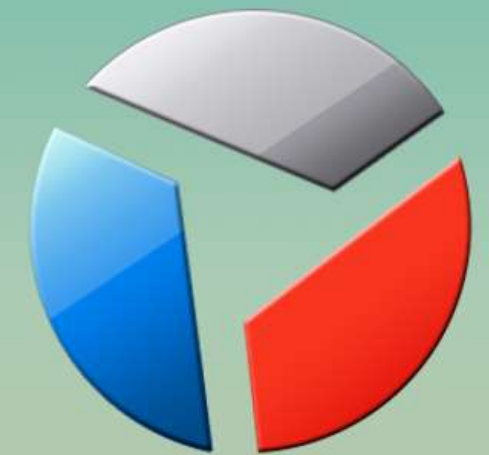


Oracle JET



Oracle JET



Contents

- Introduction
- Installation and Configuration of OJET
- First OJET Application

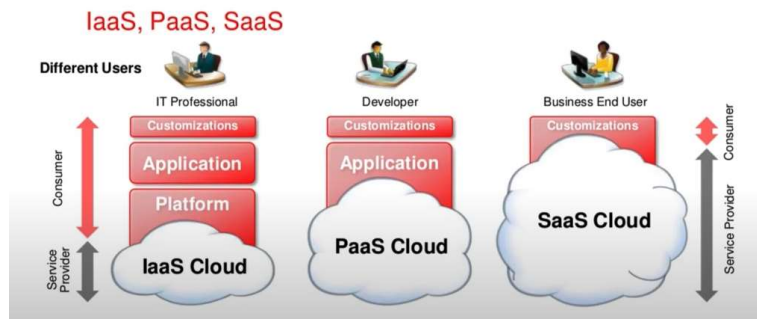
OJET Toolkit



APACHE
CORDOVA™



Modern Business Applications and Cloud

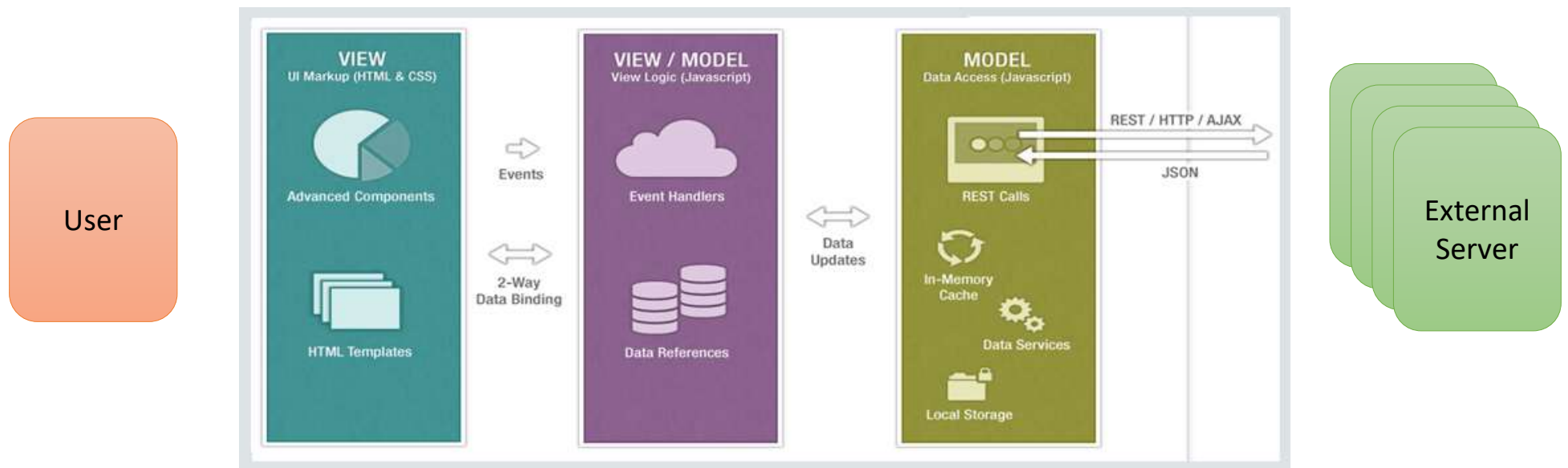


- JS is the native language of the internet
- Research with Cloud architectures
- OJET is the resultant of that research
- Oracle recommendations and best practices for enterprise JS

