***Let’s see in depths***-

**1)** Typescript 2.7+ supports

**2)** Added Angular Material and CDK Stable

**3)** Component Dev Kit (CDK) - CDK allows you to build your own library of UI components using Angular Material.

**4)** Improved decorator error messages

**5)** Fix platform-detection example for Universal

**6)** Ivy Renderer - It is a new backward compatible and main focused area - speed improvements, size reduction, and increased flexibility.

**7)** Add afterContentInit and afterContentChecked to render

**8)** Added to supports of nativeElement

**9)** Added Optional generic type for ElementRef

The Example looks like -

@ViewChild('your-element') yourElement:ElementRef;

**10)** Bazel Compiler - Bazel only rebuilds what is necessary.

**11)** Added Test Comment

**12)** Add missing lifecycle tests for projected components

**13)** Closure Compiler - Closure Compiler consistently generates smaller bundles.

**14)** Rename QueryPredicate to LQuery and LQuery to LQueries

**15)** Service Worker - Service worker is a script that runs in the web browser. It also manages caching for an application.

**16)** Added multiple validators for array method of FormBuilder

**17)** Handle string with and without line boundary - Now Handle string with and without line boundary (^ & $) on pattern validators. Previously, it works with string not boundaries.

**18)** AbstractControl statusChanges - Previous version, not emits an event when you called “markAsPending” but now emits an event of "PENDING" when we call AbstractControl markAsPending.

**19)** Updates on NgModelChange - Now emitted after value and validity is updated on its control. Previously, it was emitted before updated.

**20)** Allow HttpInterceptors to inject HttpClient –

Previously, an interceptor attempting to inject HttpClient directly would receive a circular dependency error, as HttpClient was constructed via a factory which injected the interceptor instances. Users want to inject HttpClient into interceptors to make supporting.

Either HttpClient or the user has to deal specially with the circular Dependency. This change moves that responsibility into HttpClient itself. By utilizing a new class HttpInterceptingHandler which lazily Loads the set of interceptors at request time, it's possible to inject HttpClient directly into interceptors as construction of HttpClient no longer requires the interceptor chain to be constructed.

**21)** Add navigationSource and restoredState to NavigationStart – Currently, NavigationStart there is no way to know if navigation was triggered imperatively or via the location change. These two use cases should be handled differently for a variety of use cases (e.g., scroll position restoration). This PR adds a navigation source field and restored navigation id (passed to navigations triggered by a URL change).

**22)** Add type and hooks to directive def

**23)** Enable size tracking of a minimal CLI render3 application

**24)** Add canonical view query

**25)** Language Service – The 2.6 version of Typescript’s “resolveModuleName” started to require paths passed to be separated by '/' instead of being able to handle '\'.

**Included Key Features - Angular 5**

1)     Include Representation of Placeholders to xliff and xmb in the compiler

2)      Include an Options Arg to Abstract Controls in the forms controls

3)      Include add default updateOn values for groups and arrays to form controls

4)      Include updateOn blur option to form controls

5)      Include updateOn submit option to form controls

6)      Include an Events Tracking Activation of Individual Routes

7)      Include NgTemplateOutlet API as stable in the common controls

8)     Create StaticInjector which does not depend on Reflect polyfill

9)      Include [@.disabled] attribute to disable animation children in the animations

10)  Make AOT the default

11)  Watch mode

12)  Type checking in templates

13)  More flexible metadata

14)  Remove \*.ngfactory.ts files

15)  Better error messages

16)  Smooth upgrades

17)  Tree-Shakeable components

18)  Hybrid Upgrade Application

**Included Performance Improvements - Angular 5**

1)     Use of addEventListener for the faster rendering and it is the core functionality.

2)     Update to new version of build-optimizer.

3)      Include some Improvements on the abstract class methods and interfaces

4)     Remove decorator DSL which depends on Reflect for Improve the Performance of Apps and This is the core functionality.

5)      Include an option to remove blank text nodes from compiled templates

6)     Switch Angular to use Static-Injector instead of Reflective-Injector.

7)     Improve the applications testing.

8)     Improve the performance of hybrid applications

9)     Improvements on Lazy loading for Angular

**Improvement on HttpClient – Included**

1)     Improvement on Type-checking the response

2)     Improvement on Reading the full response

3)     Improvement on Error handling and fetching error details

4)     Improvement on Intercepting all requests or responses

5)     Improvement on Logging

6)     Improvement on Caching

7)     Improvement on XSRF Protection

**Angular Router Life Cycle Events -**

Added new router life cycle events for Guards and Resolvers -

1)     GuardsCheckStart,

2)     GuardsCheckEnd,

3)     ResolveStart and

4)     ResolveEnd

[**What’s New in Angular 4? AND what are the Improvements in Angular 4?**](https://www.code-sample.com/2017/03/angular-4-vs-angular-2-difference.html)

Angular 4 contains some additional Enhancement and Improvement. Consider the following enhancements.

**1.**    Smaller & Faster Apps

**2.**    View Engine Size Reduce

**3.**    Animation Package

**4.**    NgIf and ngFor Improvement

**5.**    Template

**6.**    NgIf with Else

**7.**    Use of AS keyword

**8.**    Pipes

**9.**    HTTP Request Simplified

**10.** Apps Testing Simplified

**11.** Introduce Meta Tags

**12.** Added some Forms Validators Attributes

**13.** Added Compare Select Options

**14.** Enhancement in Router

**15.** Added Optional Parameter

**16.** Improvement Internationalization