Drupal 8, Services and Decoupling Patterns

Dinesh Waghmare
About My Self

Dinesh Waghmare, Drupal Mumbai, India

- I’m working with **Tata Consultancy Services**, Powai-Mumbai, India
- Started my Drupal journey in 2009 with Drupal 6, long journey with Drupal 7
  Finally in :Love: of Drupal 8  [https://www.drupal.org/u/dineshw](https://www.drupal.org/u/dineshw)
- Member of Drupal Mumbai, India
- I love Drupal Camps and Cons, Travelling, Yoga Camps, Make Friends and Learn
- Interested areas: E-Commerce, LMS and Decouple
Session Agenda

Drupal 8, Services and Decoupling Patterns

- What is Decoupled Drupal?
- Decoupled Drupal Architecture Patterns
- Factors while deciding Path for Decoupled Drupal 8
- Landscape of possibility with Decoupled Drupal 8
- Risk and Rewards in Decoupled Drupal 8
- User experience for Decoupled Drupal 8
- Modules / Frameworks and Process for bare minimum setup of Decoupled Drupal 8
- Big Picture
- Decoupled Drupal 8 Demo
What is Decoupled Drupal?
Decoupling is the concept of separating applications' backend layer from its presentation layer. Decoupling provides flexibility to front-end developers while backend developers are allowed to focus more on content modelling and building the data access layer. Decoupling is a process of adding a layer of technical abstraction between what content providers create and what content consumers see. Decoupled Drupal is the concept of splitting content from Drupal website's Theme layer and how it gets displayed in single or multiple independent systems.
What is Decoupled Drupal?

Traditional CMS

- Traditional CMS allows web editors to Add / Edit / Publish Content as well as provides control over Presentation logic
- It’s monolithic in nature
- Front end or Presentation layer use Template engine from Theme Layer provided by CMS
- Front End could be Page, Widgets, Blocks etc.
- Example: https://goo.gl/ZR7e46
What is Decoupled Drupal?

Decoupled CMS

- Decoupled CMS allows Web editor to Add / Edit / Publish Content as part of backend while front end is not part of CMS
- Front end developer will able to take full control over presentation logic
- Content of site is accessible via RESTful API's
What is Decoupled Drupal?

Progressively Decoupled

- In Decoupled CMS one may lose existing features of Drupal and it may increase cost to rewrite such features in front end frameworks.
- With Decoupled CMS performance and development time may vary for personalised content blocks, a deep integration of Big Pipe and Theme layer may be lost with adoption of front end framework.
- Thus, rather than decoupling entire page, a part of page or its blocks were decoupled as Progressive decoupled Drupal which able to offer best of both Drupal 8 and Front End JS frameworks.
- Drupal's HTML or serialised output, and then use a JavaScript framework to add client-side interactivity on the client side.
What is Decoupled Drupal?

Progressively Decoupled

Progressive decoupled Drupal uses Drupal to render initial HTML output and then use a JavaScript framework to add client-side or on server side.
Architecture Patterns In Decoupled Drupal 8

Drupal 8, Services and Decoupling Patterns
Static Sites: Central Content Hub pushing Content based on consumer and allowing to generate static content

- Suitable for little or not much end user interactivity
- Good for Responsive design
Architectural Patterns in Decoupled Drupal 8

Decoupled Drupal Patterns for ‘Single Page Web App’

- Single Page Apps: Act is a Progressive application-in-browser.
- It uses new end JS frameworks like Angular, React, Backbone.
- Good for Responsive design.
Anjanaya asana
The Sanskrit word anjaneya means salutation or praise from the root anj which means to honor, to celebrate, to anoint.

Surya Namaskar
The Sanskrit word surya means sun, Namaskar is the word with origin Namaste, from the root Nam means to bow, So Namaskar means salutation, salute, greeting or praise.
Ardha Matsyendrasana

Asana Sanskrit Name: Ardha-matsyendrasana
Asana Pronunciation: ard-ha-mat-sye-n-ah-dra-sa-na
Asana Name: The Half Spinal Twist Pose

Asana Meaning:
Ardha means half. Matsyendra is one of many Siddhas or masters who where accomplished Yogis mentioned in the medieval Yoga text the Hatha-Yoga-Pradipika. This posture is traditionally called the Spinal Twist because the spinal column is twisted gently.