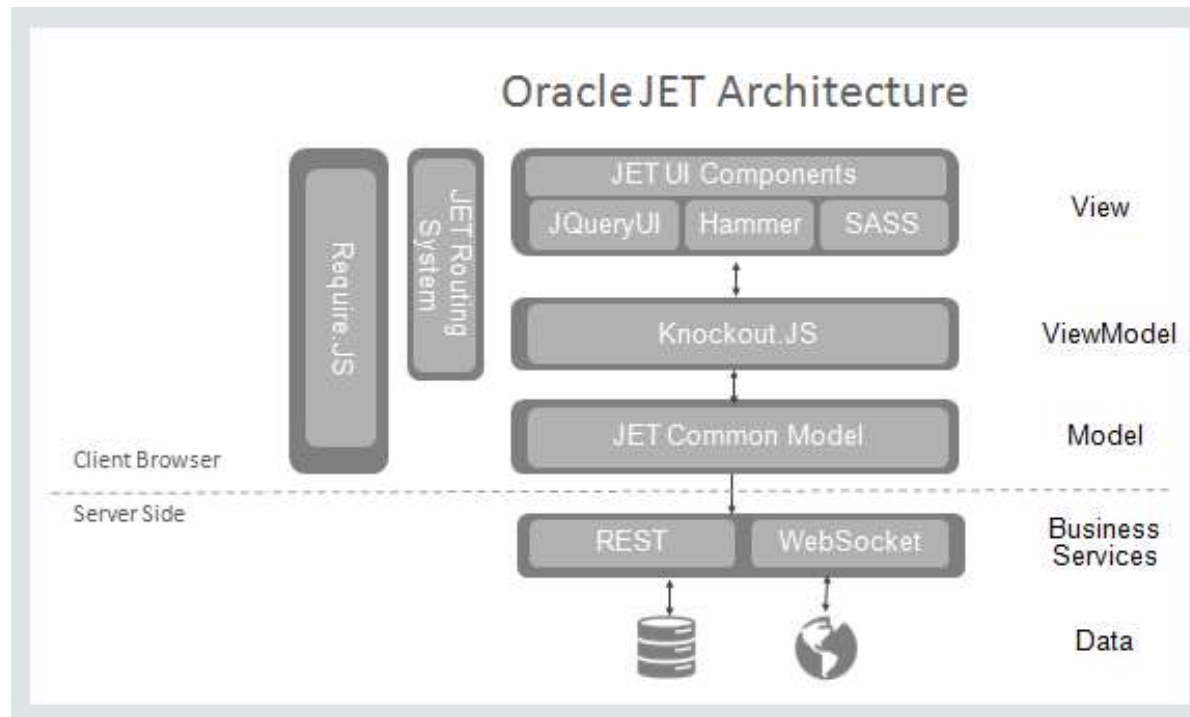
Oracle JET

- Free and Open Source
 - github.com/oracle/oraclejet
- Javascript Toolkit
- Useful Architectures, Patterns, Templates, Techniques, Components
- Focussed on Enterprise Applications
- Javascript, HTML5 and CSS3
- Result of research at Oracle in Javascript enterprise applications

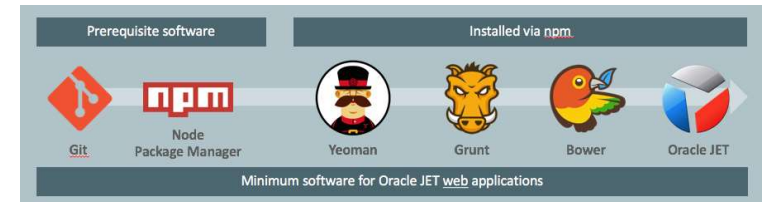
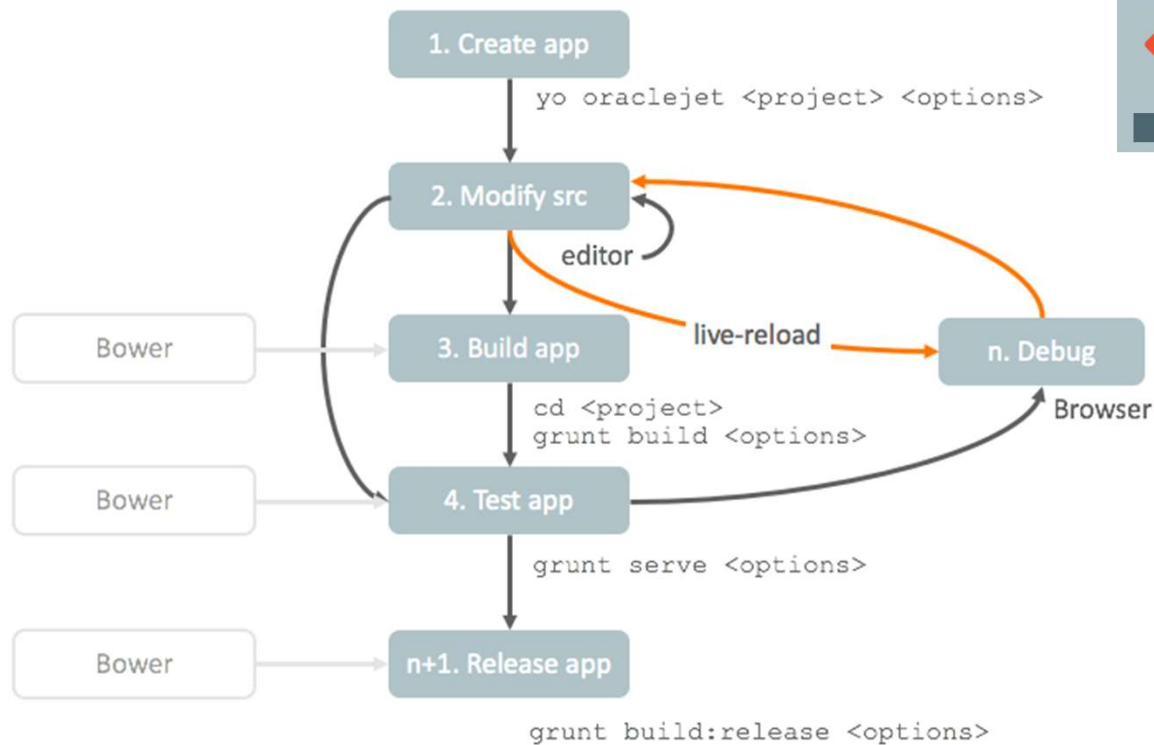
OJET MVVM Concepts



