Web Services in FileMaker

WEB004

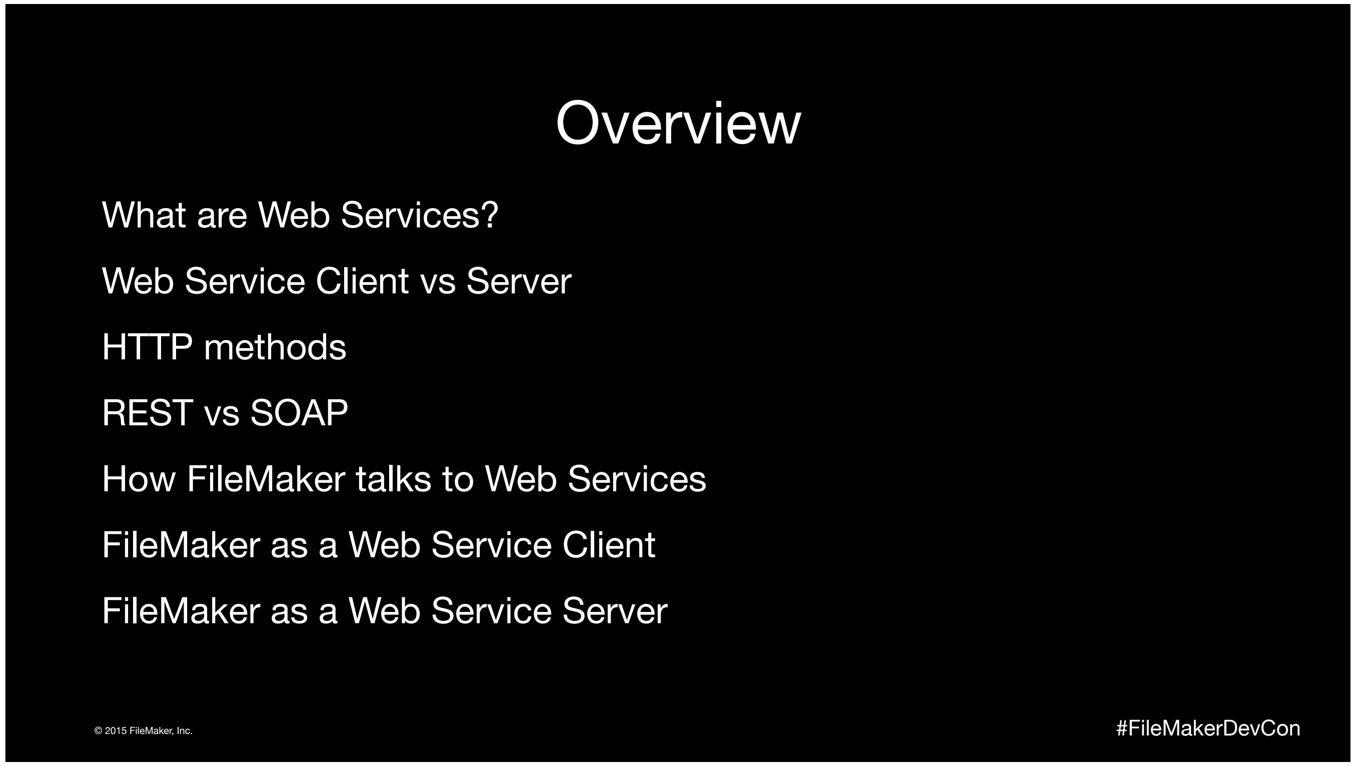
Salvatore Colangelo
Goya Pty Ltd

FILEMAKER DEVCON
EXPERIENCE THE POSSIBILITIES
20TH ANNIVERSARY | LAS VEGAS, NEVADA | JULY 20-23, 2015

Who am I?

Salvatore Colangelo

- grew up in Italy
- became a FileMaker developer in France
- live in Australia working for Goya
- always been more interested in the 'data model' than the graphics
- tried to pass a FM certification in italian - failed
- certified in all recent versions
- take frisbee seriously



Overview

What are Web Services?

Web Service Client vs Server

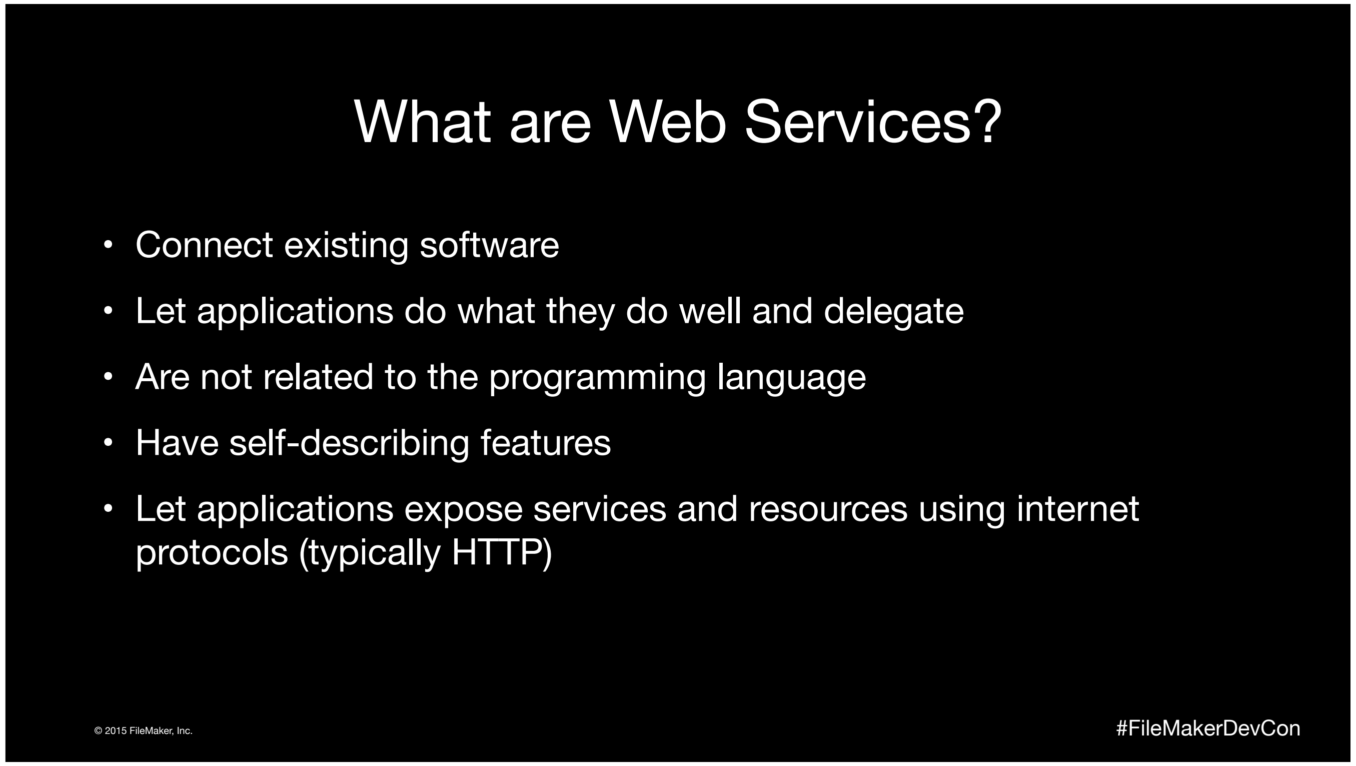
HTTP methods

REST vs SOAP

How FileMaker talks to Web Services

FileMaker as a Web Service Client

FileMaker as a Web Service Server



What are Web Services?

- Connect existing software
- Let applications do what they do well and delegate
- Are not related to the programming language
- Have self-describing features
- Let applications expose services and resources using internet protocols (typically HTTP)

they’re a major part of the architecture of the web as it stands today.
if we want our filemaker applications to be part of this universe of web applications is it better to work with the architecture or against?

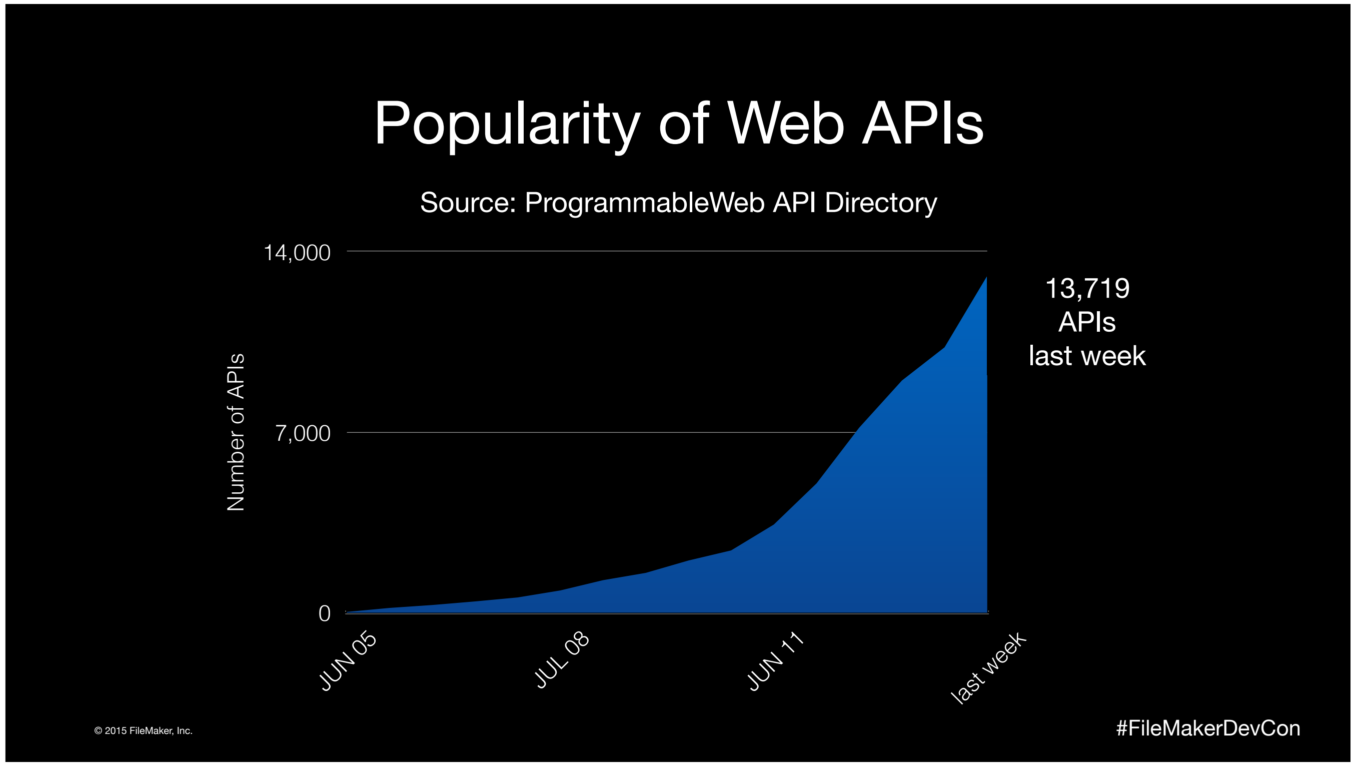
the default standard for inter application communication

if our filemaker solution is the central point of our client data flow it will be a lot more valuable to them instead than being just an application to do something.

web services can help us making this scenario as easy as possible for us, using our filemaker skills and leaving other specialised tasks to someone else.
Who here is an app developer? a great web developer?

Anything that could be considered an “application”, like your FileMaker solution, someone else’s FileMaker solution, other applications on your computer, websites, web apps etc. stores data and needs to send that data somewhere else, then this will likely be done via Web Services these days.

The thing that makes this worth learning is that if you build your FileMaker solution as a silo it’s value is always going to be limited. If you build it as a central hub that lets other applications talk to it, and that it talks to, then its value becomes much higher for your client, with very little extra work on your behalf.

