fält)

Anger vilket fält som ska användas för sortering.

**Värdet är:** fältets namn.

**Kan användas med:** frågekommandona –find eller –findall

Frågeparametern –sortfield kan användas flera gånger om du vill göra upprepade fältsorteringar.

Använd följande syntax när du vill ange vilken prioritet sorteringsfälten ska ha:

–sortfield.precedence-number=fully-qualified-field-name

där prioritetsordning i frågeparametern –sortfield.prioritetsordning är ett tal (som börjar med 1) som anger prioritet när flera fält ska sorteras.

I följande exempel sorteras fältet dept först och därefter fältet rating. Båda fälten sorteras i stigande ordning eftersom frågeparametern –sortorder inte har angetts.

### Exempel:

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=performance&-sortfield.1=dept  
&-sortfield.2=rating&-findall

## Frågeparametern –sortorder (sorteringsordning)

Anger sorteringsordningens riktning.

**Värdet är:**Sorteringsordningen. Giltiga sorteringsordningar visas nedan, där <value-list-name> är namnet på en värdelista, till exempel Custom:

Nyckelord	FileMaker Pro, motsvarande operator
ascend (stigande)	Sortera a till ö, -10 till 10
descend (fallande)	Sortera z till a, 10 till -10
<värdelista>	Sortera enligt den angivna värdelistan som är associerad med fältet i layouten

**Kan användas med:** frågekommandona –find eller –findall

**Kräver:** frågeparametern –sortfield

Du kan använda frågeparametern –sortorder tillsammans med frågeparametern –sortfield om du vill ange sorteringsordning för flera sorteringsfält. Använd följande syntax när du vill ange sorteringsordningen för ett sorteringsfält:

–sortorder.precedence-number=sorteringsmetod

där:

- precedence-number i parametern –sortorder.precedence-number är ett tal mellan 1 och 9 som anger den –sortfield-frågeparameter som frågeparametern –sortorder ska användas på.
- sorteringsmetod är ett av nyckelorden i föregående tabell som anger sorteringsordning, till exempel ascend.

I följande exempel är sorteringsordningen för det högst prioriterade sorteringsfältet (dept) stigande och sorteringsordningen för det näst högst prioriterade sorteringsfältet (rating) är fallande. precedence-number 2 i –sortorder.2 anger att frågeparametern –sortorder.2=descend gäller för frågeparametern –sortfield.2=rating.

### Exempel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=performance&-sortfield.1=dept  
&-sortorder.1=ascend&-sortfield.2=rating&-sortorder.2=descend&-findall
```

**Obs!** Om ingen `-sortorder`-frågeparameter anges för ett sorteringsfält används standardsorteringen, som är stigande.

### **-stylehref (Style href), frågeparameter**

Genererar en bearbetningsinstruktion för en XML-formatmall i utdatadokumentet. Värdet för href-attributet anges (`href=/mystylesheet.css` eller `href=/stylesheets/mystylesheet.xml`), så att du kan använda överlappande formatmallar (CSS) eller XSLT-formatmallar på klientsidan i XML-dokumentet. Värdet för parametern `-stylehref` måste innehålla en absolut sökväg. Namnet på formatmallen kan vara vilket som helst, men det måste innehålla ändelsen `.css` eller `.xml`. Se "Bearbeta formatmallar på serversidan och klientsidan" på sidan 50. Denna parameter används tillsammans med parametern `-styletype`.

**Kan användas med:** alla frågekommandon.

**Kräver:** parametern `-styletype`

**Exempel** (förutsatt att `mystylesheet.xml` finns i webbserverprogrammets rotmapp):

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-styletype=text/xml  
&-stylehref=/mystylesheet.xml&-findall
```

### **Frågeparametern -styletype (formatmallstyp)**

Skapar en bearbetningsinstruktion för en XML-formatmall i utdatadokumentet. Värdet för typattributet anges (`type=text/css` eller `type=text/xml`), så att du kan använda överlappande formatmallar (CSS) eller XSLT-formatmallar på klientsidan i XML-dokumentet. Se "Bearbeta formatmallar på serversidan och klientsidan" på sidan 50. Denna parameter används tillsammans med parametern `-stylehref`.

**Kan användas med:** alla frågekommandon.

**Kräver:** parametern `-stylehref`

**Exempel** (förutsatt att `mystylesheet.css` finns i webbserverprogrammets rotmapp):

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-styletype=text/css  
&-stylehref=/mystylesheet.css&-findall
```

### **-token.[string] (överför värden mellan XSLT-formatmallar), frågeparameter**

Överför användardefinierad information mellan XSLT-formatmallar utan sessioner eller cookies. Denna parameter kan endast användas med anpassad webbpublicering med XSLT-förfrågningar.

**sträng i -token.[sträng] är:** en alfanumerisk sträng av valfri längd, förutom blanksteg, inklusive siffrorna 0-9, små bokstäver a-z eller stora bokstäver A-Z.

**Det användardefinierade parametervärdet är:** en teckensträng som är URL-kodad.

**Kan användas med:** alla XSLT-förfrågningar.

**Exempel:**

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xml?-db=employees&-lay=departments  
&-grammar=fmresultset&-token.D100=Active&-findall
```

Se "Överföra information mellan formatmallar med hjälp av token" på sidan 61.

# Bilaga B

## Felkoder för anpassad webbpublicering

Web Publishing Engine stöder tre typer av felkoder för anpassad webbpublicering:

- **För databaser och frågesträngar.** Web Publishing Engine genererar en kod för en publicerad databas varje gång en XML-dataförfrågan görs. Se nästa avsnitt, "Felkodsnummer för FileMaker-databaser".
- **Fel för Web Publishing Engine.** När Web Publishing Engine är i utvecklingsläge genereras en felsida när ett fel uppstår i själva Web Publishing Engine. I produktionsläge visas ett allmänt textmeddelande. Se "Felkodsnummer för Web Publishing Engine" på sidan 114.
- **Fel i FileMaker-tilläggsfunktioner för XSLT.** Du kan använda tilläggsfunktionen `fmxml:check_error_status()` i en XSLT-formatmall när du vill kontrollera felstatus i tilläggsfunktionerna när de anropas. Se "Felkodsnummer för FileMaker-tilläggsfunktioner för XSLT" på sidan 115.

En lista med uppdaterade felkoder hittar du i FileMaker Knowledge Base (<http://www.filemaker.com/kb/>).

