

COLOR CONVENTION

@myndpm/dyn-forms

@myndpm/dyn-forms/ui-package

@myndpm/dyn-forms/core

Core Service

**DynFormConfig**

```

{
  controls: DynBaseConfig
  modeParams?: DynModeParams
  modes?: DynModeControls
}
    
```

dedicated service DynFormMode to calculate the configs in dyn-factory

**DynBaseConfig**

```

{
  name?: string
  controls?: DynBaseConfig[]
  modes?: DynControlModes;
}
    
```

**dyn-forms**  
 [form]: FormGroup  
 [config]: DynFormConfig  
 [mode?]: DynControlMode

Invokes with a new Injector  
 full of internal tokens  
 MODE  
 VALIDATORS  
 FUNCTIONS

\*ngFor="config.controls"

**dyn-factory**  
 [config]: DynBaseConfig  
 [injector?]: Injector / Parent

retrieves the  
 DynControls from the  
 DynFormRegistry  
 DYN\_CONTROLS\_TOKEN

**UI Modules**

Configs can be build with  
 typed Factory Methods

Controls are provided with  
 UIModule.forFeature()

new level of the hierarchy for subcontrols

**dyn-group**  
 [name?]: string  
 [group]: FormGroup  
 [controls]: DynBaseConfig[]

new level of the hierarchy with a local DynFormNode  
 dynamically and update the input parameters

**DynControl**  
 [config]: DynBaseConfig  
 [params]: DynControlParams

extends a DynForm\* class

**DynControl**  
 dynControl: DynControlType  
 dynInstance: DynInstanceType

```

_logger: DynLogger
_formFactory: DynFormFactory
_ref: ChangeDetectorRef
_paramsChanges: Subject<void>
_unsubscribe: Subject<void>
    
```

**DynControl Setup**

```

DynControlEvents:
node

DynForm* class:
control

DynFactory:
config
params
    
```

dyn-form facades the supported stuff by the Nodes

**DynFormNode**  
 load(config, control)  
 unload()

Load/Unload DynControls  
 manages the hierarchy  
 tracks the control's path  
 register the child Nodes  
 TODO create the Validators

listen observable\$  
 and call internal DynControl methods  
 bidirectional link

propagate events  
 calling child/parent nodes

**DynControlEvents**  
 node = Injector.get(DynFormNode)

```

callHook(event)
callChildHooks(event)
    
```

extends

**DynFormControl**  
 [control]: FormControl  
 node.load(config, control)

**DynFormContainer**  
 [control]: FormGroup / Parent  
 node.load(config, control)

**DynFormGroup**  
 [control]: FormGroup / Parent  
 node.load(config, control)

**DynFormArray**  
 [control]: FormArray  
 node.load(config, control)  
 callChildHooks(event)  
 addItem()  
 removeItem(index)

controls are created with  
 DynFormFactory  
 DynBaseConfig -> AbstractControl