

Angular uses [Semantic Versioning (SEMVER)](http://semver.org/).

Semantic Versioning is all about adding meaning to version numbers. This allows developers to not only reason about any upgrade we do, but we can even let tools such as NPM do it in a automatic and safe manner for us.  
A semantic version consists of three numbers:



Whenever you fix a bug and release it, you increase the last number, if a new feature is added, you increase the second number and whenever you release a breaking change you increase the first number.

“A breaking change happens whenever you as a developer and consumer of a library, have to step in and adjust your code after a version upgrade.”

For example, giving a compiler error for existing application bugs that went unnoticed with the previous compiler version, anything, that will break an existing application when upgrading Angular, requires the team to bump the major version number.

We went from Angular 1 to Angular 2, and it was a total breaking change, with new APIs, new patterns. That was obvious: ultimately Angular 2 was a complete rewrite.

Changing from version 2 to version 4, 5, … won’t be like changing from Angular 1. It won’t be a complete rewrite, it will simply be a change in some core libraries that demand a major SEMVER version change. Also, there will be proper [deprecation phases](http://angularjs.blogspot.com/2016/10/versioning-and-releasing-angular.html#Deprecation_policy_31) to allow developers to adjust their code.

**Naming guidelines**

Basically from now on, you should name versions 2.0.0 or later of Angular simply “Angular”. Try to avoid using the version number, unless it is really necessary to disambiguate.  
  
Three simple guidelines:

* Use “Angular” for versions 2.0.0 and later (e.g. “I’m an Angular developer”, “This is an Angular meetup”, “The Angular ecosystem is growing quickly”)
* Use "AngularJS" to describe versions 1.x or earlier
* Use the version number “Angular 4.0” "Angular 2.4" when needed to talk about a specific release (e.g. when talking about a newly introduced feature - “This is an introduction to feature X, introduced in Angular 4”, “I’m proposing this change for Angular 5”)
* Use full semver version when reporting a bug (e.g. “This issue is present as of Angular 2.3.1”)

For example :“This article uses Angular v2.3.1.”

That helps avoid confusion for your readers, especially when you are writing about specific APIs.

# Why not version 3 then?

* The core Angular libraries live in one single GitHub repository at [github.com/angular/angular](https://github.com/angular/angular). All of them are versioned the same way, but distributed as different NPM packages:



**Due to this** [**misalignment of the router package’s version**](http://angularjs.blogspot.com/2016/06/improvements-coming-for-routing-in.html), the team decided to go straight for Angular v4. In this way again, all the core packages are aligned which will be easier to maintain and help avoid confusion in the future.  
Also it is important to understand how Angular is being used and integrated inside Google (Igor [speaks about this here in his keynote](https://youtu.be/aJIMoLgqU_o?t=9m10s)