### Felkodsnummer för FileMaker-databaser

Web Publishing Engine genererar en felkod för databaser som är publicerade i XML-format varje gång data efterfrågas. Denna typ av felkod infogas i början av XML-dokumentet i elementet `<error code>` för `fmresultset`-syntaxen eller i elementet `<ERRORCODE>` för `FMPXMLRESULT` eller `FMPDSORESLT`-syntaxen. Felkod 0 betyder att inget fel har inträffat.

Här är ett exempel på en databasfelkod i `fmresultset`-syntaxen:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN" "fmi/xml/fmresultset.dtd">
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
  <error code="0"></error>
```

Här är ett exempel på en databasfelkod i `FMPXMLRESULT`-syntaxen:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN" "fmi/xml/FMPXMLRESULT.dtd">
<fmpxmlresult xmlns="http://www.filemaker.com/fmpxmlresult">
  <ERRORCODE>0</ERRORCODE>
```

Som utvecklare av en anpassad webbpubliceringslösning måste du kontrollera värdet för elementet `<error code>` eller `<ERRORCODE>` och hantera det på rätt sätt. Web Publishing Engine hanterar inte databasfel.

Felnummer	Beskrivning
-1	Okänt fel
0	Inget fel
1	Användaren har avbrutit åtgärden
2	Minnesfel
3	Kommandot är inte tillgängligt (exempelvis fel operativsystem, fel läge osv)
4	Okänt kommando
5	Ogiltigt kommando (ett Tilldela fält-steg har exempelvis ingen beräkning angiven)

<b>Felnummer</b>	<b>Beskrivning</b>
6	Filen är skrivskyddad
7	Minnet räcker inte.
8	Tomt resultat
9	Otillräcklig behörighet
10	Begärda data saknas
11	Namnet är ogiltigt
12	Namnet finns redan
13	Filen eller objektet används
14	Utanför det tillåtna intervallet
15	Går inte att dela med noll
16	Åtgärden misslyckades, begär nytt försök (t.ex. en användarfråga)
17	Försök att konvertera främmande teckenuppsättning till UTF-16 misslyckades
18	Klienten måste ange kontoinformation för att fortsätta
19	Strängen innehåller andra tecken än A-Z, a-z, 0-9 (Ascii)
20	Kommando eller åtgärd avbröts av utlöst manus
100	Fil saknas
101	Post saknas
102	Fält saknas
103	Relation saknas
104	Manus saknas
105	Layout saknas
106	Tabell saknas
107	Index saknas
108	Värde lista saknas
109	Behörighetsuppsättning saknas
110	Relaterade tabeller saknas
111	Fältrepetitionen är ogiltig
112	Fönstret saknas
113	Funktionen saknas
114	Filreferensen saknas
115	Menyuppsättningen saknas
116	Layoutobjektet saknas
117	Datakällan saknas
130	Filerna är skadade eller saknas och måste installeras om
131	Språkpaketfiler saknas (t.ex. mallfiler)
200	Tillgång till posten nekad
201	Det går inte att ändra fältet

Felnummer	Beskrivning
202	Tillgång till fältet nekad
203	Det finns inga poster i filen att skriva ut eller så saknar lösenordet behörighet för utskrift
204	Ingen tillgång till fält i sorteringsordning
205	Användaren har inte behörighet att skapa nya poster; importen kommer att skriva över befintliga data
206	Användaren har inte behörighet att ändra lösenord eller så kan filen inte ändras
207	Användaren har inte tillräcklig behörighet för att ändra databasschemat eller så kan filen inte ändras
208	Lösenordet innehåller inte tillräckligt många tecken
209	Det nya lösenordet måste skilja sig från det befintliga
210	Användarkontot är inte aktivt
211	Lösenordet har upphört att gälla
212	Ogiltigt användarkonto och/eller lösenord. Försök igen
213	Användarkontot och/eller lösenordet finns inte
214	För många inloggningsförsök
215	Administratörsbehörighet kan inte dupliceras
216	Gästkontot kan inte dupliceras
217	Användaren har inte tillräcklig behörighet för att ändra administratörskontot
300	Filen är låst eller används
301	Posten används av en annan användare
302	Tabellen används av en annan användare
303	Databasschemat används av en annan användare
304	Layouten används av en annan användare
306	Postens ändrings-ID matchar inte
400	Sökvillkoren är tomma
401	Inga poster matchar sökposten
402	Det valda fältet är inte ett matchande fält för en länk
403	Maxpostgränsen för testversionen av FileMakerPro har överskridits
404	Sorteringsordningen är ogiltig
405	Det angivna antalet poster överskrider antalet poster som kan utelämnas
406	Ogiltiga villkor för ersättning/ändring av löpnummer
407	Ett av eller båda de matchande fälten saknas (ogiltig relation)
408	Det angivna fältet har fel datatyp för den här åtgärden
409	Importordningen är ogiltig
410	Exportordningen är ogiltig
412	Fel version av FileMakerPro används för att reparera filen
413	Det angivna fältet är av fel typ
414	Resultatet kan inte visas i layouten
415	En eller flera nödvändiga relaterade poster är inte tillgängliga

Felnummer	Beskrivning
416	En primär nyckel krävs från datakällstabellen
417	Databasen är inte en datakälla som stöds
500	Datumvärdet stämmer inte med kontrolltillvalen
501	Tidsvärdet stämmer inte med kontrolltillvalen
502	Det numeriska värdet stämmer inte med kontrolltillvalen
503	Värdet i fältet stämmer inte med intervallet som angetts i kontrolltillvalen
504	Värdet i fältet är inte unikt, vilket angetts som ett krav i kontrolltillvalen
505	Värdet i fältet är inte ett befintligt värde i databasfilen, vilket angetts som ett krav i kontrolltillvalen
506	Värdet i fältet finns inte med i värdelistan som angetts i kontrolltillvalet
507	Värdet i fältet klarade inte beräkningstestet för kontrolltillvalet
508	Ett ogiltigt värde har angetts i sökläget
509	Fältet kräver ett giltigt värde
510	Det relaterade värdet är tomt eller inte tillgängligt
511	Värdet i fältet överskrider antalet tillåtna tecken
512	Posten har redan ändrats av en annan användare
513	För att skapa en post måste posten ha ett värde i minst ett fält
600	Utskriftsfel
601	Sidhuvudet och sidfoten ryms inte tillsammans på en sida
602	Huvuddelen ryms inte på en sida med den aktuella kolumninställningen
603	Ingen skrivaranslutning
700	Fel filformat för import
706	EPSF-filen saknar granskningsbild
707	Det går inte att hitta grafikkonverteraren
708	Det går inte att importera filen eller så krävs stöd för färgskärm för att importera filen
709	Import av QuickTime-film misslyckades
710	Det går inte att uppdatera QuickTime-filmreferensen eftersom databasfilen är skrivskyddad
711	Det går inte att hitta importkonverteraren
714	Lösenordet saknar behörighet för den här åtgärden
715	Det angivna Excel-kalkylbladet eller intervallet saknas
716	En SQL-fråga med DELETE, INSERT eller UPDATE är inte tillåten vid ODBC-import
717	Det finns inte tillräcklig XML/XSL-information för att importen eller exporten ska kunna fortsätta
718	Ett fel inträffade vid tolkningen av XML-filen (från Xerces)
719	Ett fel inträffade vid konverteringen av XML med hjälp av XSL (från Xalan)
720	Ett fel inträffade vid exporten, det angivna formatet har inte stöd för repeterade fält
721	Ett okänt fel inträffade vid tolkningen eller konverteringen
722	Det går inte att importera data till en fil utan fält
723	Du har inte behörighet att lägga till eller ändra poster i måltabellen

