

# Connecting Angular and CFML

Trip Ward

## About Me

- Senior Technical Specialist at ICF
- Owner & Chief Consultant at Trir Software
  - Software Architecture and Development
- ColdFusion(1998), Java, jQuery, HTML5, SQL, etc...
- Member of Northern Virginia CF User Group

## About Nolan Erck

- Code Samples from **Nolan Erck**
  - Owner & Chief Consultant at South of Shasta
    - Software Development, Training

- OpenTable, ZenDesk, Turn It In, AcademyX, HackerLab
- ColdFusion, C++, Java, jQuery, PHP, .NET, HTML5, Android, SQL, etc...
- Manager, SacInteractive User Group
- Music Super Freak!

## **Slide deck and code samples are here:**

[https://github.com/tripward/Presentation\\_Angular\\_CF](https://github.com/tripward/Presentation_Angular_CF)  
Presentation\_Angular\_CF

## **Today's Talk**

- A quick intro APIs
- A quick intro to AngularJS. 1.x
- Several code demos
- Hooking different types of CFML up to an Angular app

# Show of Hands

- Who is brand new to AngularJS?
- Who is *not* using the API Manager?
- Who is *not* using MVC (ColdBox, Framework1, etc)?
- Who is *not* OOP?

# You're all in the right place

- You don't *have* to use API Manager, but you *want* to monitor you.
- You don't *have* to have proper MVC code.
- You don't *have* to have be full OOP to deliver clean API response, but it is easier...
- You don't *have* to have be an Angular guru to start.
- ...but the more you know is obviously helpful.

# For today's talk

- We're using Angular 1.x
- Path of least resistance, fewer moving parts than v2 or 4
- Adobe ColdFusion 2016
- But nothing in the talk is CF 2016 specific

## What's an API?

- Application Programming Interface
- Quasi-generic way to interact with a system
- Just a bunch of URLs
- The URLs return data to whatever system requested it
- Data comes back (hopefully!) in a standard format: XML, SOAP, JSON, etc
- Stateless
- Abstraction of complexity

## API vs REST API

- All REST APIs are APIs

- But not all APIs are RESTful
- They all *do* the same thing
- Just a different style in how you talk to the API

## Non-REST API

- <http://sitename/api/getUser.cfm?ID=123>
- <http://sitename/api/createUser.cfm>
- <http://sitename/api/updateUser.cfm?ID=123>
- <http://sitename/api/deleteUser.cfm?ID=123>
- All HTTP "get" requests
- Can copy/paste all URLs and run them in a browser (for debugging)

## REST API

- <http://sitename/api/user/123> (GET, POST, PUT, DELETE)
- The *same* URL for all actions.
- Change the HTTP verb to change the behavior.

# Verbs

- POST = Create, 201 (Created)
- GET = Read, 200 (OK)
- PUT = Update/Replace, 200 (OK) or 204 (No Content). 404 (Not Found), if ID not found or invalid.
- DELETE = Delete, 200 (OK) or 204 (No Content). 404 (Not Found), if ID not found or invalid.

Note- If using ColdBox/FW1 4 and utilizing routes you may also use your browser to debug.

## API vs REST API

- So which is best?
- REST has a more *consistent* interface out of the box.
  - `http://sitename/api/user/123` (get)
- REST is the current "standard".
- But under the hood, your code does the same "thing" either way.

## Does this mean....?

- ...I have to install something on my CF server? - No
- ...I *have* to use AngularJS? - No
- ...I have to rewrite my whole CF app? - No
- We'll show several ways to use Angular and CFML

# What is AngularJS

- AngularJS is a JavaScript framework. It can be added to an HTML or CF page with a script tag.
- AngularJS extends HTML attributes with Directives, and binds data to HTML with Expressions.
- AngularJS extends HTML with ng-directives.
  - The ng-app directive defines an AngularJS application.
  - The ng-model directive binds the value of HTML controls (input, select, textarea) to application data.
  - The ng-bind directive binds application data to the HTML view.

## [Your first AngularJS app](#)

[Angular + Basic CFM template](#)

[Angular + CFFunction](#)

[Angular + CFC ReturnFormat](#)

[Angular Routes](#)

[Your first RESTful AngularJS app](#)

## Other info

- Don't have to "configure" AngularJS to use CFML. It "just works".
- CF just knows something requested a URL.
- AngularJS just knows it got back some JSON from a URL.
- It all "just works".
- The same is true for any other JS framework - Vue, React, NativeScript, jQuery, etc.

## Other info



- ColdBox REST is great option for REST APIs. Talk to the Ortus folks for more details. Or better yet, attend Into The Box 2018!
- ColdBox REST also "just works" with any JavaScript framework. No config needed.
- FW1 v4 supports REST routing out of the box.
- Taffy is a ColdFusion WebService framework allowing simple remote access calls and self documentation. Works on all versions of CF from 8 thru 2016

**v1.x -> v2 -> v4**

- v1.7 - `input type="text" ng-model="name"`
- v2 - `input [(ngModel)]="username"`
- v4 - `input [(ngModel)]="username"`

## **So what's the catch?**

- A few new conventions to learn, but they're not

terribly hard.

- If you have lots of strange server mappings, may take a little time to configure.
- `preserveStructKeyCase` nuances between CF 9/10/11.

## Other Resources

- South of Shasta - onsite and remote training
- "Object-Oriented Programming in ColdFusion" by Matt Gifford
- [docs.angularjs.org](http://docs.angularjs.org)
- ColdBox Relax
- [docs.taffy.io](http://docs.taffy.io)
- Swagger
- Talk to people at the conference!

## Questions? Comments?

**Trip Ward**

- [trip.ward@icf.com](mailto:trip.ward@icf.com)
- Github: [tripward](https://github.com/tripward)

## **Nolan Erck**

- southofshasta.com
- nolan@southofshasta.com
- Twitter: @southofshasta
- Github: nolanerck

**Thanks!**