**Time intervals** are irregular! For example, there are 60 seconds in a minute, but 24 hours in a day. Even more confusing, some days have 23 or 25 hours due to [daylight saving time](http://en.wikipedia.org/wiki/Daylight_saving_time), and the standard [Gregorian calendar](http://en.wikipedia.org/wiki/Gregorian_calendar) uses months of differing lengths. And then there are leap years and leap seconds!

To simplify manipulation of and iteration over time intervals, D3 provides a handful of time utilities in addition to the time [scale](Time-Scales.html) and [format](Time-Formatting.html). The utilities support both local time and UTC time. Local time is determined by the browser's JavaScript runtime; arbitrary time zone support would be nice, but requires access to the Olson zoneinfo files..

Time intervals是不规则的。例如，一分钟有60秒，但一天有24个小时。更麻烦的是，按照夏令时有些一天23小时，有些却是25小时，标准的公历年每个月也是日数不一样，而且还有闰年和闰秒！

为了time interval的简单操作和循环，D3除了time  [scale](Time-Scales.html) 和[format](Time-Formatting.html)之外，还提供了大量的time工具。这些工具支持本地时间和UTC时间，本地时间就是浏览器的Javascript的运行时间；支持随机的时区可能会很好，但需要读取Olson时区信息文件。

## Interval（时间间隔）

[#](" \l "wiki-interval) d3.time.*interval*

Returns the specified interval. The following intervals are supported:

返回指定的interval。支持以下时间间隔：

* d3.time.[second](#wiki-second)
* d3.time.[minute](#wiki-minute)
* d3.time.[hour](#wiki-hour)
* d3.time.[day](#wiki-day)
* d3.time.[week](#wiki-week) (alias for d3.time.[sunday](#wiki-sunday))
* d3.time.[sunday](#wiki-sunday)
* d3.time.[monday](#wiki-monday)
* d3.time.[tuesday](#wiki-tuesday)
* d3.time.[wednesday](#wiki-wednesday)
* d3.time.[thursday](#wiki-thursday)
* d3.time.[friday](#wiki-friday)
* d3.time.[saturday](#wiki-saturday)
* d3.time.[month](#wiki-month)
* d3.time.[year](#wiki-year)

[#](" \l "wiki-_interval) *interval*(*date*)

Alias for interval.floor(date). For example, d3.time.day(new Date()) returns midnight (12:00 AM) on the current day, in local time.

等同于 interval.floor(date)。例如， d3.time.day(new Date())返回本地时间的0点日期时间。

[#](" \l "wiki-interval_floor) *interval*.**floor**(*date*)

Rounds down the specified date, returning the latest time interval before or equal to date. For example, d3.time.day.floor(new Date())returns midnight (12:00 AM) on the current day, in local time.

向下四舍五入参数*date*，返回最近的time interval早于或等于*date*。例如，d3.time.day.floor(new Date())返回本地时间的0点日期时间。

[#](" \l "wiki-interval_round) *interval*.**round**(*date*)

Rounds up or down the specified date, returning the closest time interval to date. For example, d3.time.day.round(new Date()) returns midnight (12:00 AM) on the current day if it is on or before noon, and midnight of the following day if it is after noon.

向上或向下四舍五入参数*date*，返回最接近于*date*的时间。

比如： d3.time.day.round(new Date()) ，如果当前时间是在中午之前刚返回当前0点的日期时间；如果当前时间是在中午之后刚返回下一日0点的日期时间。

[#](" \l "wiki-interval_ceil) *interval*.**ceil**(*date*)

Rounds up the specified date, returning the earliest time interval after or equal to date. For example, d3.time.day.ceil(new Date()) returns midnight (12:00 AM) on the following day, in local time (unless you happen to run this code at exactly midnight, in which case it returns the current time).

向上四舍五入参数*date*，返回最近的time interval晚于或等于*date*。例如，d3.time.day.ceil(new Date())返回晚于或等于本地当前时间的0点日期时间。

比如： d3.time.day.ceil(new Date()) ，返回的是下一日的0点日期时间（除非代码在运行时正好是0点，这样它就返回当前的日期时间）。

[#](" \l "wiki-interval_range) *interval*.**range**(*start*, *stop*[, *step*])

Returns every time interval after or equal to start and before stop. If step is specified, then every step'th interval will be returned, based on the interval number (such as day of month for d3.time.day). For example, a step of 2 will return the 1st, 3rd, 5th etc. of the month with d3.time.day.

返回晚于或等于start 和 早于 stop的时间点。如果还设置了step步长参数， 那么将返回基于interal数字的每个step'th interval。比如，步长为2时，将返回d3.time.dayg 一个月的1st，3rd，5th...

[#](" \l "wiki-interval_offset) *interval*.**offset**(*date*, *step*)

Returns a new date equal to date plus step intervals. If step is negative, then the returned date will be before the specified date; if step is zero, then a copy of the specified date is returned. This method does not round the specified date to the interval. For example, if it is currently 5:34 PM, then d3.time.day.offset(new Date(), 1) returns 5:34 PM tomorrow (even if Daylight Savings Time changes!).

返回一个新的日期时间，其等于 date加上step 。如果step 为负值，那么返回的日期时间将早于date；如果step为0，则返回date的拷贝。

比如：当前时间是5:34PM，d3.time.day.offset(new Date(), 1) 返回明天的5:34PM（即使变化为夏令时）。

[#](" \l "wiki-interval_utc) *interval*.**utc**

Returns a corresponding time interval in UTC rather than local time. For example, d3.time.day.range(start, stop) returns local time days between start and stop, while d3.time.day.utc.range(start, stop) returns UTC days between start and stop.