Difficulty: 4

Asana Steps:
Asana Steps:

1. Sit in any comfortable cross-legged position.

2. Straighten the legs out in front. Bend the right knee and bring the heel of the right foot close to the left hip.

3. Inhale and bend the left knee upward and place the left foot flat on the floor to the right of the right leg with the ankle touching the right thigh.

React Single Page Website URL – Details Page: https://goo.gl/CGK1Qj
Hybrid Websites

Both Drupal Front end and JS framework combine to serve front end either loosed or tight integration.

- Good for applications like: Dashboard’s, Quiz, Polls, Graphs, Reports etc.
- Part of front end may communicate with Drupal RESTful API’s
Architecture Patterns In Decoupled Drupal 8

Decoupled Drupal Patterns for ‘Hybrid App’ (Android/iOS/Windows etc)

- Native and Hybrid App: Drupal serves back end and RESTful API’s and acts as Content Repository
- Frontend usually served as native mobile app
- Despite App presence, some user still may prefer to access responsive website
Android App Download URL build within Ionic: https://goo.gl/GV8mhp or https://goo.gl/6mAoqm
Android App Download URL build within Ionic: https://goo.gl/GV8mhp or https://goo.gl/6mAoqm
Multi CMS: Decouple a Drupal CMS may exist with one or more Drupal or Non-Drupal CMS

Helps in separation of concerns between content and layout/presentation and still gives Web Editor flexibility of CMS
Architecture Patterns In Decoupled Drupal 8

Decoupled Drupal Patterns for ‘Integrations’

- Integrations:
  - Voice first integration: Alexa / Google Home etc.
  - Chabot’s
  - IoT Devices: Push button, Monitoring Tools etc.
  - Digital Signage
  - Wearables
Drupal 8 + Alexa Demo Video on Youtube : Yoga Asana Listing and Details
https://goo.gl/LDkiCb
Factors while deciding Path for Decoupled Drupal 8
Deciding Path for Decoupled Drupal 8

- Front End Freedom Vs Drupal Theme
- Integration Point and Omni Channel Touch Points Vs Website: Website, Responsive Mobile Sites, App, Digital Signage, IoT Devices, Wearables, Bot’s and many more...
- One Publisher to Many Consumer: Central content hub and Many Consumers
- Multi-channel and Multi-part Content: Multiple CMS, Media (Videos / Audio / Docs), Social feeds etc.
- Multiple Teams: Mobile App Team, Web Team, Integration Team, Content Team, Review or Compliance Team
Deciding Path for Decoupled Drupal 8

- What do you intend to build: one experience or multi omni channel experience
- Want to create a single standalone website or web application?
- Want to create multi experiences echo system (app, sites, integration points, bots etc.)

Ref: https://dri.es/how-to-decouple-drupal-in-2018
Deciding Path for Decoupled Drupal 8

- Are there things you can't live without: Views, Caching, Role/Access Management
- Check Web Editorial requirements: Preview Content, In Line Editing, Layout Editor
- Check requirements of Developer both front end developer and back end developer

Ref: https://dri.es/how-to-decouple-drupal-in-2018
Deciding Path for Decoupled Drupal 8

Ref: https://dri.es/how-to-decouple-drupal-in-2018
Landscape of possibility with Decoupled Drupal 8
Landscape with Decoupled Drupal 8

- Conversational Integrations
- Full Decoupled
- Progressive Decoupled
- Monolithic
Risk and Rewards in Decoupled Drupal 8

Drupal 8, Services and Decoupling Patterns
Rewards in Decoupled Drupal

- **Reduced dependency or reliance** on Drupal Developers for Backend or Front-End
- **Frontend developers have full Flexibility and control** on Presentation Logic and framework, redesign of website is possible without re-implementing backend CMS
- **True Agility with Decoupled teams**: Front-end and back-end developers no longer need to understand the complex theming layer, modular structures of Drupal 8 architecture, **both team can work independently** allows differentiated development velocities hence improve the velocity and agility of a project
- **Faster Content Publishing**: Decouple CMS are build with mind-set of easing content creation and content workflow to speed up content publishing for editors
Rewards in Decoupled Drupal

- **Flexible and Clean API’s**: All Integration points consumes relatively same API with power of Query, Filters, Include relations etc.