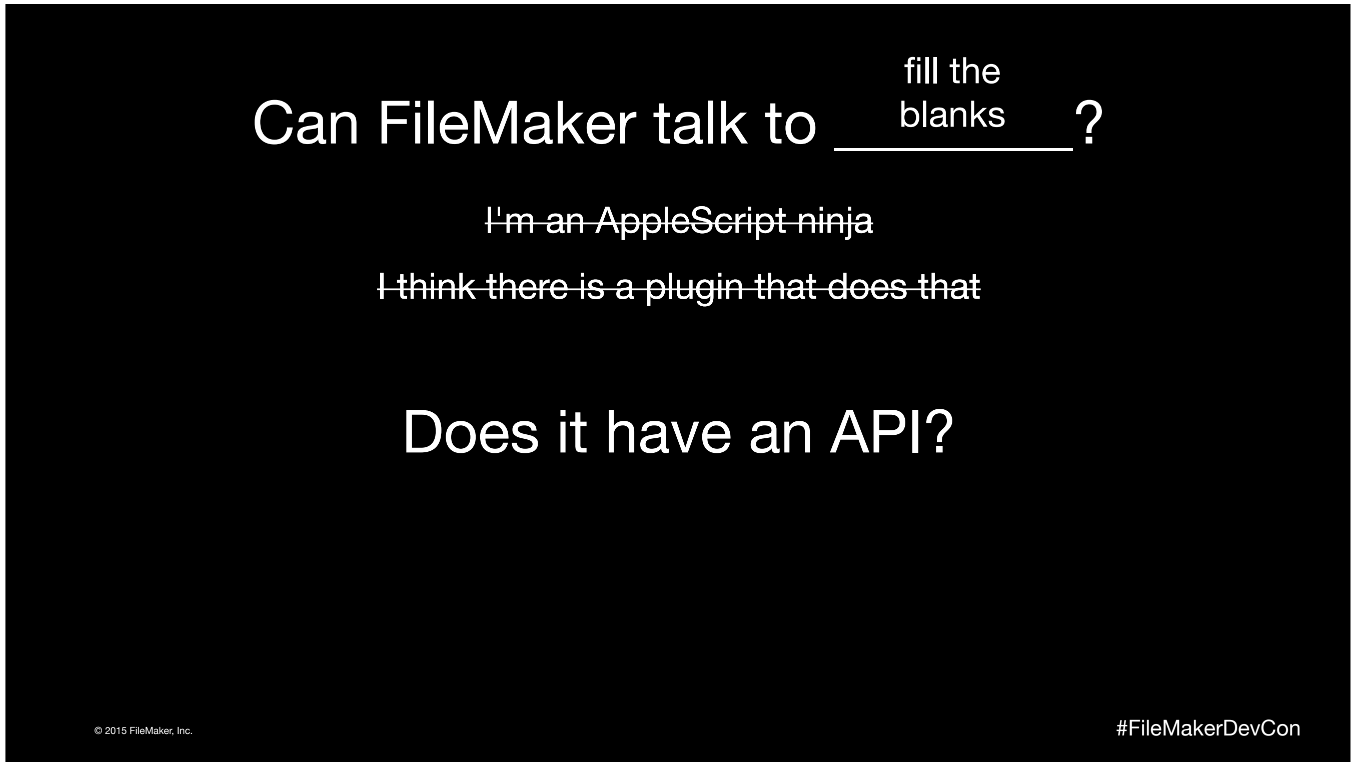


Popularity of Web APIs

Source: ProgrammableWeb API Directory

#FileMakerDevCon

so how important APIs really are?



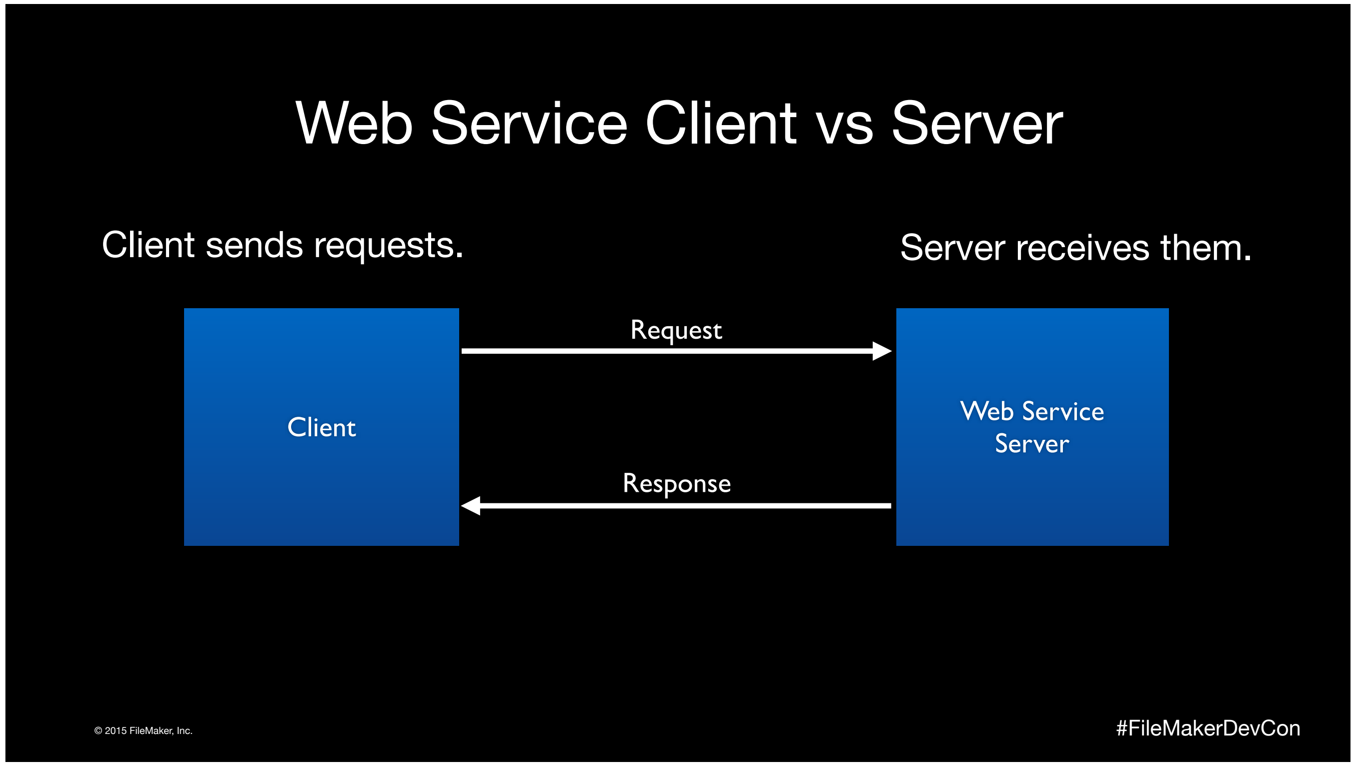
Can FileMaker talk to fill the blanks ?

~~I'm an AppleScript ninja~~

~~I think there is a plugin that does that~~

Does it have an API?

Ever had this conversation? I've found myself answering more and more often "there's an API for that" which makes my life much easier



Web Service Client vs Server

Client sends requests.

Server receives them.

Client

Request

Response

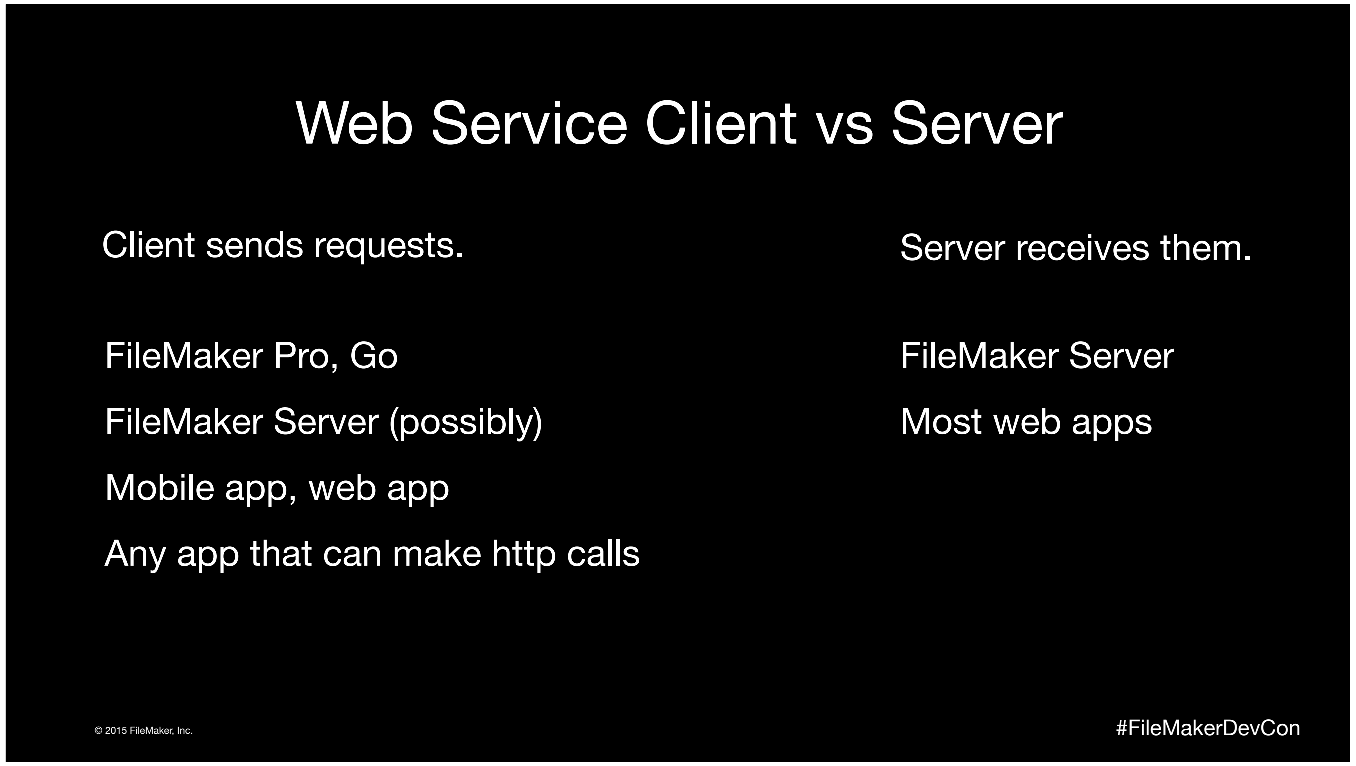
Web Service Server

#FileMakerDevCon

the client makes a request, the server gives you a response. very simple

The reason it’s this simple is not my poor drawing skills, it’s that it’s based on HTTP, the same as your browser.

like you web browser for websites, but when we talk about web services the client is a system and not someone typing on keyboard.



Web Service Client vs Server

Client sends requests.

FileMaker Pro, Go

FileMaker Server (possibly)

Mobile app, web app

Any app that can make http calls

Server receives them.

