



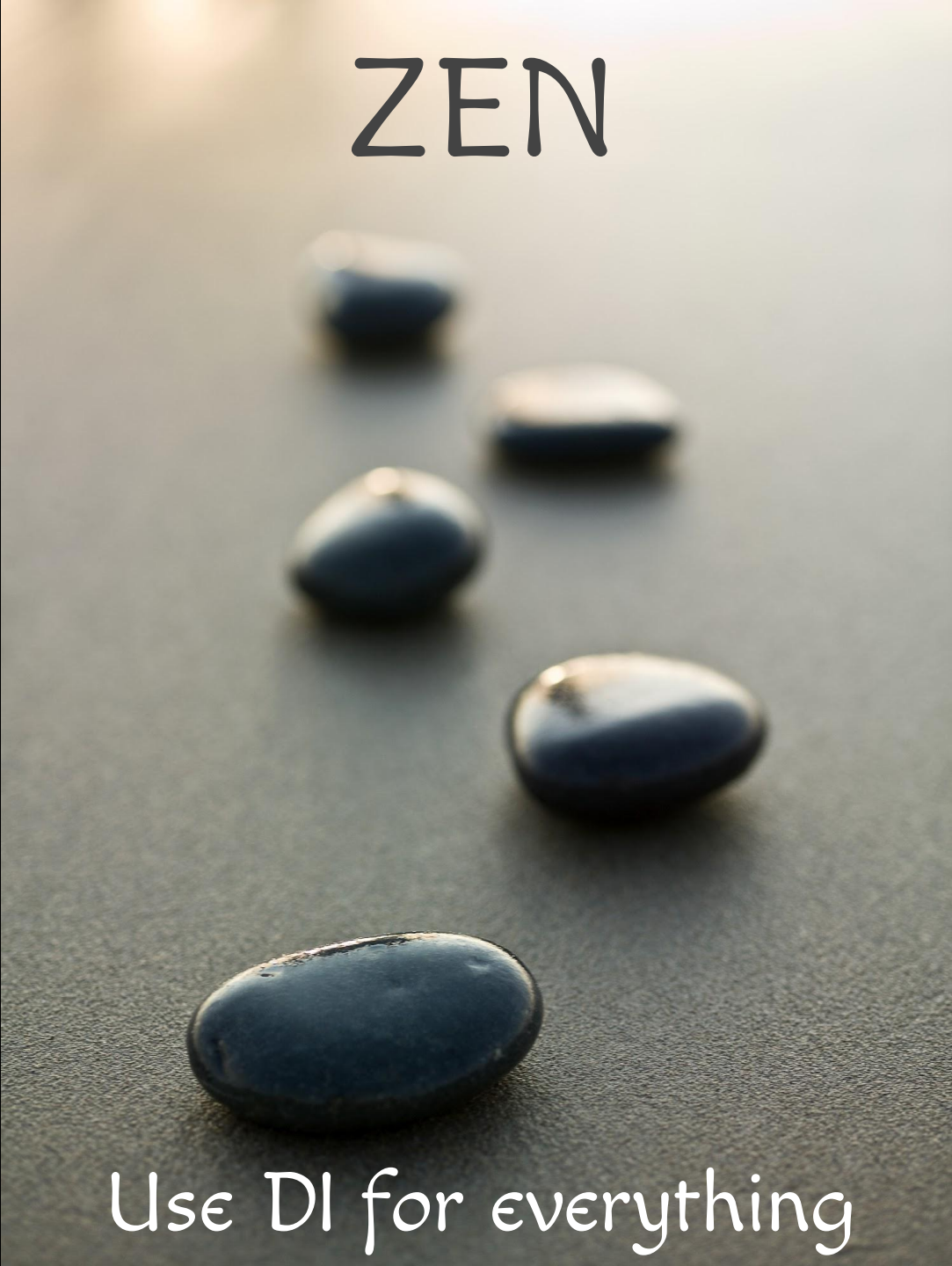
Best Practices

Directory Structure

- Do you need it?
 - Not really, but...
 - Why invent the wheel when you can generate?
 - Working with others
 - Not forgetting important components
- Use what is there:
 - [Yeoman](#)
 - [angular-seed](#)