Felnummer	Beskrivning
724	Du har inte behörighet att lägga till poster i måltabellen
725	Du har inte behörighet att ändra poster i måltabellen
726	Importfilen innehåller fler poster än måltabellen. Alla poster importerades inte
727	Måltabellen innehåller fler poster än importfilen. Alla poster uppdaterades inte
729	Fel inträffade vid importen. Posterna kunde inte importeras
730	Excel-versionen stöds inte. Konvertera filen till Excel 7.0 (Excel 95), 97, 2000, XP eller 2007 och försök igen.
731	Filen som du importerar från innehåller inga data
732	Den här filen kan inte sättas in eftersom den innehåller andra filer
733	En tabell kan inte importeras till sig själv
734	Den här filtypen kan inte visas som en bild
735	Den här filtypen kan inte visas som en bild. Den kommer att sättas in och visas som en fil.
736	Det finns för många data för att exportera till detta format. Informationen kommer att trunkeas.
737	Den Bento-tabell du försöker importera finns inte.
800	Det går inte att skapa filen på disken
801	Det går inte att skapa någon tillfällig fil på systemdisken
802	Det går inte att öppna filen Detta fel kan orsakas av något av följande: <ul style="list-style-type: none"> <li>■ Ogiltigt databasnamn</li> <li>■ Filen är stängd i FileMakerServer</li> <li>■ Ogiltig behörighet</li> </ul>
803	Filen är en enanvändarfil eller så går det inte att hitta värden
804	Filen går inte att öppna som skrivskyddad i aktuellt tillstånd
805	Filen är skadad; använd kommandot Reparera
806	Det går inte att öppna filen i den här versionen av FileMaker Pro
807	Filen är inte en FileMakerPro-fil eller så är den allvarligt skadad
808	Det går inte att öppna filen eftersom behörigheten är skadad
809	Disken/volymen är full
810	Disken/volymen är låst
811	Det går inte att öppna den tillfälliga filen som en FileMaker Pro-fil
813	Postsynkroniseringsfel i nätverket
814	Det går inte att öppna filen eftersom maximala antalet tillåtna filer redan är öppna
815	Det går inte att öppna den länkade filen
816	Det går inte att konvertera filen
817	Det går inte att öppna filen eftersom den inte hör till denna lösning
819	Det går inte att spara en lokal kopia av en fjärrfil
820	Filen håller på att stängas
821	Frånkoppling framtvindad av värden
822	FMI-filer hittades inte; installera om saknade filer

Felnummer	Beskrivning
823	Det går inte att göra filen till en enanvändarfil eftersom gäster är anslutna
824	Filen är skadad eller så är den inte en FileMaker-fil
825	Filen har inte behörighet att referera till den skyddade filen
900	Allmänt fel i stavningsmotorn
901	Standardordlistan är inte installerad
902	Det gick inte att starta hjälpsystemet
903	Detta kommando kan inte användas i en delad fil
904	Kommandot kan bara användas i en fil som delas under FileMaker Server
905	Inget aktivt fält har valts och kommandot kan bara användas om det finns ett aktivt fält
906	Den aktuella filen är inte delad; kommandot kan endast användas för delade filer
920	Det går inte att initiera stavningsmotorn
921	Det går inte att öppna den egna ordlistan för redigering
922	Det går inte att hitta den egna ordlistan
923	Den egna ordlistan är skrivskyddad
951	Ett oväntat fel har inträffat
954	XML-syntax stöds inte
955	Inget databasnamn
956	Det maximala antalet databassessioner har överskridits
957	Motstridiga kommandon
958	Parameter saknas i fråga
959	Tekniken för anpassad webbpublicering är inaktiverad
1200	Allmänt beräkningsfel
1201	För få parametrar i funktionen
1202	För många parametrar i funktionen
1203	Oväntat slut på beräkningen
1204	Ett numeriskt värde, en textkonstant, ett fältnamn eller "(" krävs
1205	Kommentaren avslutas inte med "*/"
1206	Textkonstanten måste avslutas med ett citationstecken
1207	Udda parentes
1208	Operator saknas, funktionen hittades inte eller "(" förväntades inte
1209	Ett namn (t.ex. ett fältnamn eller ett layoutnamn) saknas
1210	Plugin-funktionen har redan registrerats
1211	Det är inte tillåtet att använda listor i den här funktionen
1212	En operator (t.ex. +, -, *) behövs här
1213	Den här variabeln har redan definierats i Let-funktionen
1214	AVERAGE, COUNT, EXTEND, GETREPETITION, MAX, MIN, NPV, STDEV, SUM och GETSUMMARY: ett uttryck påträffades där det bara behövs ett fält
1215	Den här parametern är en ogiltig Get-funktionsparameter