FileMaker Server

Most web apps

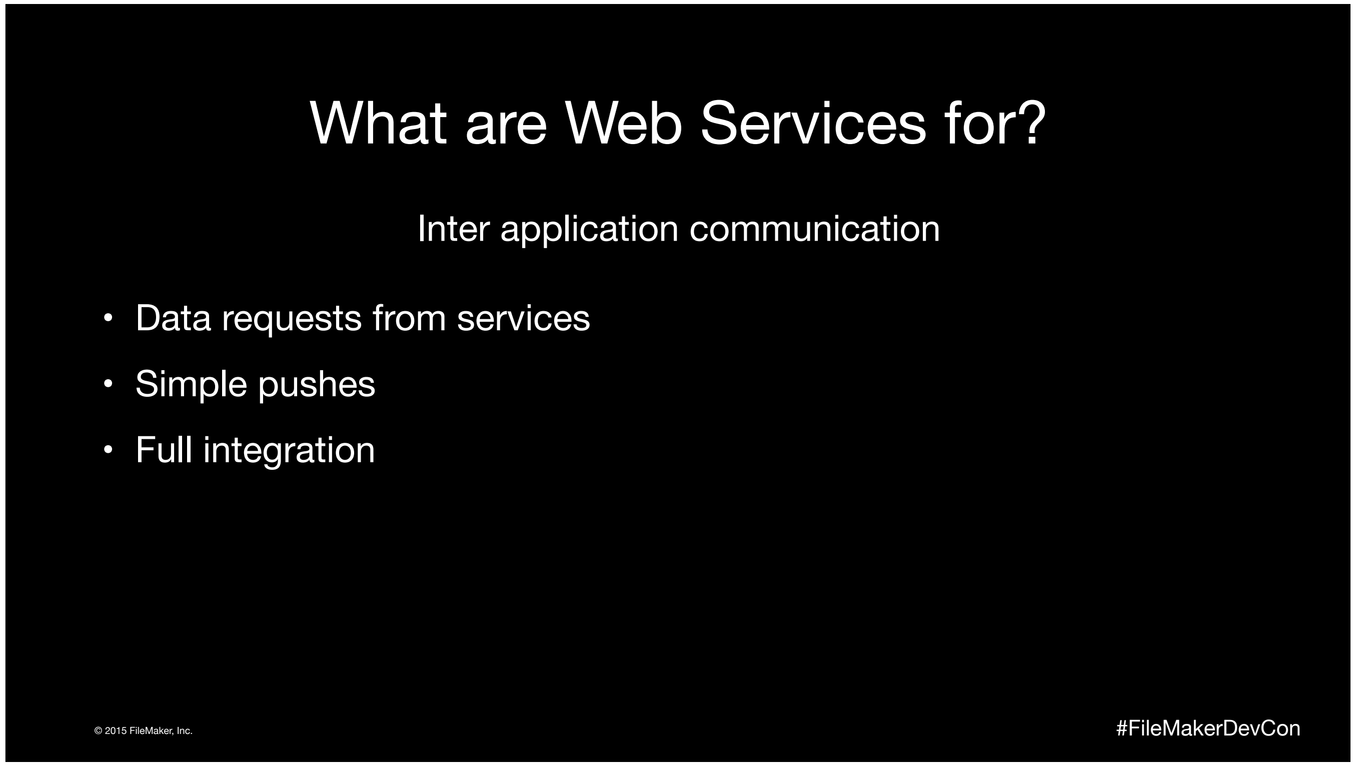
#FileMakerDevCon

So FMP can be a client, it can send requests in lots of ways. FMS can also be a client, because it can also do scripting, PSOS, WebDirect and Server side schedules.

FMP can’t answer HTTP requests, and although there are some plugins that let you create an HTTP server in Pro, it’s not practical because FMP is intermittent – only open when the client app is running.

Other apps can be clients, making requests of each other, or of FMS, and other apps can be servers, waiting patiently for requests.

Mostly if you’re running a FileMaker based Web Services Server, then it will be in FMS, and there’s some built in tools (Web Publishing) to do that, and some third party, free and paid products as well.



What are Web Services for?

Inter application communication

- Data requests from services
- Simple pushes
- Full integration

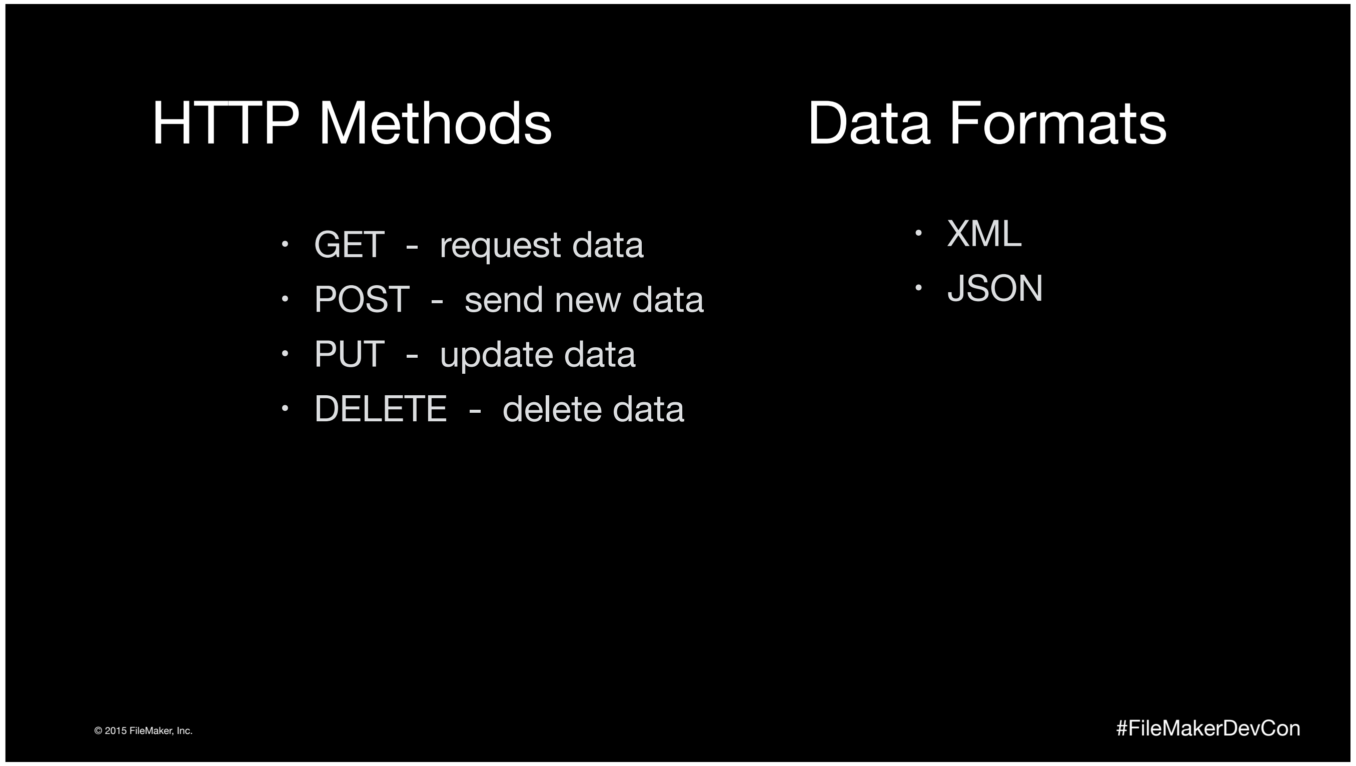
#FileMakerDevCon

All of the things that are web services can be considered different ideas around the phrase “inter application communication”. This is a great way of describing it, one app talking to another, but we often have some fixed ideas about what WS should be. We can try to group the type of action in three types.

Request data from services : you send a request, so you request something from the USPS postcode lookup url, and the parameter is the post code you want to lookup. The USPS’s WS responds with the full details of the name of that location. Or vice versa, where you put in a name, and it gives you all the possible matches and their post codes.

Simple pushes : this is where you send off a quick set of details that the server does some work, based on your info. So we tell it a phone number and a message, and the server sends an SMS. Or you send an inventory system the details of a new purchase, and the purchasers details and it organise the item to be packed and shipped.

Full integrations : this is a bit like a sync system, but on a section of each at a time. Typical general example would be a project management system that does invoicing with an accounts system. The PM system creates invoices and contacts and sends them to accounts. Accounts updates payment status for the invoice, and keeps track of changes to contacts. So invoices are created on one system, but updated (payment status) on the other. Contacts are normally created on one, but could be created in either. Payments are always created in the accounts package, but are sent back to the PM system. So the two systems are syncronising these table between each system, but all of that syncing is done via WS. There’s no export import, and depending on the setup, it can be in near real time, but when you think about browsers sending request to web servers, so you don’t think about syncing your accounts package, but today, this is how it’s done.



HTTP Methods

- GET - request data
- POST - send new data
- PUT - update data
- DELETE - delete data

Data Formats

- XML
- JSON

#FileMakerDevCon

Web Services are based on HTTP. You send data between the two systems, but all the requests are done over HTTP. This does not mean HTML. HTML is a web browser display language, great for humans, not so good for computers.

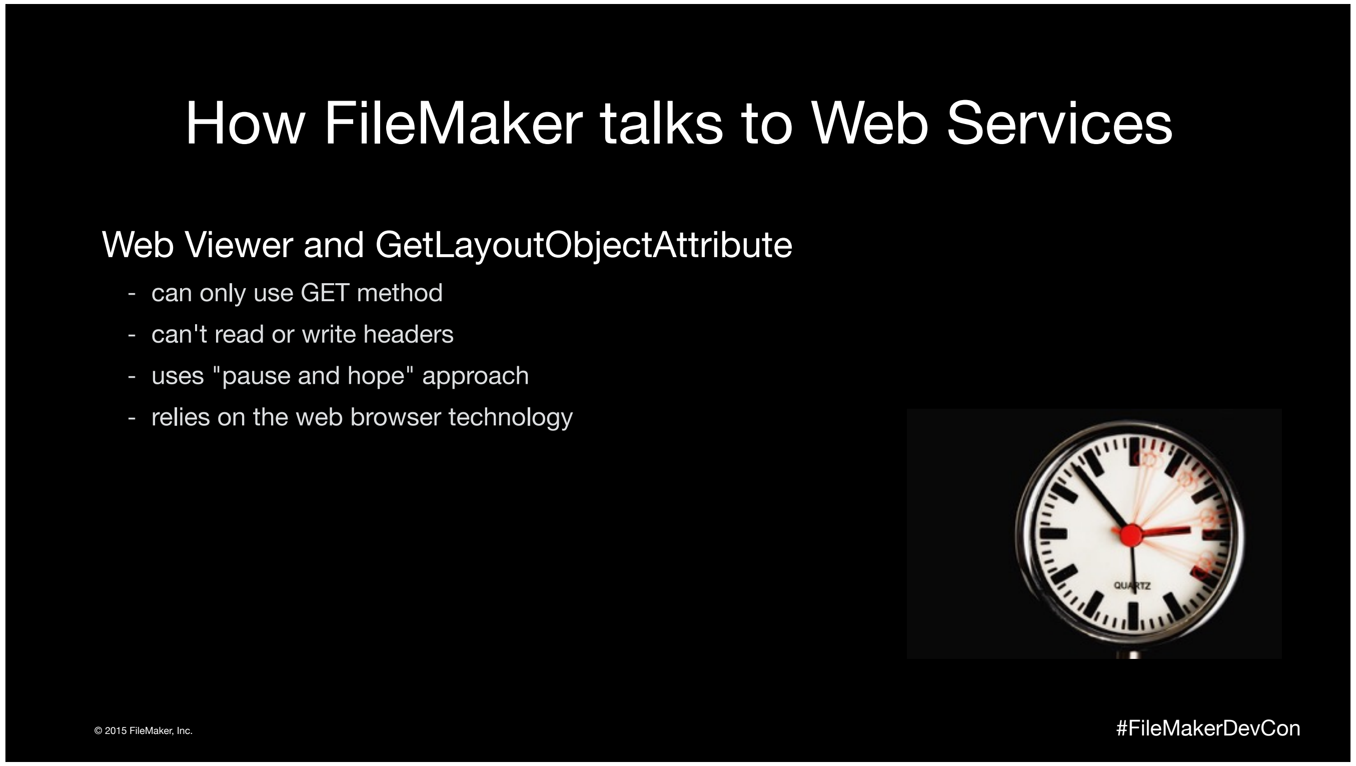
Methods : GET is what your web browser does. GET is the basic request of the web, and of web services. It’s exactly as it’s name suggests, GET the data at this url, with these optional parameters.

POST is the FileMaker equivalent of create. It means send some data to a server, for that server to keep and do something with.

PUT is the equivalent of a modify. Given an existing url (which represents a record somewhere in a database) change some or all of the data.

And delete should make sense.

Computers like languages like XML and JSON because they’re structured. In other words they’re easy for computers to generate and read.