ZEN

The image features a minimalist composition with five smooth, dark, oval-shaped stones arranged in a diagonal line from the bottom left towards the top right. The stones are set against a light-colored, textured background that resembles sand or a fine-grained surface. The lighting is soft and directional, coming from the upper left, which creates a gentle gradient across the scene and highlights the smooth, polished surfaces of the stones. The overall mood is serene and contemplative.

Use DI for everything

Script Loading

- Options:
 - `<script>` tags at the bottom
 - [ng-app](#) bootstrap
 - [builtwith.angularjs.org](#)
 - AMD (such as [require.js](#))
 - [manual](#) bootstrap



ZEN

Wrap callbacks for 3rd party API into \$apply

Flash of Unstyled Content

- Issue: User will see `{{}}` on index.html
- Solution:
 - hide it with [ng-cloak directive](#)
 - use [ng-bind](#) instead of `{{interpolation}}`

Minification and Compilation

- Do you need it?
 - Angular apps are smaller already
- Minification is an issue because...
 - Angular views use reflection to access model data.
 - basic minification only (no property rename)
 - Dependency injection uses reflection to assemble the application
 - Use proper [annotation](#).
- Never compile angular.min.js (it's already compiled)
 - We are working on better [jscompiler](#) support

A photograph of a narrow, paved path winding through a dense bamboo forest. The bamboo stalks are tall and green, creating a natural archway over the path. Sunlight filters through the leaves, creating a dappled light effect. The path is bordered by low, dry brush. The word "ZEN" is written in large, white, serif capital letters in the upper center of the image.

ZEN

Don't fight the HTML, extend it

Templates

- Extend your HTML and turn it into DSL
 - `<my-component>`
 - `<div my-component>`
 - `<div class="my-component">`
 - `<!-- directive:my-component -->`
- Use a prefix
 - `<my-component>`
 - `<my:component>`
 - broken on IE so use: `<div my-component>`
- Optionally make it valid
 - `<div x-my-component>`
 - `<div data-my-component>`

ZEN

Separate presentation and business logic

Structuring Business Logic

- Controllers
 - should not reference DOM
 - should have view behavior
 - What should happen if user does X
 - Where do I get X from?
- Services
 - should not reference DOM (mostly)
 - are singletons
 - have logic independent of view
 - Do X operation

PS: Do put DOM manipulation in Directives

ZEN

Treat scope as read-only in templates
Treat scope as write-only in controllers



Scope

- Treat scope as read-only in templates & write-only in controllers
 - The purpose of the scope is to refer to model not to be the model.
 - The model is your javascript objects
- When doing bidirectional binding (ng-model) make sure you don't bind directly to the scope properties.
 - unexpected behavior in child scopes

A person is standing in a field of tall, vibrant green grass. The person is wearing a light blue button-down shirt and dark blue jeans. Their right hand is reaching down, with fingers gently touching the blades of grass. The background is a soft-focus expanse of the same grass, creating a sense of depth and tranquility.

ZEN

The \$watch getter function
must always be fast &
idempotent

Structuring modules

- Why multiple modules?
- Usually one module for application
- One module per third-party reusable library
- Possibly for testing
 - create test modules which override services (mocks ngMock)
 - test portion of the app (*)
- Possibly for incremental code loading
- If you have multiple modules per app
 - Group by functionality / feature not by type
 - You should group by view since views will be lazy loaded in near future



ZEN

Develop with non-minified libraries

Deployment Techniques

- minify and concatenate your JS
- gzip enable your server
- index.html (non-cachable)
- cache by version:
 - views
 - code
 - images
 - CSS

Q&A