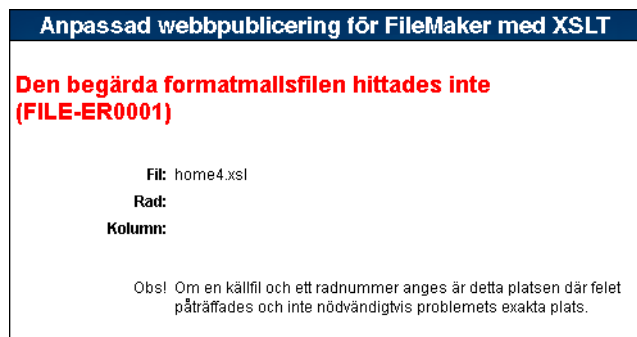
Felnummer	Beskrivning
1216	Första argumentet i en GETSUMMARY måste vara ett statistikfält
1217	Brytfältet är ogiltigt
1218	Det går inte att utvärdera talet
1219	Ett fält kan inte användas i sin egen formel
1220	Fälttypen måste vara Normal eller Beräkning
1221	Datotypen måste vara numerisk, datum, tid eller tidsstämpel
1222	Beräkningen får inte vara lagrad
1223	Funktionen som hänvisas till har inte implementerats än
1224	Den refererade funktionen finns inte
1225	Funktionen som hänvisas till stöds inte i denna kontext
1400	Det gick inte att initiera ODBC-klientdrivrutinen. Kontrollera att ODBC-klientdrivrutinerna är korrekt installerade.
1401	Det gick inte att ställa in miljön (ODBC)
1402	Det gick inte att frigöra miljön (ODBC)
1403	Det gick inte att koppla från (ODBC)
1404	Det gick inte att tilldela en anslutning (ODBC)
1405	Det gick inte att frigöra en anslutning (ODBC)
1406	Kontroll av SQL-API misslyckades (ODBC)
1407	Det gick inte att tilldela en instruktion (ODBC)
1408	Utökat fel (ODBC)
1409	Utökat fel (ODBC)
1410	Utökat fel (ODBC)
1411	Utökat fel (ODBC)
1412	Utökat fel (ODBC)
1413	Utökat fel (ODBC)
1450	Åtgärden kräver PHP-behörighetstillägg
1451	Åtgärden kräver att aktuell fil är en fjärrfil
1501	SMTP-verifieringen misslyckades
1502	Anslutningen avvisades av SMTP-servern
1503	Fel med SSL
1504	SMTP-servern kräver att anslutningen är krypterad
1505	Angiven verifiering stöds inte av SMTP-servern
1506	E-post kunde inte skickas
1507	Det går inte att logga in på SMTP-servern

## Felkodsnummer för Web Publishing Engine

När Web Publishing Engine är i utvecklingsläge genereras en felsida när ett fel uppstår i själva Web Publishing Engine. Denna typ av fel kan uppstå av olika anledningar, exempelvis om Web Publishing Engine inte kan

- hitta en efterfrågad eller kapslad (via <xsl:include>) formatmallfil
- tolka en efterfrågad eller kapslad formatmallfil på grund av ett XML-fel i filen
- generera en formatmall från filen på grund av ett XSLT- eller Xpath-fel i filen
- bearbeta förfrågan eftersom XML-syntaxen inte har angetts på rätt sätt i CGI
- kommunicera med webbpubliceringskärnan och hämta XML.

När Web Publishing Engine körs i utvecklingsläge innehåller felsidan för denna typ av fel ett felmeddelande och ett felnummer inom parentes. Till exempel:



Exempel på felsida när Web Publishing Engine är i utvecklingsläge

Här följer en lista över några av Web Publishing Engines felkoder:

Felkod	Beskrivning
QUERY-ER0001	Ingen XML-syntax har angetts i frågeparametern –syntax
QUERY-ER0002	”xxx” är inte en giltig XML-syntax för FileMaker XSLT
FILE-ER0001	Den begärda formatmallfilen hittades inte
FILE-ER0002	Den begärda filen kunde inte hittas
UNKNOWN	Ett oväntat fel har inträffat
MCS-000 till MCS-600	Ett oväntat fel har inträffat
MCS-601	Resursen ”x” kunde inte läsas in, eftersom det inte finns stöd för resurser av följande typ: ”x”
MCS-602	URL-adressen ”x” kunde inte tolkas
MCS-603	HTTP-förfrågan för ”x” returnerade ett fel av typen ”x”
MCS-604	Resursen ”x” kunde inte läsas in på grund av ett oväntat fel
MCS-605	Resursen ”x” kunde inte läsas in, eftersom innehållstypen var ogiltig
MCS-606	Resursen ”x” kunde inte läsas in på grund av ett XML-fel i dokumentet
MCS-607	Resursen ”x” kunde inte läsas in på grund av ett verifieringsproblem
MCS-700	Ett oväntat fel har inträffat
MCS-800	Ett oväntat fel har inträffat

Om Web Publishing Engine är i produktionsläge visas följande allmänna textmeddelande på felsidan `pe_server_error.html` för Web Publishing Engine-fel:

Ett oväntat fel inträffade vid användning av FileMakers funktion för anpassad webbpublicering och XSLT.

Standardfilen `pe_server_error.html` innehåller föregående textmeddelande på sex språk.

Som utvecklare kan du redigera texten på felsidan `pe_server_error.html` så att den passar din lösning. Filen `pe_server_error.html` finns i mappen `cwpe` i publiceringsmotormappen på den dator där du installerade Web Publishing Engine.

Information om hur du konfigurerar Web Publishing Engine för utvecklingsläget eller produktionsläget finns i hjälpen till ee FileMaker Server.

## Felkodsnummer för FileMaker-tilläggsfunktioner för XSLT

Tilläggsfunktionen `fmxml:check_error_status()` (se ”Kontrollera felstatus i tilläggsfunktioner” på sidan 78) returnerar ett av felen i följande tabell:

Felkod	Beskrivning
-1	Okänt fel
0	Inget fel
<b>Allmänna fel</b>	
10000	Ogiltigt filhuvudnamn
10001	Ogiltig HTTP-statuskod
<b>Sessionsfel</b>	
10100	Okänt sessionsfel
10101	Efterfrågat sessionsnamn används redan
10102	Det går inte att få åtkomst till sessionen (den kanske inte finns)
10103	Tidsgränsen för sessionen har överskridits
10104	Det angivna sessionsobjektet finns inte
<b>Meddelandefel</b>	
10200	Okänt meddelandefel
10201	Formateringsfel i meddelande
10202	SMTP-fältfel i meddelande
10203	”To Field”-fel i meddelande
10204	”From Field”-fel i meddelande
10205	”CC Field”-fel i meddelande
10206	”BCC Field”-fel i meddelande
10207	”Subject Field”-fel i meddelande
10208	”Reply-To Field”-fel i meddelande

<b>Felkod</b>	<b>Beskrivning</b>
10209	Brödtextfel i meddelande
10210	Rekursivt e-postfel. Försökte anropa send_email() i en XSLT-formatmall
10211	SMTP-verifieringsfel. Antingen misslyckades inloggningen eller så var fel typ av verifiering medskickad
10212	Ogiltig funktionsanvändning. Försökte anropa set_header(), set_status_code() eller set_cookie() i en XSLT-formatmall
10213	SMTP-servern är ogiltig eller fungerar inte.
<b>Formateringsfel</b>	
10300	Okänt formateringsfel
10301	Ogiltigt datum- och tidformat
10302	Ogiltigt datumformat
10303	Ogiltigt tidformat
10304	Ogiltigt dagformat
10305	Felaktigt formaterad datum- och tidssträng
10306	Felaktigt formaterad datumsträng
10307	Felaktigt formaterad tidssträng
10308	Felaktigt formaterad dagsträng
10309	Textkodning stöds inte
10310	Ogiltig URL-kodning
10311	Mönsterfel i reguljärt uttryck