OJET Architecture



Development Process with OJET



Javascript Ecosystem

- Vibrant, innovative ecosystem
- Increasing popularity of mobile devices
- Expense of native applications
- Popularity of hybrid mobile applications
- Challenges of a volatile, dynamic, ecosystem
- Relevant and Maintainable over a number of years

Installation and Configuration

- Download and install node.js
<https://nodejs.org/en/download/>
- Install OJET Client
> (sudo) npm install -g @oracle/ojet-cli
- Verify the OJET CLI
> ojet help
> ojet --version

First OJET Web Application

- Developing client-side web application with Oracle JET is simple and efficient
- Create a starter template
 - > `ojet create firstwebapp --template=navdrawer`
- Run the web application
 - Change the directory to firstwebapp
 - Execute the command

> `ojet serve`



Using Require JS and Knockout JS in OJET

- Require JS helps in dependency management
- Knockout JS helps in data-binding and automatic tracking of value/property changes, it updates all references to that property automatically

Oracle JET Components

- Powerful and clean way of organizing UI code into self-contained reusable chunks
- Development patterns beneficial for large applications, simplifying by encapsulation
- Improve performance by incremental loading of resources

Loading Resources

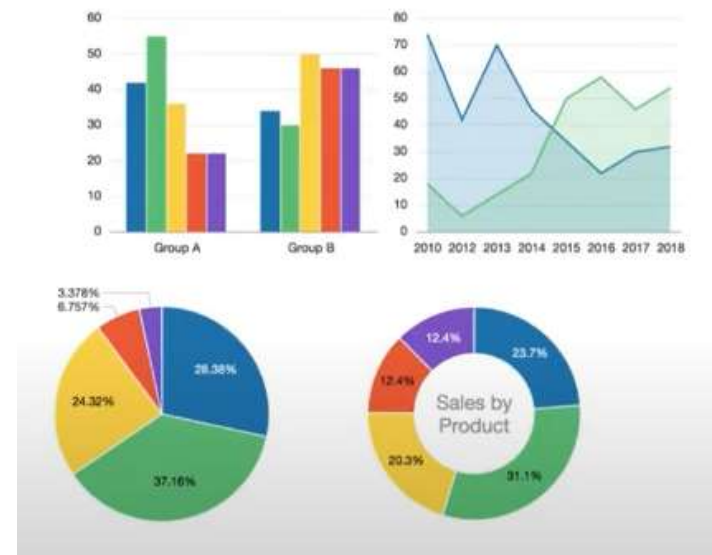
- Require.js Text Plugin is an extension of Require.js
- Enables you to load text based resources
- Let's you separate HTML templates from business logic in Javascript files

Creating Modules

- OJET applications are inherently modular
- An OJET module has its business logic in Javascript and its view in HTML
- An OJET module represents a section in your application providing a new feature to the user

OJET Cookbook Components

- Open sourced
- Online OJET Cookbook provides recipes for each component
 - HTML View
 - JS Business Logic
- JSDoc describes each component