- **Helps in Diversity and Innovation, Interactive Experience beyond Website**: API first, API Ready model allows building quality content and increases app development with various integration points like Conversational UIs, digital signage, Voice First Integrations, IoT, WebVR, Chatbot, Augmented Reality etc.

- **Write Once, Publish Everywhere**: Allows to build Eco-System

- **Get combined benefits of using RESTful API’s, client-side frameworks, Caching** for front end helps in various facts like
Rewards in Decoupled Drupal

- Upgrade proof approach: Front end is independent of upgrade
- Benefits of client-side frameworks
  - Better HTML Templating with clean mark-up
  - Component based design helps reuse and overriding presentation logic
  - Faster Performance by means of maintaining content in a virtual DOM and is rendered to the actual DOM when content changes occur, so auto refresh possible
  - Data Management: application state is managed within the framework's data models instead of scattered throughout the DOM
Risks in Decoupled Drupal

- **One Point of Failure**: Sometimes decoupled architecture is complex to understand in situation of troubleshooting or debugging.
- **Good testing strategy** for API’s and UI should be in place.
- **Strict separation must be followed for front end logic**: Business requirements needs to be evaluated carefully while drafting abstraction and constraints of Decoupled application.
- **High Skilled team with relatively large team size is required**
  - **Full Stack Developer Army**: team with different skillsets
  - **Drupal Ninja’s**: team with multiple skills sets along with Drupal 8
Risks in Decoupled Drupal

- “There is module for that” does not apply: deep integration provided by contrib modules becomes useless for decouple architecture.
- Effort to Reinvent the features like analytics, SEO, UI Layout, Display management, Content and Site Preview, Access Control, Routing etc.
- Loss of Drupal benefits like Security and Input sanitization (CSRF, XSS), improved performance, caching, accessibility, Big Pipe Rendering, Theme Layer and Templating.
- Hard to build UI with Personalized content.
- Limitation of Client side rendering with slow internet connection.
- Pain of Review and Audit.

**Drupal Europe**

Darmstadt, Germany

10 - 14 September 2018
User experience for Decoupled Drupal 8
# User experience for Decoupled Drupal 8

## Front End
- Learn new JS frameworks and select the right framework
- Consider security and performance measures
- Understanding of request formats e.g., JSON / GraphQL

## Site Builder and Backend
- Strong content modelling considering multi-channel content distribution
- RESTful API standards and formats JSON API, GraphQL
- Strategy for testing and documenting API’s
- Strong Authentication

## Business and Editorial
- Re-align with separation of front end and back end
- Revisit Site Preview, Content Publishing workflow
- Auditing and compliance
Modules or Frameworks, Process for setup of Decoupled Drupal 8
## Modules for setup of Decoupled Drupal 8

### Modules for RESTful API and Documentation

<table>
<thead>
<tr>
<th>REST</th>
<th>JSON API</th>
<th>GraphQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modules</td>
<td>rest</td>
<td>jsonapi</td>
</tr>
<tr>
<td>Modules</td>
<td>serialization</td>
<td>jsonapi extras</td>
</tr>
<tr>
<td>Modules</td>
<td>basic auth</td>
<td>jsonapi defaults</td>
</tr>
<tr>
<td>Modules</td>
<td>hal</td>
<td>jsonapi include</td>
</tr>
<tr>
<td>Modules</td>
<td>restui</td>
<td></td>
</tr>
<tr>
<td>Docs</td>
<td>rest api doc</td>
<td>openapi ui + openapi ui swagger or redoc</td>
</tr>
<tr>
<td>Docs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specification</td>
<td>RESTful</td>
<td>jsonapi</td>
</tr>
<tr>
<td>Specification</td>
<td>for non-entity data</td>
<td>for entity, out-of-box</td>
</tr>
<tr>
<td>Specification</td>
<td>any format, any logic, any HTTP method</td>
<td></td>
</tr>
</tbody>
</table>
Big Picture: Decoupled Drupal 8