# Index

## A

- Admin Console för Web Publishing Engine 35
- administrationskonsol för Web Publishing Engine 27
- aktivera anpassad webbpublicering i databas 19
- ange layout vid fråga om XML-data 49
- ange XML-syntax 56
- Anpassad Web Publishing Engine (CWPE) 26, 34
- Anpassad webbpublicering
  - aktivera i databas 19
  - aktivera i Web Publishing Engine 20
  - använda en statisk IP-adress 18
  - använda manus 22
  - begränsa åtkomst av IP-adresser på webbserver 20
  - beskrivning 15, 16
  - definition 11
  - Gäst, konto 20
  - krav för 17
  - manus 23
  - med PHP 13
  - med XML 13, 33
  - med XSLT 13, 26, 53
  - nya funktioner i 16
  - utökad behörighet för 19
  - webbanv. åtkomst till lösningar 19, 63
  - översikt 11
- användarnamn
  - ange för anpassad webbpublicering 19
  - grundlägg. verifiering av webbanv. 19, 63
  - åtkomst till XML-dokument 64
- ASCII-tecken, i XML-dokument 47
- automatisk inmatning, attribut 41

## B

- begränsa portalfältrader 102
- begäran om XML-data 35
- behandla en webbpubliceringsförfrågan 12
- behörighet 20
- behörighetsuppsättning, tilldela för anpassad webbpublicering 19
- break\_encode(), tilläggsfunktion 72
- buffring, använda i formatmall 65
- byta layout för ett XML-svar 49

## C

- check\_error\_status(), tilläggsfunktion 78, 115
- compare\_date(), tilläggsfunktion 75
- compare\_datetime(), tilläggsfunktion 76
- compare\_day(), tilläggsfunktion 75
- compare\_time(), tilläggsfunktion 75
- containerfält
  - hur webbanvändare använder data 21
  - publicera innehållet i 21, 30
  - URL-syntax för åtkomst i XML-lösningar 37
  - URL-syntax för åtkomst i XSLT-lösningar 55
- contains\_checkbox\_value(), tilläggsfunktion 73
- convert\_datetime(), tilläggsfunktion 76
- cookies
  - lagra sessions-ID 66
  - tilläggsfunktioner, använda 71
- create\_session(), tilläggsfunktion 67

## D

- dag, tilläggsfunktioner, använda 75
- databas, skydda publicerad 20
- databaser, felkoder 39
- databassessioner, aktivera 68, 93
- <datasource>-element 40
- datum, tilläggsfunktioner, använda 75
- datumformatsträngar 76
- db, frågeparameter 96
- dbnames, frågekommando 93
- definiera tilläggsfunktioner 78
- delete, frågekommando 93
- delete.related, frågeparameter 92
- Direkt webbpublicering
  - definition 11
  - dokumentation 9
- document(), funktion 64
- dokument, läsa in via funktionen document() 64
- dokumentation 9, 18
- dokumentation online 9
- dokumenttypdefinitioner (DTDer) 39, 43
- dup, frågekommando 94
- dölja metadata med formatmallar 25

**E**

- edit, frågekommando 94
- ej tomt attribut 41
- elektronisk dokumentation 9
- element
  - databasfelkoder 39
  - i FMPXMLLAYOUT-syntaxen 44
  - i syntaxen FMPXMLRESULT 43
  - i syntaxen fmresultset 40
- encoding, frågeparameter 97
- endast numeriskt, attribut 41
- e-postmeddelanden
  - tilläggsfunktioner för 68
  - ursprunglig standardinställning för kodning 59
- <error code> och <ERRORCODE>, element 107
- exempel på
  - FMPXMLLAYOUT-grammatikfil 46
  - FMPXMLRESULT -grammatikfil 44
  - fmresultset, genererad syntax 42
- exportera XML-data 33
- Extensible Markup Language (XML). *se* XML

**F**

- fel
  - databas, felkodsnummer 107
  - felkoder i databaser 39
  - kontrollera felstatus i tilläggsfunktioner 78, 115
  - loggfiler för webbserver 86
  - om felkoder 107
  - pe\_application\_log.txt, loggfil 86
  - pe\_server\_error.html, felsida 115
  - tilläggsfunktioner, felkodsnummer 115
  - Web Publishing Engine, felkodsnummer 114
- felsökning
  - Egna webbpubliceringsplatser 84
  - XML-dokumentåtkomst 51
  - XSLT-formatmallar 31
- field, frågeparameter (container) 97
- FileMaker API för PHP 13
  - definition 13
- FileMaker Pro, jämfört med Web Publishing Engine 33
- FileMaker Server
  - dokumentation 9
  - installera 9
- FileMaker Server Admin
  - se* Admin Console 20
- FileMaker Server Admin Console 35
- FileMaker Site Assistant. *Se* XSLT Site Assistant

- FileMaker-specifika XSLT-parametrar 61
- FileMaker-tilläggsfunktioner för XSLT
  - Se även* tilläggsfunktionerna för fmxslt
- filtrera data med formatmallar 25
- filtrera portalfältrader 102
- findall, frågekommando 94
- findany, frågekommandot 94
- findquery, frågekommando 95
- FMPDSORESULT, syntax
  - jämfört med andra syntaxer 38
- fmresultset, syntax 33, 39–42
  - jämfört med andra syntaxer 38
- fmxml, nyckelord för aktivering av XML-publicering 19, 35
- fmxslt, nyckelord för aktivering av XSLT-publicering 19, 27
- fmxslt, tilläggsfunktioner
  - fmxslt:break\_encode(), funktion 72
  - fmxslt:check\_error\_status(), funktion 78, 115
  - fmxslt:compare\_date(), funktion 75
  - fmxslt:compare\_datetime(), funktion 76
  - fmxslt:compare\_day(), funktion 75
  - fmxslt:compare\_time(), funktion 75
  - fmxslt:contains\_checkbox\_value(), funktion 73
  - fmxslt:convert\_datetime(), funktion 76
  - fmxslt:create\_session(), funktion 67
  - fmxslt:get\_cookie(), funktion 71
  - fmxslt:get\_cookies(), funktion 71
  - fmxslt:get\_date(), funktion 75
  - fmxslt:get\_datetime(), funktion 76
  - fmxslt:get\_day(), funktion 75
  - fmxslt:get\_fm\_date\_format() function 75
  - fmxslt:get\_fm\_time\_format() function 75
  - fmxslt:get\_fm\_timestamp\_format() function 75
  - fmxslt:get\_header(), funktion 70
  - fmxslt:get\_long\_date\_format(), funktion 75
  - fmxslt:get\_long\_day\_format(), funktion 75
  - fmxslt:get\_long\_time\_format(), funktion 75
  - fmxslt:get\_session\_object(), funktion 67
  - fmxslt:get\_short\_date\_format(), funktion 75
  - fmxslt:get\_short\_day\_format(), funktion 75
  - fmxslt:get\_short\_time\_format(), funktion 75
  - fmxslt:get\_time(), funktion 75
  - fmxslt:html\_encode(), funktion 72
  - fmxslt:invalidate\_session(), funktion 67, 68
  - fmxslt:regex\_contains(), funktion 73
  - fmxslt:remove\_session\_object(), funktion 67
  - fmxslt:send\_email(), funktioner 68
  - fmxslt:session\_encode\_url(), funktion 66, 67
  - fmxslt:session\_exists(), funktion 67
  - fmxslt:set\_cookie(), funktion 71