How FileMaker talks to Web Services

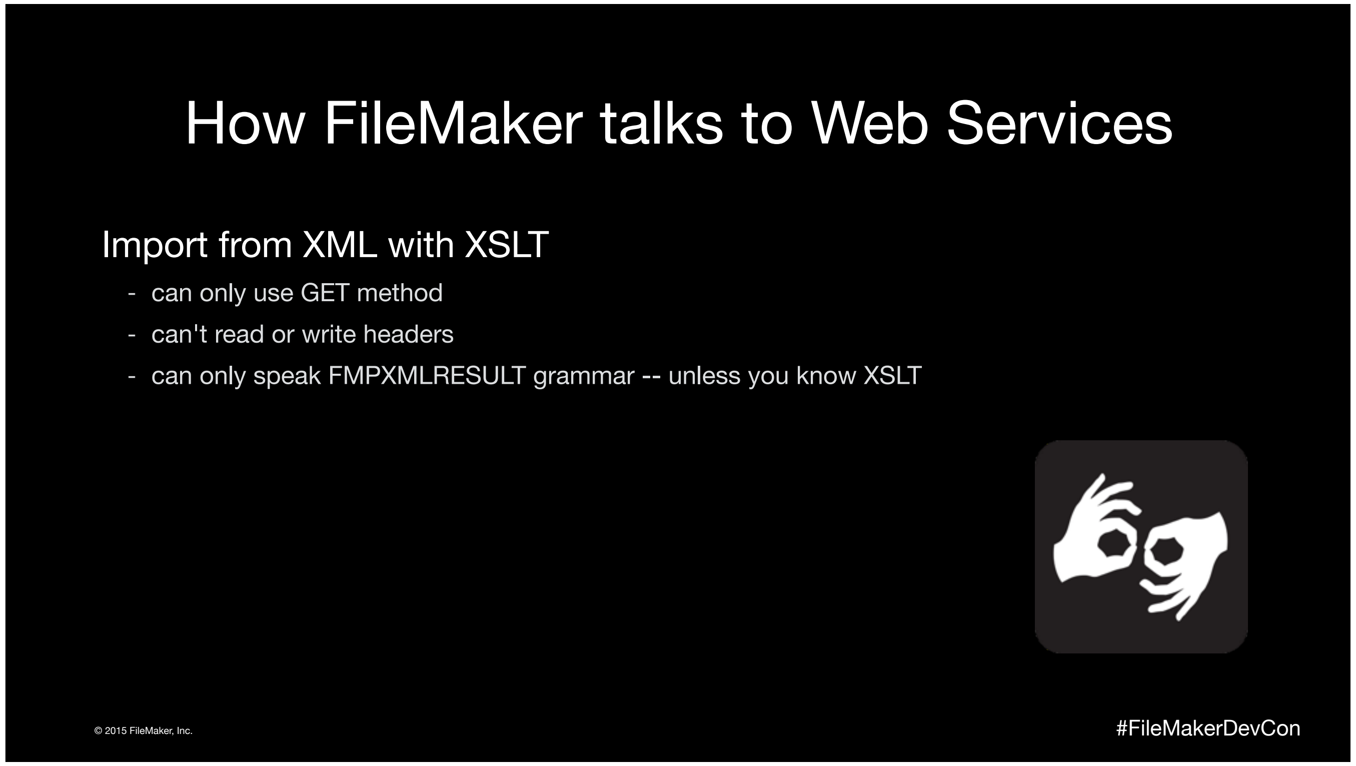
Web Viewer and GetLayoutObjectAttribute

- can only use GET method
- can't read or write headers
- uses "pause and hope" approach
- relies on the web browser technology

WebViewer and GetLayoutObjectAttribute to parse URL response... anyone used this since 2012?

You could extend the WebViewer, by writing custom javascript code, that you store on the device that does all the work and when complete, sets a script to call with the result.

So … in theory … you could write a whole SOAP or REST implementation in JavaScript that ran in a WebViewer.



How FileMaker talks to Web Services

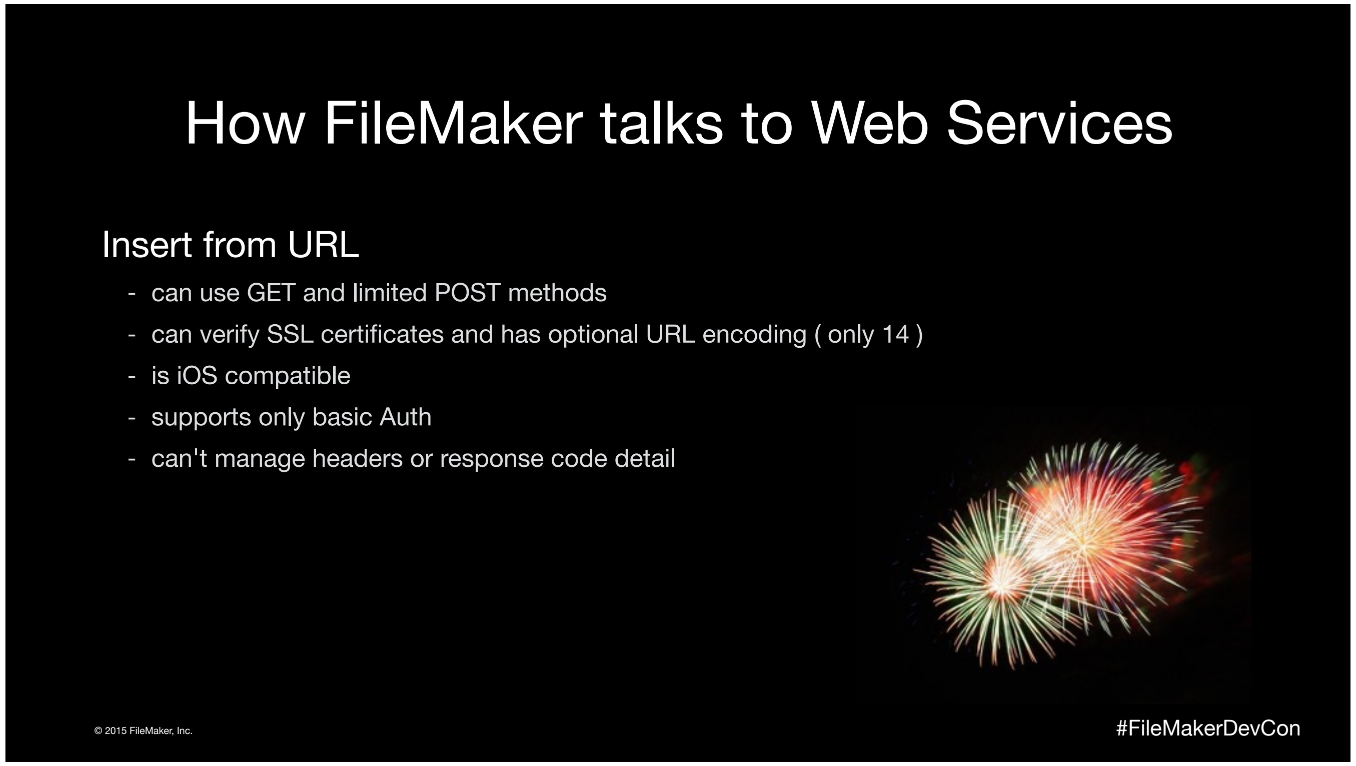
Import from XML with XSLT

- can only use GET method
- can't read or write headers
- can only speak FMPXMLRESULT grammar -- unless you know XSLT

FileMaker knows it’s own XML format. And nothing else in the world speaks FileMaker’s format. That’s actually intentional, that’s the idea of XML that each app has it’s own structure, and they can speak because we can use XSLT as a translation engine between each format.

I know a couple of guys who write XSLT in their spare time. How many there?

But for a site doing XML where you know the format and can do XSLT, then it’s certainly an option.



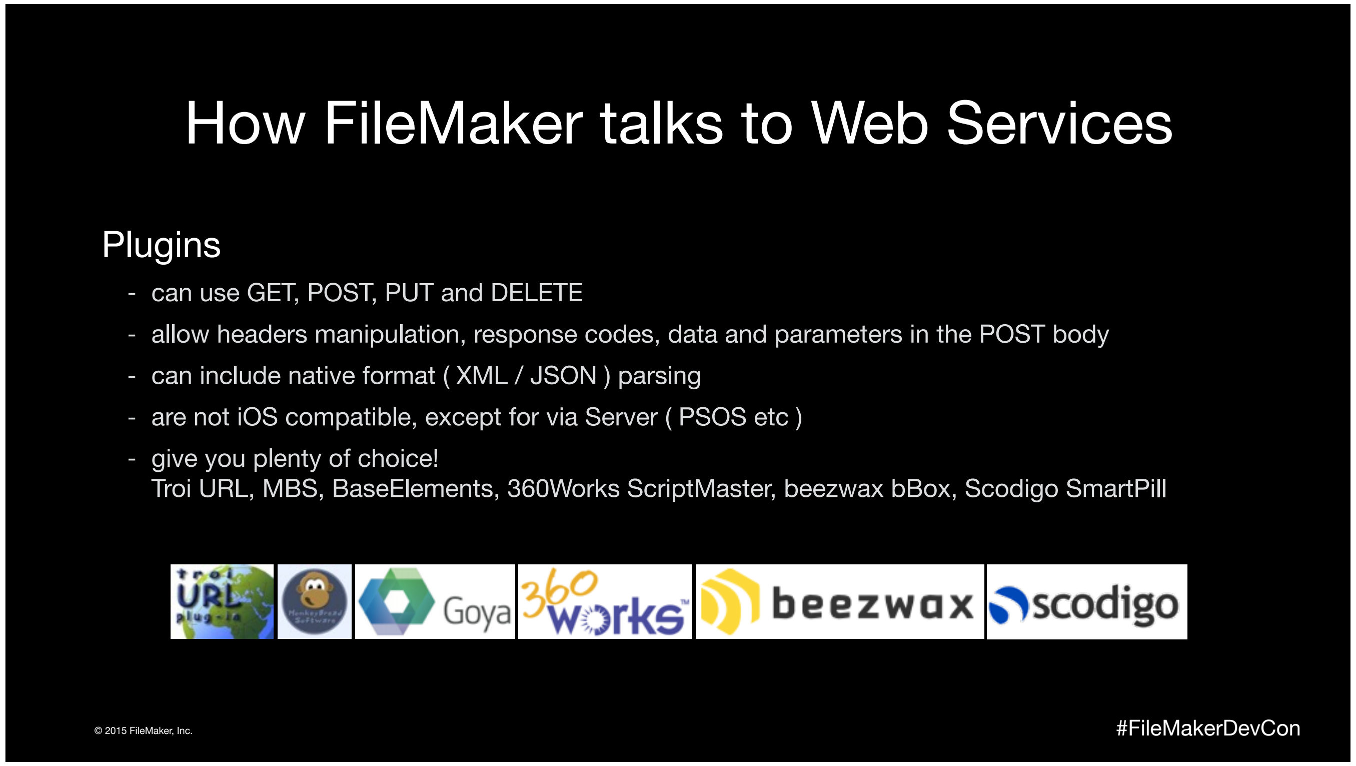
How FileMaker talks to Web Services

Insert from URL

- can use GET and limited POST methods
- can verify SSL certificates and has optional URL encoding (only 14)
- is iOS compatible
- supports only basic Auth
- can't manage headers or response code detail

The optional url encoding isn’t really optional. If you wanted to send an ampersand character in 13, well you can’t. It would try to encode it poorly and break your data.

The bigger issue is the limited part of the limited POST. The way the POST works is that you can’t have both parameters and data. FileMaker turns your parameters in the data.



