

# MOBILE DEVELOPMENT NETWORKING: PART 2

Rudd Taylor  
Founder, SALT

---

## NETWORKING

---

# LEARNING OBJECTIVES

- Practice modeling our data using the MVC pattern
- Reorganize our data into a model

**NETWORKING**

---

# REVIEW OF NETWORKING

---

# NETWORKING

---

## PAIR ACTIVITY

- <http://api.openweathermap.org/data/2.5/weather?q=nyc,ny>
- Create an app, from scratch that displays the weather in NYC. Request the above JSON when the app is started, parse it, then display the weather in a label
  - Note: OWM uses Kelvin by default
- Your program must display an error message in the event of an error
- Bonus: Use SwiftyJSON to parse JSON
- Bonus: Display temperature in either C or F
- Bonus: Add a text input box that accepts a city name and prints out the weather in that city

**NETWORKING**

---

**MODELING DATA & MVC**

---

## **NETWORKING**

---

# **WHAT IS MVC?**

# NETWORKING

---

## WHAT IS MVC?

- A very general design pattern
- Models contain the data, algorithms and business logic of our app
- Views are what the user sees and interacts with
- Controllers are the glue in between models and views (our view controllers)

## NETWORKING

---

# MVC BEST PRACTICES

- Parsing and normalizing of data (e.g. JSON) should be part of your model
  - DEFINITELY not your views
  - Also, not part of your controllers
- Why?



---

## NETWORKING

---

# MVC BEST PRACTICES

- Parsing and normalizing of data (e.g. JSON) should be part of your model
  - DEFINITELY not your views
  - Also, not part of your controllers
- Why?
  - You want to use the same model in multiple screens
  - You don't want changes in your parsing logic to affect your views, or vice versa
  - You want your models and views to be easily reusable in other applications or tests

## NETWORKING

---

# PARSING DATA BEST PRACTICES

- Put parsing logic into a model class
- Model classes defines the important fields you want to use from the JSON
- Create a way to ‘apply’ a blob of data (e.g. JSON) to your model, usually by taking it in in the constructor
- Have a ‘get’ method on the model class that fetches from the appropriate URL and returns a model/models

**NETWORKING**

---

**DATA MODEL CODE-ALONG**

---

## NETWORKING

---

# GROUP ASSIGNMENT

- Change your Mashable headline reader to parse data into a collection of 'Headline' models
- 'Headline' is a class of your own definition that at least contains the title of the headline
- Your view controller should display logic from the collection of headlines
- Bonus: The 'Headline' class should have a class getter that fetches recent headlines. It should take completion handler.

**NETWORKING**

---

**FINAL PROJECT**