- fmxls: set\_header(), funktion 70
  - fmxls: set\_session\_object(), funktion 67
  - fmxls: set\_session\_timeout(), funktion 67
  - fmxls: set\_status\_code(), funktion 70
  - fmxls: url\_decode(), funktion 72
  - fmxls: url\_encode(), funktion 72
  - formatmallar
    - använda layoutinformation in 65
    - använda på webbplatser och i program 30
    - använda XSLT Site Assistant för att skapa 28
    - cookies, tilläggfunktioner 71
    - datum- och tidsformatsträngar 76
    - datum, tid och dag, tilläggfunktioner 75
    - e-postmeddelanden, skicka 68
    - exempel på användning 25
    - felstatus i tilläggfunktioner, kontrollera 78
    - frågesträngar för 56
    - grammar, parameter för 56
    - innehållsbuffering, använda 65
    - jämföra strängar via reguljära Perl-uttryck 73
    - klientsidan 50
    - koda 60
    - kryssrutor, söka efter värden i 73
    - om 25
    - på serversidan 25, 53
    - riktlinjer för utveckling 53
    - sessionsfunktion, använda 66, 67
    - sidhuvudfunktioner, använda 70
    - stränghantering, tilläggfunktioner 72
    - testa 84, 85
    - utveckla 53
    - XML-formatmall, bearbetningsinstruktion för 50
    - XSLT, beskrivning 25
  - formatmallar på klientsidan 35, 50
  - formatsträngar, datum och tid 76
  - fråga portalfält 92
  - frågeinformation, använda i förfrågan 62
  - frågekommandot –find. 94
  - frågestr. 47, 56, 89
    - efterfråga XML-data 47, 89
    - frågesträngar, referens 90
    - fullständigt fältnamn, syntax för 91
    - globala fält, syntax för 93
    - kommandon och param. 47, 56, 89
    - lägga till poster till portaler 91
    - redigera poster i portaler 92
    - riktlinjer för 90
    - statiskt definierade i XSLT-formatmallar 58
    - XSLT-formatmallar, använda i 56
  - frågesträngar, referens 90
  - fullständigt fältnamn, syntax för 91
  - fysiffrigt årtalsattribut 41
  - fält som använder global lagring
    - använda med sessioner 68, 93
    - databassessioner, aktivera 68, 93
    - syntax för 93
  - <fältdefinition>-element 41
  - fältnamn, frågeparameter (ej container) 97
  - fältnamn, fullständig syntax 91
  - fältnamn.op, frågeparameter 98
  - fördelar med XML och XSLT 14
- ## G
- get\_cookie() extension, funktion 71
  - get\_cookies(), tilläggfunktion 71
  - get\_date(), tilläggfunktion 75
  - get\_datetime(), tilläggfunktion 76
  - get\_day(), tilläggfunktion 75
  - get\_fm\_date\_format() tilläggfunktion 75
  - get\_fm\_time\_format() extension function 75
  - get\_fm\_timestamp\_format() extension function 75
  - get\_header(), tilläggfunktion 70
  - get\_long\_date\_format(), tilläggfunktion 75
  - get\_long\_day\_format(), tilläggfunktion 75
  - get\_long\_time\_format(), tilläggfunktion 75
  - get\_session\_object(), tilläggfunktion 67
  - get\_short\_date\_format(), tilläggfunktion 75
  - get\_short\_day\_format(), tilläggfunktion 75
  - get\_short\_time\_format(), tilläggfunktion 75
  - get\_time(), tilläggfunktion 75
  - Gif-filer, publicera på webben 21
  - globalt attribut 41
  - grammar, frågeparameter 56, 99
  - grundlägg. verifiering av webbanv. 19, 63
  - grundläggande URI-parameter 63
  - Gäst, konto
    - aktivering 20
    - inaktivera 20
    - med anpassad webbpublicering 20
- ## H
- ### HTML
- formulär för XML-begäran 35
  - utforma om XML-data till 33
  - html\_encode(), tilläggfunktion 72
  - hämtar layoutinformation 96
  - hämtar layoutnamn 95
  - hämtar tillgängliga manusnamn 96

**I**

importera XML-data 33  
 innehållsbuffering, använda 65  
 installationsdokumentation 9  
 integrera data med formatmallar 25  
 invalidate\_session(), tilläggsfunktion 67, 68  
 ISO-2022-JP, kodning 59  
 ISO-8859-1, kodning 59  
 ISO-8859-15, kodning 59

**J**

JavaScript  
   definiera tilläggsfunktioner 78  
 JDBC-dokumentation 9  
 Jpeg-filer, publicera på webben 21  
 jsessionid, parameter 66  
 jämföra strängar 73  
 jämförelseoperatorer för fält 98

**K**

klientinformation, hämta via XSLT-parametrar 62  
 kodning  
   använda tilläggsfunktioner för stränghantering 72  
   –encoding, frågeparameter 59, 97  
   förfrågningar 59  
   URL-adresser 37, 66  
   utdata via elementet <xsl:output> 60  
   XML-data 38, 47  
   XSLT-formatmallar 60  
 kommandon för frågor. *Se* frågestr.  
 konton och behörighet  
   aktivera för anpassad webbpublicering 19  
   Gäst, konto 20  
   manus 22  
 konvertera data med formatmallar 25  
 krav för anpassad webbpublicering 17  
 kryssrutor, söka efter värden i 73