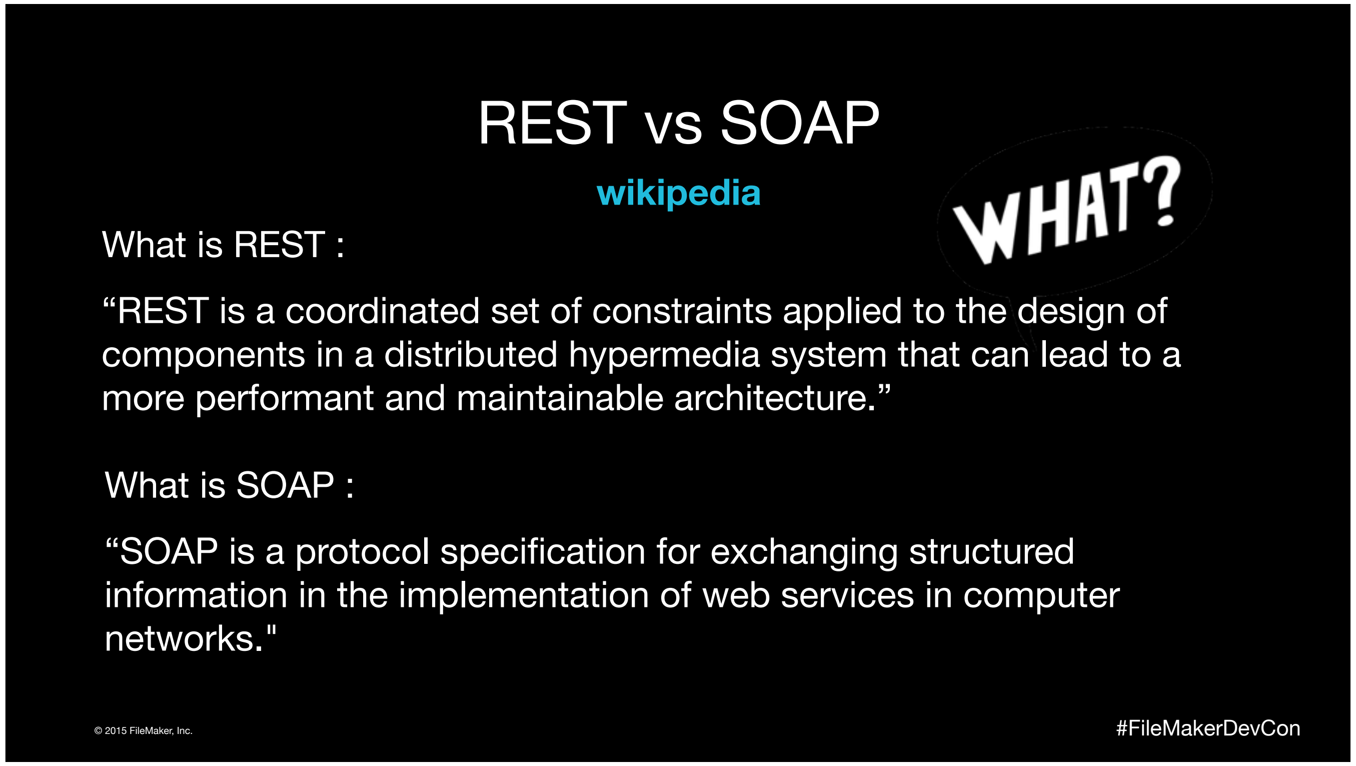
How FileMaker talks to Web Services

Plugins

- can use GET, POST, PUT and DELETE
- allow headers manipulation, response codes, data and parameters in the POST body
- can include native format (XML / JSON) parsing
- are not iOS compatible, except for via Server (PSOS etc)
- give you plenty of choice!
  Troi URL, MBS, BaseElements, 360Works ScriptMaster, beezwax bBox, Scodigo SmartPill

Plugins give you ultimate control, essentially you can do pretty much anything you want with a plugin.

But they'll never come to iOS.



REST vs SOAP

wikipedia

What is REST :

“REST is a coordinated set of constraints applied to the design of components in a distributed hypermedia system that can lead to a more performant and maintainable architecture.”

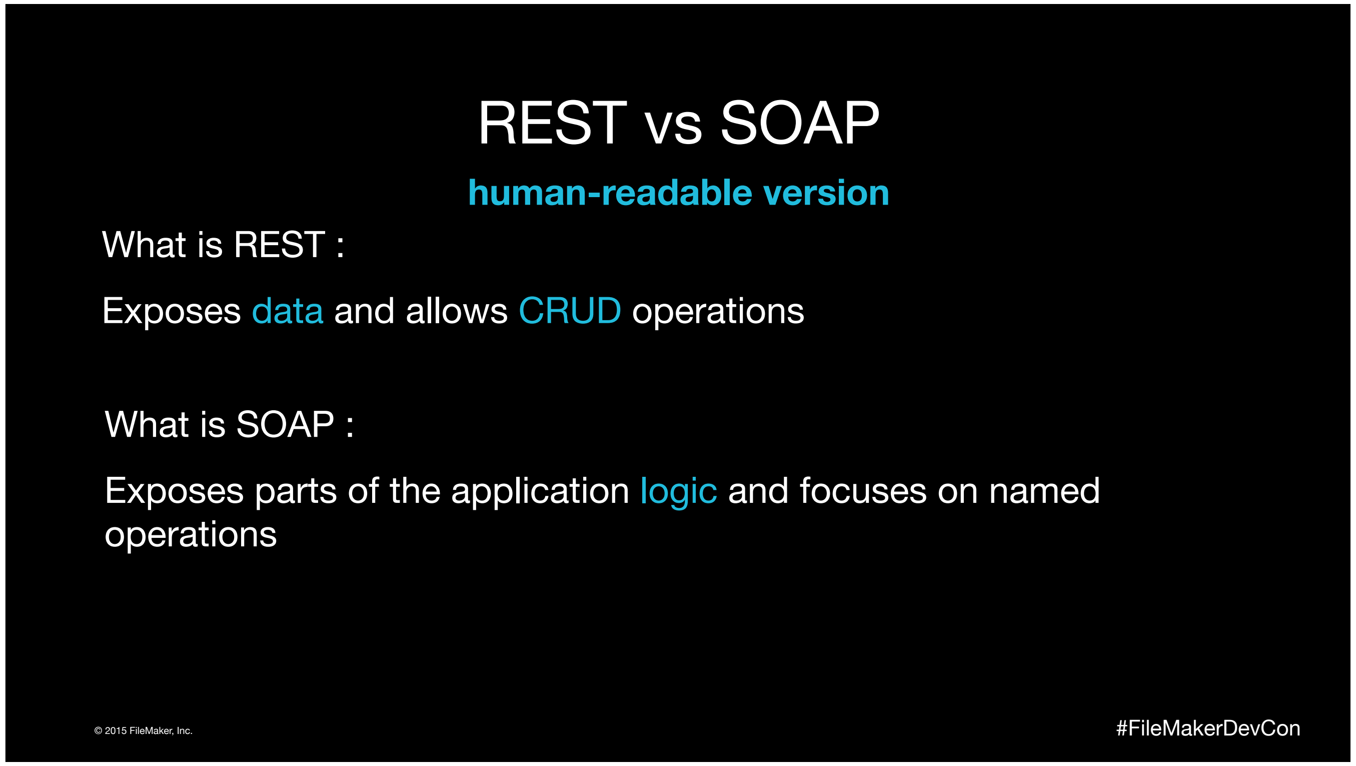
What is SOAP :

“SOAP is a protocol specification for exchanging structured information in the implementation of web services in computer networks."

#FileMakerDevCon

REST and SOAP are the two main standards for Web Services, and are designed to make it easier for different services to talk to each other in ways that each will understand.

Did you understand the REST definition? I didn't. REST is supposed to be the simple one, but I have no idea what that paragraph means



REST vs SOAP

human-readable version

What is REST :

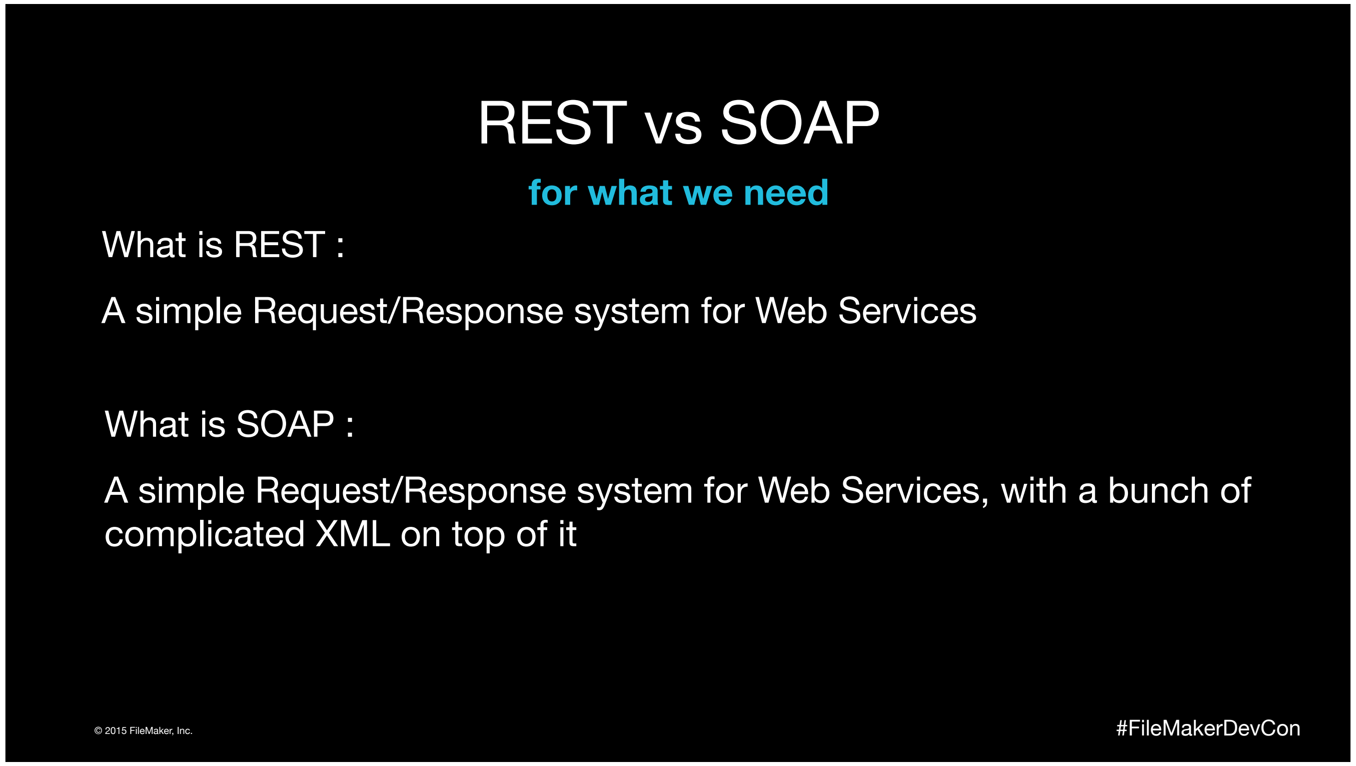
Exposes data and allows CRUD operations

What is SOAP :

Exposes parts of the application logic and focuses on named operations

Better?

What is CRUD? Create, Read, Update Delete.   Maps really well to :
Create is POST
Read is GET
Update is PUT
Delete is Delete.



REST vs SOAP

for what we need

What is REST :

A simple Request/Response system for Web Services

What is SOAP :

A simple Request/Response system for Web Services, with a bunch of complicated XML on top of it

#FileMakerDevCon

This is much easier I think. I much prefer working in REST systems. They’re much simpler, much quicker to pick up, and you don’t need anything fancy to get started.

Why anyone would choose SOAP?
SOAP has it’s place. It adds on a whole heap of structures and systems around the services, that are great … in theory. The idea is that the services document themselves. So any application that does SOAP can talk to any other application that does SOAP. You send a request, and ask for the WSDL (Web Services Definition Language). In theory that should tell you everything you need to know about the service and let you talk to it.

A few years back Vince Mennano from BeezWax created a FileMaker plugin that did exactly this. Point it at a WSDL, and it will automagically create a set of FileMaker functions corresponding to the services that the site offers.
So one request to configure the system, and you’re up and running.

It became such a nightmare of not quite right services, different implementations, variations on the spec, authorisation changes that the plugin is no longer developed or supported.

SOAP, great in theory, hard in reality.

REST vs SOAP example

what is the current price of Apple stocks?