Explore Cookbook, Understand Workflow

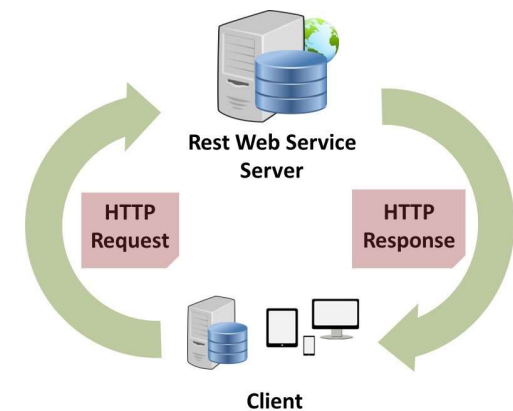
LAB #6

OJET Flex Layouts

- Oracle ALTA UI design systems
<https://www.oracle.com/webfolder/ux/middleware/alta/index.html>
 - Set of CSS classes used to design modern Oracle applications
 - Default theme of Oracle JET
 - Design with mobile first approach
- Flex box layout is the standard part
 - Efficient way to layout, align and distribute space for applications
 - Helps accommodate variety of device visual spaces

Accessing and Integrating Data

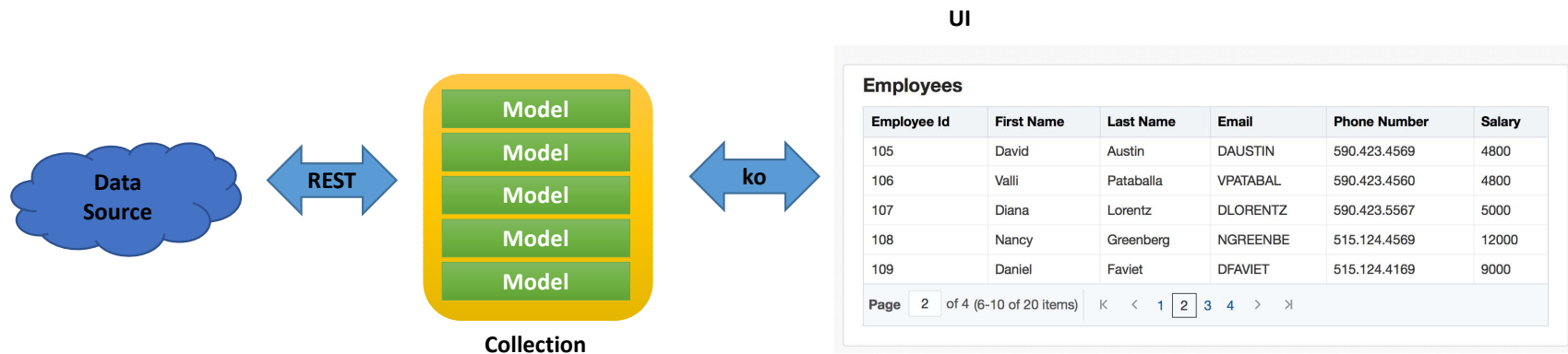
- We access data using REST Services
- Representation State Transfer
 - Data and functionality are resources
 - Resources are accessed using URI
 - Benefit: Simplicity and it's a standard
 - Can expose data via Java EE, Spring Boot, Node, etc
- OJET allows you to create REST clients to integrate REST services
- jQuery API can be used for REST calls



Experiment with a REST endpoint

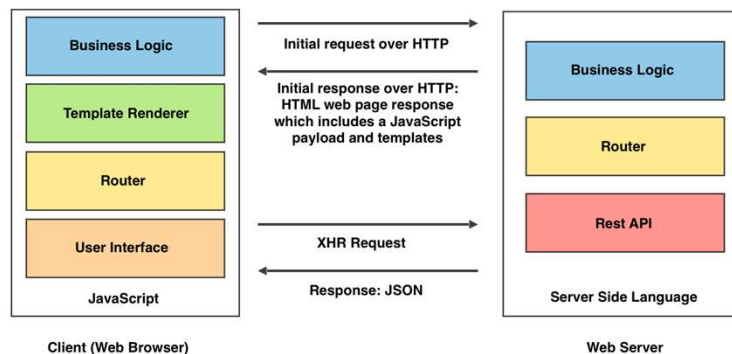
Common Model

- Idiomatic alternative to working with data
- Inspired by Backbone JS
 - Model and Collections
 - CRUD



Single Page Applications

- A single-page application is a web application or website that interacts with the web browser by dynamically rewriting the current web page with new data from the web server, instead of the default method of the browser loading entire new pages
- <https://www.excellentwebworld.com/what-is-a-single-page-application/>



<https://globekit.co/>

Building OJET Applications from Scratch

- Create a blank or basic application and proceed
 - > `ojet create webapp --template=blank|basic`