**L**

lagra information i sessioner 66, 67  
 -lay, frågeparameter 49, 99  
 -lay.response, frågeparameter 49, 99  
 layout, byta för ett XML-svar 49  
 layoutinformation, använda i formatmall 65  
 -layoutnames, frågekommando 95

loggfiler 84, 87  
   beskrivning 86  
   logga via elementet <xsl:message> 78  
   pe\_application\_log.txt 86  
   pe\_internal\_access\_log.txt 87  
   web\_server\_module\_log.txt 87  
   webbserveråtkomst 86  
   <xsl:message>, element 86  
 loggmapp 86  
 -lop, frågeparameter 100  
 läge, spara i sessioner 66, 67  
 läsa in ytterligare dokument 64  
 lösenord  
   ange för anpassad webbpublicering 19  
   grundlägg. verifiering av webbanv. 19, 63  
   inget lösenord för inloggning 20  
   åtkomst till XML-dokument 64  
   Ändra lösenord, manussteg 20

**M**

manus  
   databassessioner, aktivera 68  
   för XML-förfrågningar 35  
   i anpassad webbpublicering 22  
   konton och behörighet 22  
   Logga in igen 20  
   tips och råd 22  
   utlösare 23  
   Ändra lösenord 20  
 Manus Logga in igen 20  
 mata ut data med formatmallar 25  
 -max, frågeparameter 100  
 maximalt antal repetitioner, attribut 41  
 maximalt antal tecken, attribut 41  
 meddelanden. *Se* e-postmeddelanden  
 metadata, dölja med formatmallar 25  
 <metadata>-element 41  
 metodattribut, <xsl:output>, element 60  
 MIME-typer (Multipurpose Internet Mail Extensions) 21  
 -modid, frågeparameter 100

**N**

namnattribut 41  
 namnuttryck för  
   XML 38  
   XSLT 57  
 -new, frågekommando 95

nummer för  
 databaser, felkoder 107  
 tilläggsfunktioner, felkoder 115  
 Web Publishing Engine, felkoder 114  
 nya funktioner i anpassad webbpublicering 16  
 nyckelord för aktivering av anpassad  
 webbpublicering 19, 27, 35

## O

ODBC-dokumentation 9  
 operatorer, jämförelse 98  
 ordningsföljd för XML-frågebearbetning 49

## P

param. för frågor. *Se* frågestr.  
 parametrar för XSLT, FileMaker-specifika 61  
 PDF-filer 9  
 pe\_application\_log.txt, loggfil 86  
 pe\_internal\_access\_log.txt, loggfil 87  
 pe\_server\_error.html, felsida 115  
 Perl, reguljära uttryck, jämföra strängar 73  
 PHP  
 felsökning 31  
 fördelar 14  
 PHP Api for anpassad webbpublicering 13  
 portaler  
 lägga till poster 91  
 namn 102  
 radera poster 92  
 redigera poster 92  
 sortera poster 102  
 portalfältfrågor 102  
 -process, frågekommando 60, 96  
 produktionsläge, Web Publishing Engine 115  
 programlogg 78, 86  
 publicera på webben  
 ansluta till Internet eller intranät 18  
 containerfältobjekt 21, 30  
 databaser, felkoder 107  
 krav för 17  
 med XML 15, 35  
 med XSLT 16, 27, 53  
 QuickTime-filmer 21  
 skydda databaser 20

## Q

-query, frågeparameter 100  
 QuickTime-filmer, publicera på webben 21

## R

radera portalposter 92  
 -recid, frågeparameter 101  
 regex\_contains(), tilläggsfunktion 73  
 <relatedset-definition>-element 41  
 -relatedsets.filter, frågeparameter 102  
 -relatedsets.max, frågeparameter 102  
 remove\_session\_object(), tilläggsfunktion 67  
 resultatattribut 41  
 <resultset>-element 41

## S

sammanfattning av steg för  
 XSLT-publicering 27  
 åtkomst av XML-data 35  
 sammansatt frågekommando för sökning 95  
 sammansatt frågeparameter för sökning 100  
 SAT  
 se Admin Console 20  
 -script, frågeparameter 102  
 -script.param, frågeparameter 103  
 -script.prefind, frågeparameter 103  
 -script.prefind.param, frågeparameter 103  
 -script.presort, frågeparameter 104  
 -script.presort.param, frågeparameter 104  
 -scriptnames, frågekommando 96  
 send\_email() extension, funktioner 68  
 session\_encode\_url() extension, funktion 66  
 session\_exists() extension, funktion 67  
 sessionstilläggsfunktioner, använda i  
 formatmallar 66, 67  
 set\_cookie(), tilläggsfunktion 71  
 set\_header(), tilläggsfunktion 70  
 set\_session\_object(), tilläggsfunktion 67  
 set\_session\_timeout(), tilläggsfunktion 67  
 set\_status\_code(), tilläggsfunktion 70  
 Shift\_JIS, kodning 59  
 sidhuvudfunktioner, använda 70  
 Site Assistant  
 beskrivning 16  
 Site Assistant. *Se* XSLT Site Assistant  
 skapa en ny post 95  
 skapa en statisk sida 60  
 skicka information mellan formatmallar 61  
 -skip, frågeparameter 104  
 skydda publicerade databaser 20  
 -sortfield, frågeparameter 105

- sortorder, frågeparameter 105
- SSL-kryptering 20
- statisk publicering, definition 11
- statiskt definierade frågesträngar i XSLT-formatmallar 58
- strängar
  - använda tilläggfunktioner för stränghantering 72
  - jämföra via reguljära Perl-uttryck 73
- stylehref, frågeparameter 106
- styletype, frågeparameter 106
- SVG, omvandla XML-data till 33
- syntax för XML, beskrivning 38
- syntax rekommenderad för XSLT 56
- Syntaxen FMPXMLLAYOUT 33, 44–46
  - jämfört med andra syntaxer 38
- Syntaxen FMPXMLRESULT 33, 43–44
  - jämfört med andra syntaxer 38
- säkerhet
  - begränsa åtkomst från IP-adresser 20
  - dokumentation 13
  - konton och lösenord 20
  - riktlinjer för att skydda publicerade databaser 20
  - statiskt definierade frågesträngar, använda 58

## T

- tekniktest 31
- testa
  - webbplatser 84
  - XML-utdata 85
- testa PHP-publicering 31
- textkodning
  - använda tilläggfunktioner för stränghantering 72
  - encoding, frågeparameter 59, 97
  - för XSLT-förfrågningar 59
  - förfrågan och utdatasidor, standardinställning för 59
  - genererade XML-data 38
  - kodningsinställningar 59
  - URL-adresser 37, 66
  - ursprungliga standardinställningar 59
- tid, tilläggfunktioner, använda 75
- tidpunkt, attribut 41
- tidsformatsträngar 76
- tillgängliga databaslayouter 95
- tillgängliga manus 96
- token, frågeparameter 61, 106
- typtribut 41