SOAP

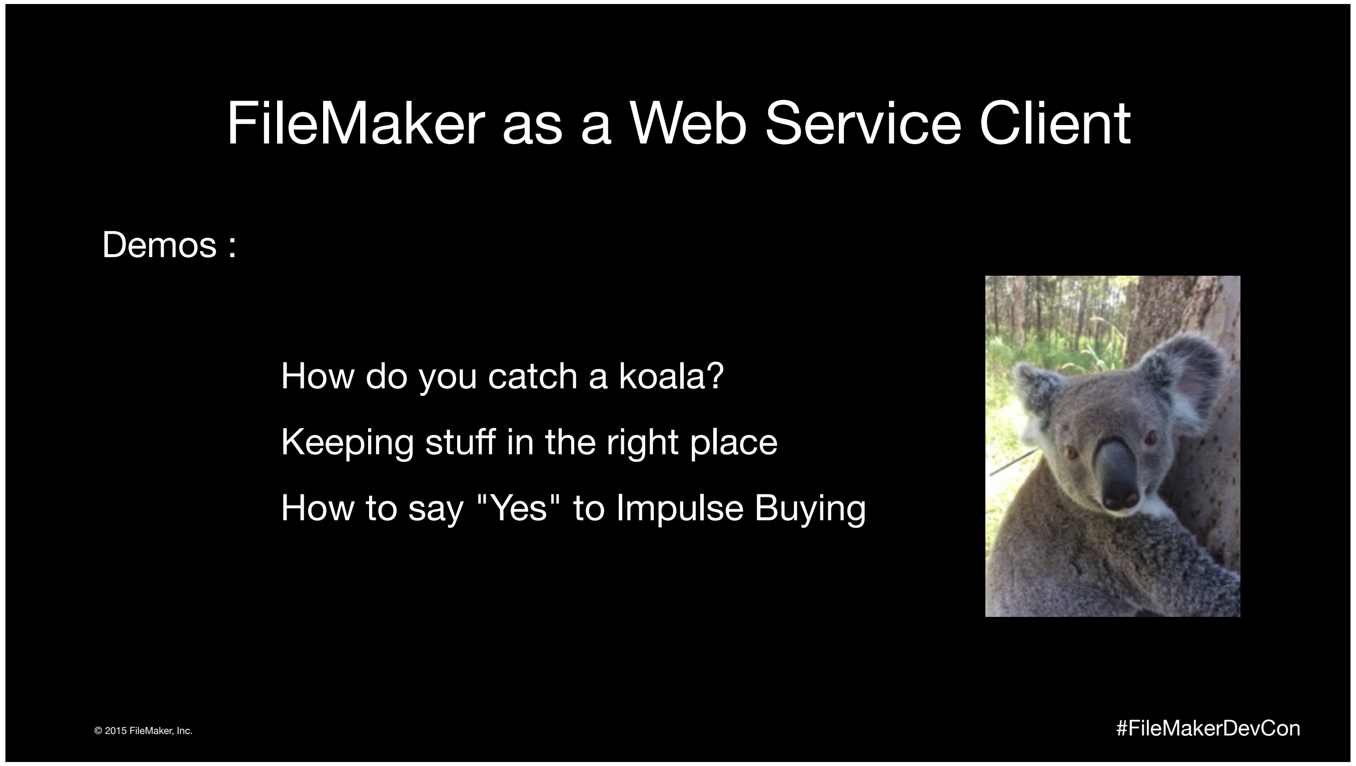
```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap=“http://www.w3.org/2001/12/
soap-envelope" soap:encodingStyle="http://www.w3.org/
2001/12/soap-encoding">

<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPrice>
    <m:StockName>AAPL</m:StockName>
  </m:GetStockPrice>
</soap:Body>

</soap:Envelope>
```

REST

http://www.example.org/stock/StockPrice/AAPL.json



FileMaker as a Web Service Client

Demos :

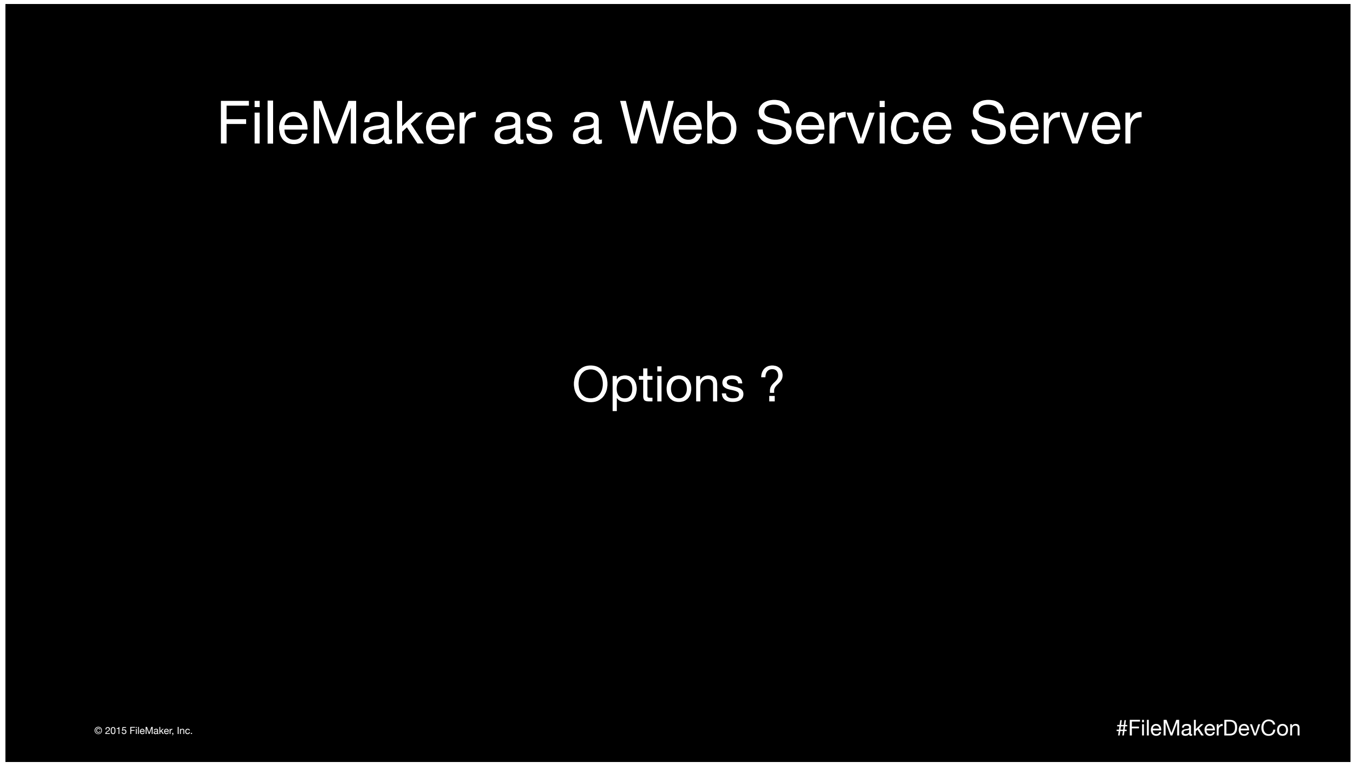
How do you catch a koala?

Keeping stuff in the right place

How to say "Yes" to Impulse Buying

GET with the WU RESTful API
POST with the Dropbox RESTful API
Shopping cart with the paypal API

Her name is Bubbles.



FileMaker as a Web Service Server

Options ?

#FileMakerDevCon

FileMaker has built in native XML Publishing, which is lots of ways is a Web Services option that we can use. On top of that we can write our own code in PHP and use that.

There’s some useful third party products.
360 Works Web Services Manager is a way of triggering a script using a SOAP architecture.

RESTfm is a way of doing RESTful Web Services in FileMaker.

FX.php is an alternative to the FileMaker native PHP API, and would probably have never been created, except they built it long before FileMaker had it’s own one. It’s stable, and continues to be updated.

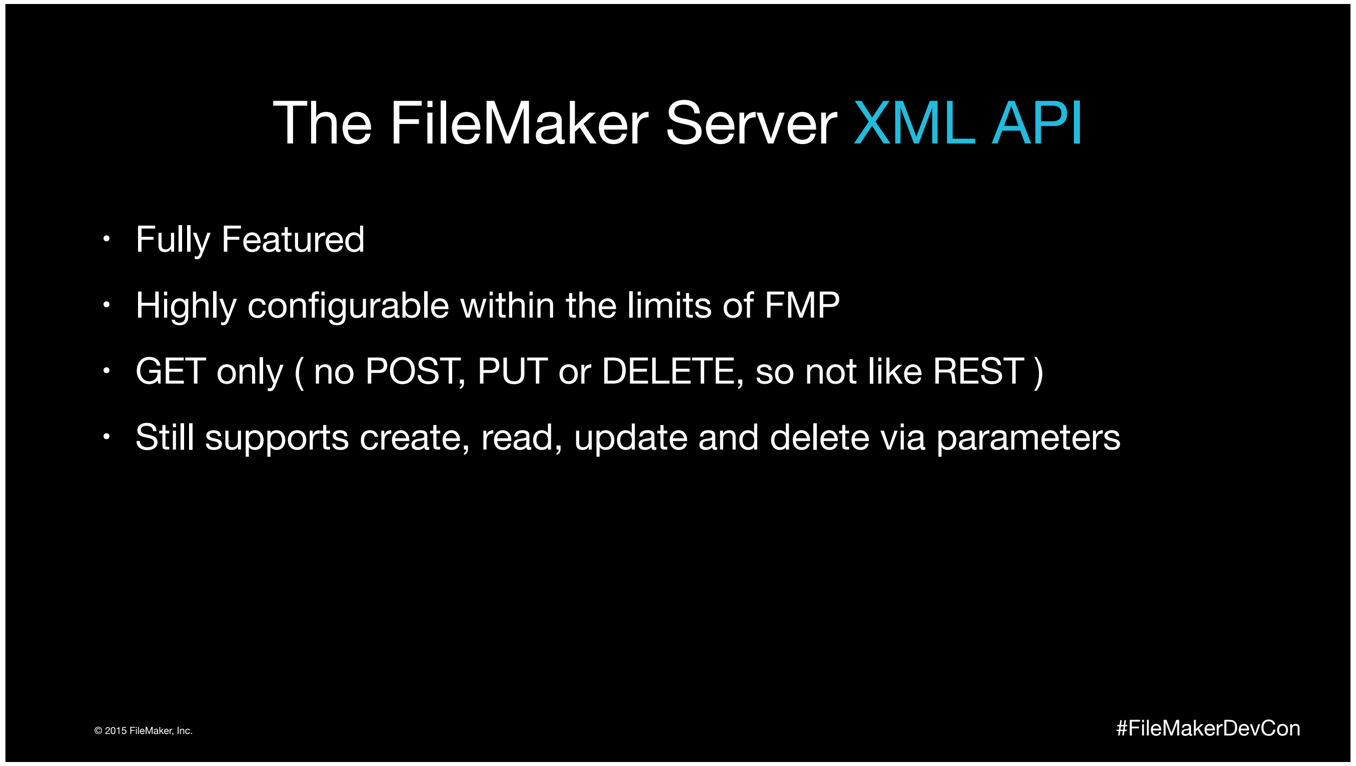
fmDotNet is a .NET dll that you can use to build desktop apps, ASP.net websites, web services and lots more.

fxphp and fmdotnet can be considered as extensions on the XML api that give you more options in terms of platform build with and are really beyond the scope of this session.

FileMaker as a Web Service Server

XML Publishing

FileMaker has built in native XML Publishing, which is lots of ways is a Web Services option that we can use. On top of that we can write our own code in PHP and use that.