Drupal 8, Services and Decoupling Patterns
Landscape with Decoupled Drupal 8

- IoT, Voice First, AI First, Distributed CMS
- API First-API Ready, Cloud Ready, Mobile Apps, Single Page Apps, Bots, Static Sites
- Conversational Integrations
- Full Decoupled
- Progressive Decoupled
- Monolithic

- Multi Front End
- Rich UI/UX
- Power of CMS and Front End Frameworks
- Traditional Factory model
- Multi Site
- Single Sites
Demo: Decoupled Drupal 8

Drupal 8, Services and Decoupling Patterns
Demo: Decoupled Drupal 8

Drupal 8 as Decoupled CMS

- Content Modelling:
  - Content Type: Yoga Asana (https://goo.gl/FoihMW)
  - Paragraphs Type: Asana Steps (https://goo.gl/2qmk4u)
  - Vocabulary: Asana Benefits, Asana Difficulty Level (https://goo.gl/LLH5YC)

- RESTful API and Modules:
  - RESTful API: JSONAPI (8.x-1.23), JSONAPI Extras (8.x-2.8)
  - API Documentation: OpenAPI (8.x-1.0-beta2), OpenAPI UI (8.x-1.0-rc1)

- Configure CORS:
  - Git Code: https://goo.gl/TxkjnC
Demo: Decoupled Drupal 8

Drupal 8 and Single Page React App

- Install Node and Test Node and NPM: `node -v`, `npm -v`
- Install create-react-app using NPM: `npm install -g create-react-app`
- Create react app with project name: `npm create-react-app "drupal8-yoga-react"`
- Start the App: `cd drupal8-yoga-react` and then `npm start`
- Use suitable HTTP client for the browser and node.js: `npm install axios`
- Code snippets API Call, Headers and Data Rendering:
  - Use App:
    - `git clone git@github.com:drupal8-decouple-yoga/drupal8-yoga-react.git`
    - `npm install`
    - `npm start` and visit `http://localhost:3000` then `npm build` (if required)
Demo: Decoupled Drupal 8

Drupal 8 and Hybrid Ionic App

To Start with Ionic App Creation

✓ Install Node and Test Node and NPM: `node -v, npm -v`
✓ install the Ionic and Cordova CLI: `npm install -g ionic cordova`
✓ Create ionic app with project name: `ionic start drupal8yoga`
✓ Start the App: `cd drupal8yoga` and then `ionic serve`

Use Angular 4 within Ionic App for Templating, also Typescript and sass can be used

✓ `import { Component, ViewChild } from '@angular/core';`
✓ `import { Config, Nav, Platform } from 'ionic-angular';`

Use App: `git clone git@github.com:drupal8-decouple-yoga/drupal8-yoga-ionic.git` `npm install` followed by `ionic serve` and visit `http://localhost:8100`
Demo: Decoupled Drupal 8

Drupal 8 and Alexa with AWS Cloud

Alexa Yoga App Architecture
Dinesh Waghmare
Solution Architect
Tata Consultancy Services
@dineshweb3
Arihant Chhajed
Cloud and AI Engineer
Tata Consultancy Services
@arihantchhajed

Shafi Khan
UI/UX Lead
Tata Consultancy Services
@Sk123Khan
Rajan Konar
Frontend Developer
Tata Consultancy Services

@rajankonar
Become a Drupal contributor
Friday from 9am

- First timers workshop
- Mentored contribution
- General contribution, I’ll be there too
Warm Thank You
Drupal Europe