## U

- Unicode-tecken 47
- url\_decode(), tilläggfunktion 72
- url\_encode(), tilläggfunktion 72
- URL-syntax för
  - containerobjekt i XML-lösningar 37
  - containerobjekt i XSLT-lösningar 55
  - XML-förfrågningar 36
  - XSLT-formatmallar 54
- URL-textkodning 37
- US-ASCII, kodning 59
- User-Agent-huvud, kontrollera 61
- utdatasidor
  - kodning, ange 60
  - ursprunglig standardinställning för kodning 59
  - utdatametod, ange 60
  - <xsl:output>, element 60
- UTF-8 (Unicode Transformation 8 Bit)
  - format 37, 47
  - kodningsinställningar 59
- utforma data med formatmallar 25
- utlösare 23
- utvecklingsläge, Web Publishing Engine 114
- utökad behörighet för anpassad webbpublicering 19

## V

- vCards, utforma om XML-data till 33
- verifierad grundläggande URI-parameter 63
- verifiering av webbanvändare 19
  - grundlägg. 63
  - lösenordet 64
- verktyg för XSLT, beskrivning 16, 28
- view, frågekommando 96
- värden, söka efter i kryssrutor 73
- Växla layout för ett XML-svar 49

**W**

## Web Publishing Engine

- Admin Console 35
- Administrationskonsol 27
- behandling av förfrågan 12
- beskrivning 12
- felkoder 107
- fördelar med 15
- generera XML-data 34
- Produktionsläge 115
- programlogg 86
- skapa sidor från XSLT-formatmallar 26
- skapa XML-dokument 35
- Utvecklingsläge 114

web\_server\_module\_log.txt, loggfil 87

## webbanvändare

- använda containerfältdata 21
- krav för åtkomst till anpassade  
webbpubliceringslösningar 17
- åtkomst till skyddad databas 19, 63, 64

## webbläsare

- roll i XML-förfrågningar 34
- roll i XSLT-CWP-förfrågningar 26
- ta emot utdata 12

webbmapp, kopiera containerfältobjekt 21

## webbplatser

- FileMakers supportsidor 9
- skapa med Web Publishing Engine 15
- testa 84
- övervaka 86

## Webbpubliceringskärna

- bild på 26, 34
- interna åtkomstloggar 87

## webbserver

- loggfiler 86
- MIME-typer, stöd för 21
- roll i XML-förfrågningar 34
- roll i XSLT-CWP-förfrågningar 26

wpc\_access\_log.txt, fil 87

**X**

## XML

- aktivera i databas 19
- använda formatmallar på klientsidan 50
- beskrivning 33
- dokumenttypdefinitioner (DTDer) 39, 43
- efterfråga data 35
- felsöka åtkomst till XML-dokument 51
- filtrera data 33
- fmresultset, syntax 40

<datasource>-element 40

<fältdefinition>-element 41

<metadata>-element 41

<relatedset-definition>-element 41

<resultset>-element 41

frågestr. 47, 89

generera XML-data från förfrågan 34

kodad med UTF-8-format 38, 47

namnuttryck för 38

ordningsföljd för frågebearbetning 49

sammanfattning av steg för åtkomst av XML-  
data 35

syntax, beskrivning 38

Syntaxen FMPXMLLAYOUT 44

Syntaxen FMPXMLRESULT 43

tolkar 35, 47

URL-textkodning 37

XML 1.0-specifikation 33

XML-formatmall, bearbetningsinstruktion för 50

XML anpassad webbpublicering 13

XML-förfrågning

ange layout 49

XML-svar

växla mellan layouter 49

Xpath-uttryck 61

<xsl:stylesheet>, element 57, 61, 62, 85

<xsl:message>, element 78

<xsl:output>, element 60

<xsl:param name="authenticated-xml-base-uri"/>,  
parameter 63

<xsl:param name="client-ip"/>, parameter 62

<xsl:param name="client-password"/>,  
parameter 62

<xsl:param name="client-user-name"/>,  
parameter 62

<xsl:param name="request-query"/>, parameter 62

<xsl:param name="xml-base-uri"/>, parameter 63

<xsl:param>, element 61

<xsl:template>, element 62, 63, 85

<xsl:variable>, element 64

## XSLT

aktivera i databas 19

använda formatmallar på webbplatser och i

- program 30
- beskrivning 25
- cookies, tilläggsfunktioner 71
- datum- och tidsformatsträngar 76
- datum, tid och dag, tilläggsfunktioner 75
- e-postmeddelanden, skicka 68
- exempel på formatmallar 25
- felstatus i tilläggsfunktioner, kontrollera 78
- felsöka formatmallar 31
- FileMaker-specifika XSLT-parametrar 61
- FileMaker-tilläggsfunktioner 61
- formatmallar på serversidan 25, 53
- frågesträngar för 56
- frågesträngar, referens 90
- grammar, parameter 56
- innehållsbuffering, använda 65
- JavaScript, tillägg 78
- jämföra strängar via reguljära Perl-uttryck 73
- kryssrutor, söka efter värden i 73
- layoutinformation, använda 65
- namnuttryck för 57
- sammanfattning av steg för publicering 27
- sidhuvudfunktioner, använda 70
- skapa sidor från XSLT-formatmallar 26
- stränghantering, tilläggsfunktioner 72
- utveckla formatmallar 53
- XSLT 1.0-specifikation 25
- XSLT Site Assistant, använda 28
- XSLT-CWP-förfrågan 26
- xslt-template-files, mapp 27, 30, 64
- XSLT anpassad webbpublicering 13
- XSLT Site Assistant
  - använda 29
  - beskrivning 28
  - förbereda för användning 28
  - genererade formatmallar, beskrivning 29
  - starta 29
- <?xslt-cwp-buffer buffer-content="true"?>, bearbetningsinstruktion 66
- <?xslt-cwp-query?>, bearbetningsinstruktion 53, 58
- XSLT-formatmallar på serversidan 25, 53
- XSLT-formatmallar, bearbeta 96
- xslt-template-files, mapp 27, 30, 64

## Å

åtkomstloggfiler för webbserver, beskrivning 86

## Ä

Ändra lösenord, manussteg 20

## Ö

- översikt
  - Anpassad webbpublicering 11
- översikt över steg för
  - XSLT-publicering 27
  - åtkomst av XML-data 35
- övervaka webbplatser 86