The FileMaker Server XML API

- Fully Featured
- Highly configurable within the limits of FMP
- GET only (no POST, PUT or DELETE, so not like REST)
- Still supports create, read, update and delete via parameters

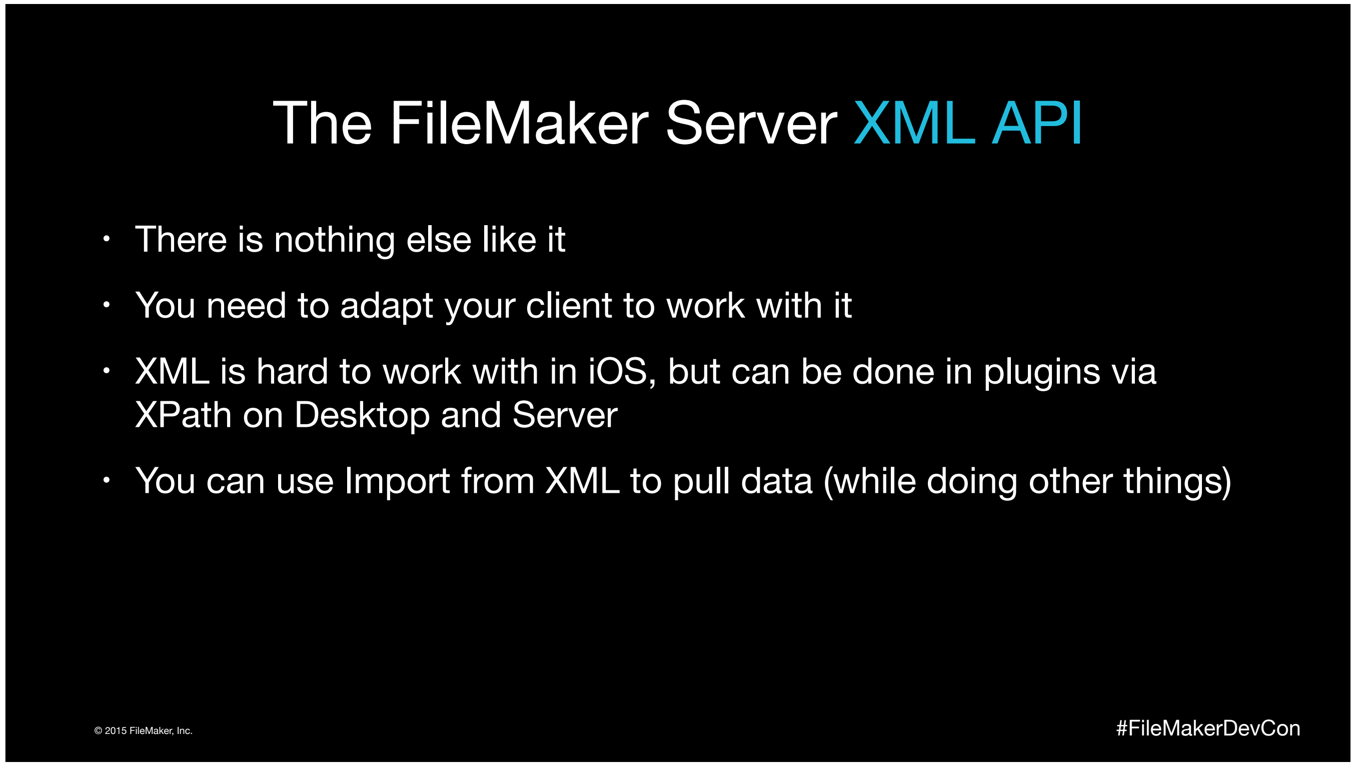
#FileMakerDevCon

The FileMaker Server XML API is very full featured. Almost anything you can do in FileMaker, you can do via XML. You could, for example :

Go to a layout.
Set some global fields.
Run a script
Perform a find or create a new record.
Perform another script.
Sort the set.
Perform another script.
Go to a different layout.
And then return the results of every field on the layout for the found set.

It handles container fields, portals, and related data.

Everything is built into the GET url string.



The FileMaker Server XML API

- There is nothing else like it
- You need to adapt your client to work with it
- XML is hard to work with in iOS, but can be done in plugins via XPath on Desktop and Server
- You can use Import from XML to pull data (while doing other things)

#FileMakerDevCon

No other Web Service I’ve come across works the way the XML API does, with all the data in parameters. It’s just a quirk of the FileMaker API.

It has it’s own XML format, and like I mentioned earlier, you need XML at each end and then XSLT in the middle to translate between formats. But FileMaker Server no longer does XSLT, so it makes it a bit harder because the web service client has to do all the work and us FileMaker developers can’t make it easier for anyone, it is what it is.

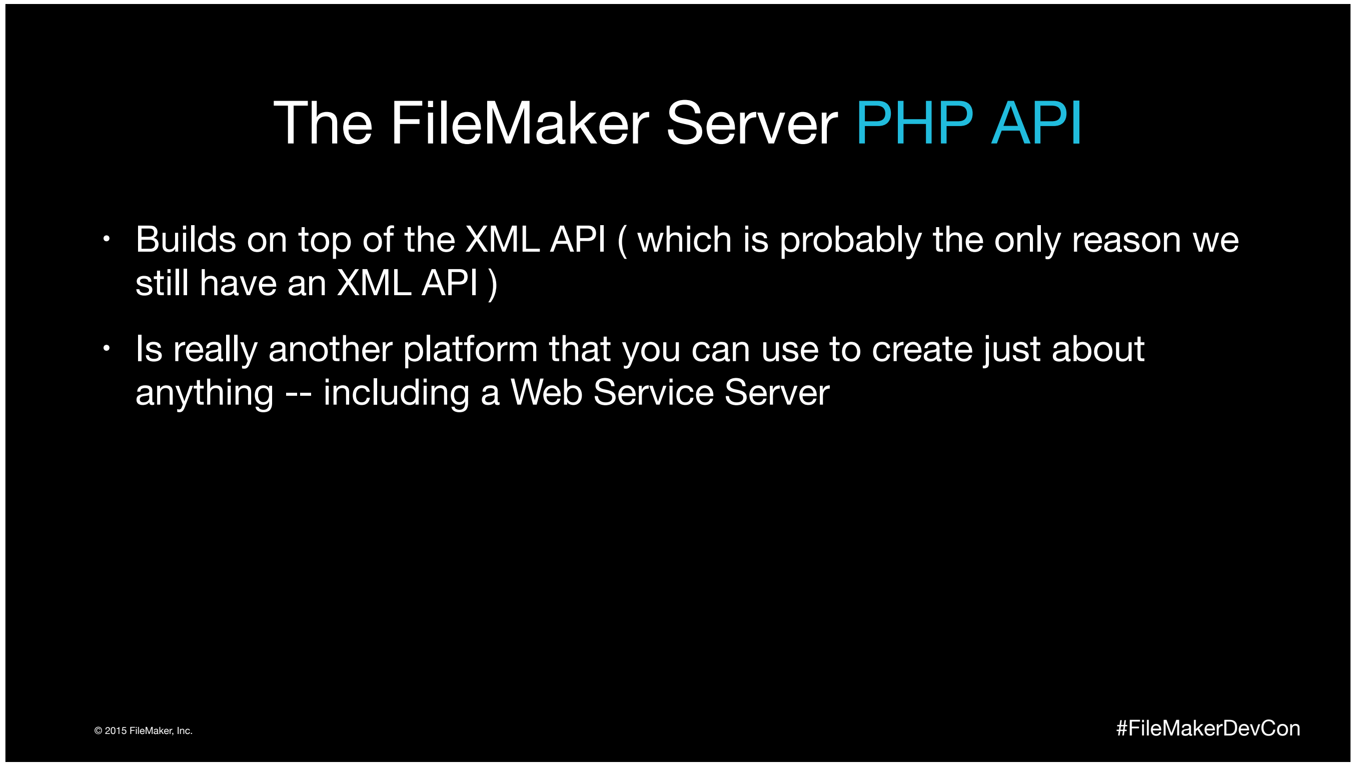
If you’re talking about a FileMaker client talking to a FileMaker Web service server, of course it understands it’s own XML, you just need to construct the url queries yourself, and then parse the output XML yourself.

Also you can import via HTTP from another FileMaker server.

So you can pull (GET) data from one file to another, but the import doesn’t have to be a found set, it might be a “results” record that is the end result of a long scripting operation that you’ve set it to run. You saw how much the API can do, you can tell the server to do whole lot of things and the “result” that you import is the result of the operation.

FileMaker as a Web Service Server

CWP via PHP

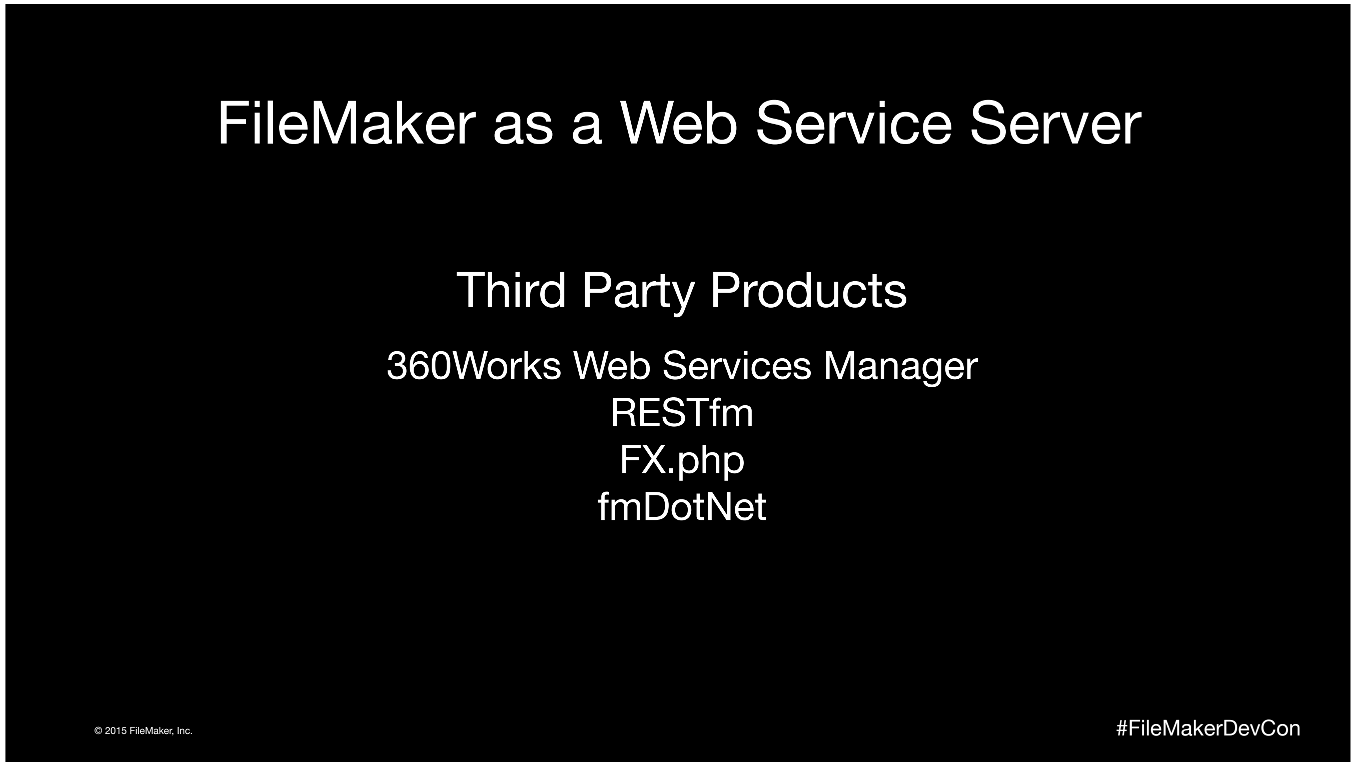


The FileMaker Server PHP API

- Builds on top of the XML API (which is probably the only reason we still have an XML API)
- Is really another platform that you can use to create just about anything -- including a Web Service Server

The PHP API is actually built on top of the XML API , mostly as a way to write custom web sites in PHP.

PHP was, and still is a good choice for a language to build FileMaker web applications with.



FileMaker as a Web Service Server

Third Party Products

360Works Web Services Manager
RESTfm
FX.php
fmDotNet

There’s some useful third party products.
360 Works Web Services Manager is a way of triggering a script using a SOAP architecture.

RESTfm is a way of doing RESTful Web Services in FileMaker.

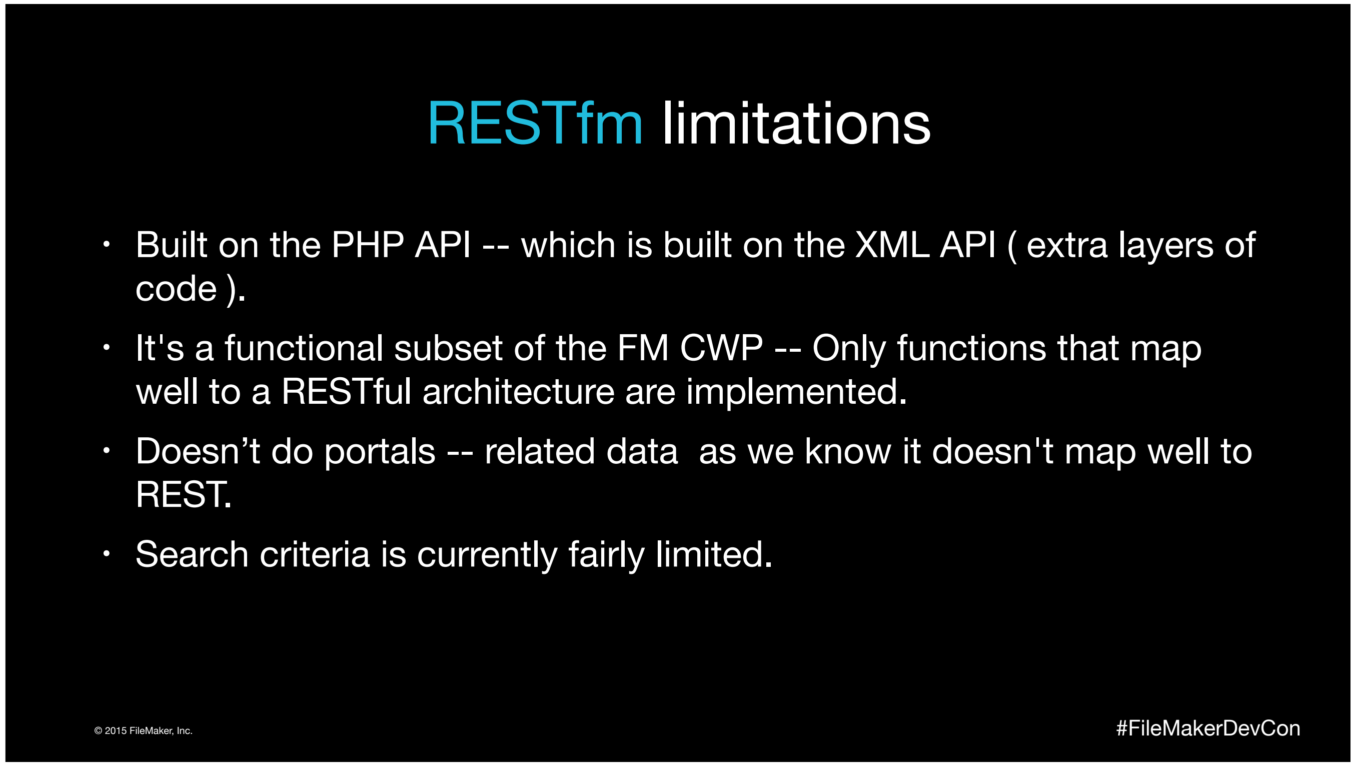
FX.php is an alternative to the FileMaker native PHP API, and would probably have never been created, except they built it long before FileMaker had it’s own one. It’s stable, and continues to be updated.

fmDotNet is a .NET dll that you can use to build desktop apps, ASP.net websites, web services and lots more.

fxphp and fmdotnet can be considered as extensions on the XML api that give you more options in terms of platform build with and are really beyond the scope of this session.

FileMaker as a Web Service Server

RESTfm



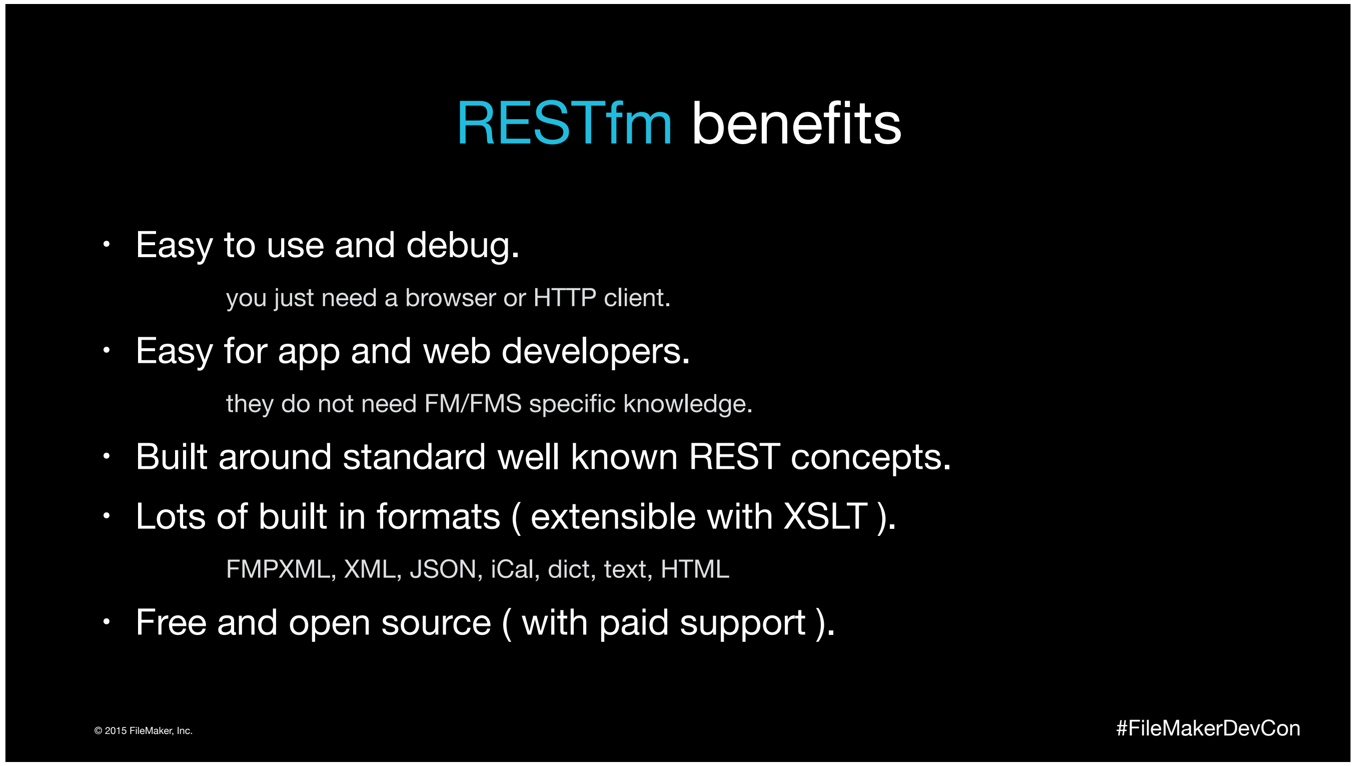
RESTfm limitations

- Built on the PHP API -- which is built on the XML API (extra layers of code).
- It's a functional subset of the FM CWP -- Only functions that map well to a RESTful architecture are implemented.
- Doesn’t do portals -- related data as we know it doesn't map well to REST.
- Search criteria is currently fairly limited.

Conflict of interest alert here: RESTfm is our product.

So it’s not as full features in terms of FileMaker functionality as the XML API, but it adds other benefits.

It doesn’t do related records, but you can do queries on different tables to get the same affect.



RESTfm benefits

- Easy to use and debug.
  - you just need a browser or HTTP client.
- Easy for app and web developers.
  - they do not need FM/FMS specific knowledge.
- Built around standard well known REST concepts.
- Lots of built in formats (extensible with XSLT).
  - FMPXML, XML, JSON, iCal, dict, text, HTML
- Free and open source (with paid support).

Conflict of interest alert here: RESTfm is our product.

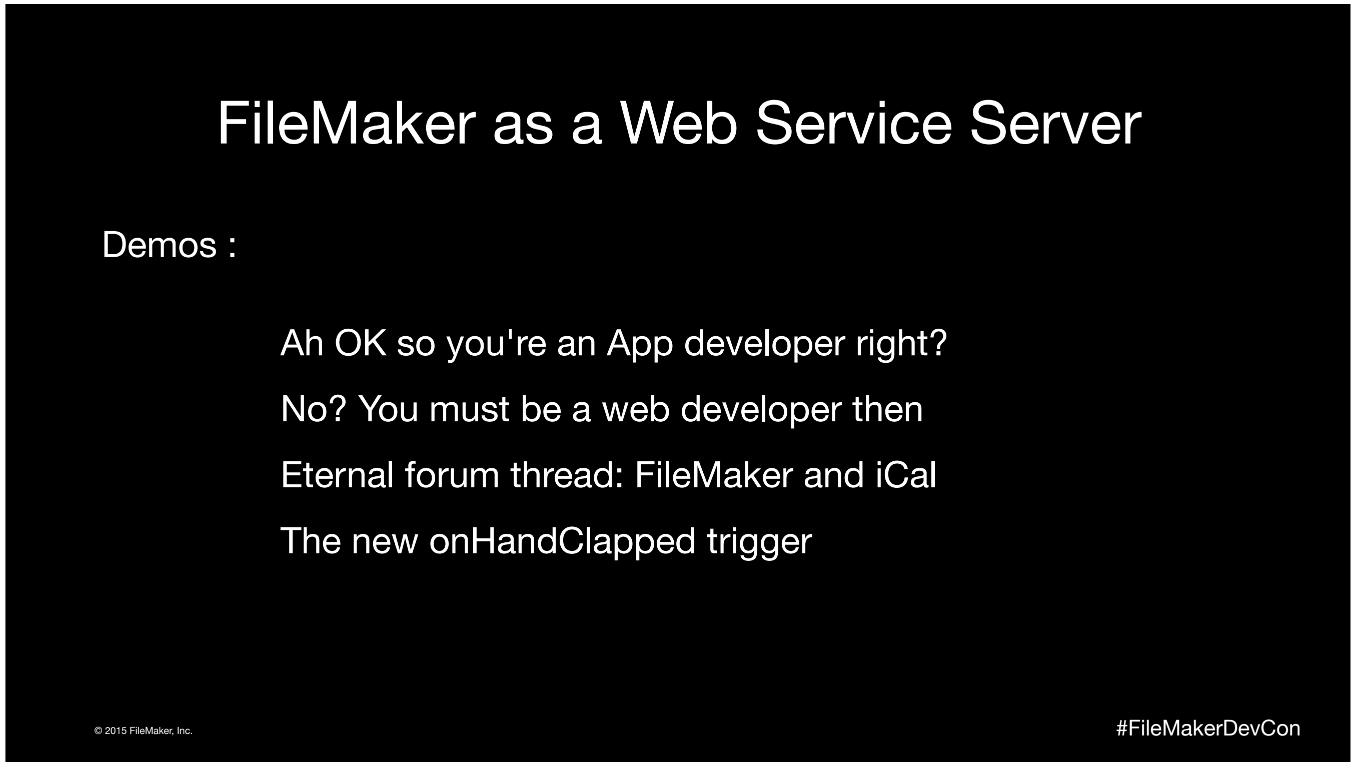
RESTfm is free to download, and has no licence costs.

The big benefit is that REST concepts are well known outside FileMaker, so for example if you've got a web developer who wants to talk to your FileMaker database and you tell him there's a REST API that does JSON you'll make him very happy. Lots of people use JSON for Web apps and it's very well supported outside FileMaker.

The extensibility item is actually a big deal. Us FileMaker developers are used to being the ones that have to make FileMaker fit the other solutions in the business. No one adapts their products to FileMaker, FileMaker adapts to the other products.

So RESTfm is similar, in that instead of the XML API approach where you need to tell the client that your server only talks one language, and they need to learn your language. You can instead tell them that we can adapt our Web Services server to speak your language, whatever it is. It means FileMaker can fit in lots of places where other solutions might not.

And RESTfm is just a way of exposing existing FileMaker functionality. You create a Privilege Set, decide which layouts the user can access, add fields to those layouts, and turn on PHP web publishing, and you're done. Changes to functionality are done with FMP not in PHP. There's no need to know any PHP at all.



FileMaker as a Web Service Server

Demos :

Ah OK so you're an App developer right?

No? You must be a web developer then

Eternal forum thread: FileMaker and iCal

The new onHandClapped trigger

FileMaker Web Service consumed by an iOS app
FileMaker Web Service consumed by a PHP app
Create a record in filemaker with the new (unofficial!) trigger

Session Updates

This session will have updates.

www.filemaker.com/devcon/speaker_updates

(This is also listed in your conference guide)

Thank You!

Please remember to fill out the session evaluation at:
www.filemaker.com/devcon/evaluations

FILEMAKER DEVCON
EXPERIENCE THE POSSIBILITIES
20TH ANNIVERSARY | LAS VEGAS, NEVADA | JULY 20-23, 2015

#FileMakerDevCon