返回一个相应的UTC时间，而非当地时间。比如：d3.time.day.range(start, stop) 返回的是间于*start*和*stop*的本地时间，而 d3.time.day.utc.range(start, stop) 返回的是间于*start*和*stop*的UTC时间。

## Intervals

[#](" \l "wiki-second) d3.time.**second**

Seconds (e.g., 01:23:45.0000 AM). Always 1,000 milliseconds long.

秒（如01:23:45.0000 AM）。总是1000毫秒为单位。

[#](" \l "wiki-minute) d3.time.**minute**

Minutes (e.g., 01:02:00 AM). Most browsers do not support leap seconds, so minutes are almost always 60 seconds (6e4 milliseconds) long.

分（如01:02:00 AM）。许多浏览器不支持闰秒，因此总是以60秒为单位。

[#](" \l "wiki-hour) d3.time.**hour**

Hours (e.g., 01:00 AM). 60 minutes long (36e5 milliseconds). Note that advancing time by one hour can return the same hour number, or skip an hour number, due to Daylight Savings Time.

小时（如01:00 AM）。以60分为单位 。

[#](" \l "wiki-day) d3.time.**day**

Days (e.g., February 7, 2012 at 12:00 AM). Most days are 24 hours long (864e5 milliseconds); however, with Daylight Savings Time, a day may be 23 or 25 hours long.

天（如February 7, 2012 at 12:00 AM）。

[#](" \l "wiki-week) d3.time.**week**

Alias for d3.time.[sunday](#wiki-sunday). A week is always 7 days, but ranges between 167 and 169 hours depending on Daylight Savings Time.

等同于d3.time.sunday。一周总是7天，167和169小时的时间域取决于是否夏令时。

[#](" \l "wiki-sunday) d3.time.**sunday**

Sunday-based weeks (e.g., February 5, 2012 at 12:00 AM).

基于周日的星期

[#](" \l "wiki-monday) d3.time.**monday**

Monday-based weeks (e.g., February 6, 2012 at 12:00 AM).

基于周一的星期

[#](" \l "wiki-tuesday) d3.time.**tuesday**

Tuesday-based weeks (e.g., February 7, 2012 at 12:00 AM).

基于周二的星期

[#](" \l "wiki-wednesday) d3.time.**wednesday**

Wednesday-based weeks (e.g., February 8, 2012 at 12:00 AM).

基于周三的星期

[#](" \l "wiki-thursday) d3.time.**thursday**

Thursday-based weeks (e.g., February 9, 2012 at 12:00 AM).

基于周四的星期

[#](" \l "wiki-friday) d3.time.**friday**

Friday-based weeks (e.g., February 10, 2012 at 12:00 AM).

基于周五的星期

[#](" \l "wiki-saturday) d3.time.**saturday**

Saturday-based weeks (e.g., February 11, 2012 at 12:00 AM).

基于周六的星期

[#](" \l "wiki-month) d3.time.**month**

Months (e.g., February 1, 2012 at 12:00 AM). Ranges between 28 and 31 days.

[#](" \l "wiki-year) d3.time.**year**

Years (e.g., January 1, 2012 at 12:00 AM). Normal years are 365 days long; leap years are 366.

年（如January 1, 2012 at 12:00 AM）。通常一年是365天，闰年是366天。

## Aliases（化名）

[#](" \l "wiki-seconds) d3.time.**seconds**(*start*, *stop*[, *step*])

Alias for d3.time.[second](#wiki-second).[range](#wiki-interval_range). Returns the second boundaries (e.g., 01:23:45 AM) after or equal to start and before stop. If step is specified, then every step'th second will be returned, based on the second of the minute. For example, a step of 15 will return 9:01:45 PM, 9:02:00 PM, 9:02:15 PM, etc.

等同于d3.time.[second](#wiki-second).[range](#wiki-interval_range) 。返回等于或晚于*start*时间点和早于*stop*时间点的精确到秒的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为15，则返回9:01:45 PM, 9:02:00 PM, 9:02:15 PM…

[#](" \l "wiki-minutes) d3.time.**minutes**(*start*, *stop*[, *step*])

Alias for d3.time.[minute](#wiki-minute).[range](#wiki-interval_range). Returns the minute boundaries (e.g., 01:23 AM) after or equal to start and before stop. If step is specified, then every step'th minute will be returned, based on the minute of the hour. For example, a step of 15 will return 9:45 PM, 10:00 PM, 10:15 PM, etc.

等同于d3.time. [minute](#wiki-minute).[range](#wiki-interval_range). 返回等于或晚于*start*时间点和早于*stop*时间点的精确到分的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为15，则返回9:45 PM, 10:00 PM, 10:15 PM…

[#](" \l "wiki-hours) d3.time.**hours**(*start*, *stop*[, *step*])

Alias for d3.time.[hour](#wiki-hour).[range](#wiki-interval_range). Returns the hour boundaries (e.g., 01 AM) after or equal to start and before stop. If step is specified, then everystep'th hour will be returned, based on the hour of the day. For example, a step of 3 will return 9 PM, 12 AM, 3 AM, etc.

等同于d3.time. [hour](#wiki-hour).[range](#wiki-interval_range).返回等于或晚于*start*时间点和早于*stop*时间点的精确到小时的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为3，则返回9PM, 12AM, 3AM…

[#](" \l "wiki-days) d3.time.**days**(*start*, *stop*[, *step*])

Alias for d3.time.[day](#wiki-day).[range](#wiki-interval_range). Returns the day boundaries (midnight) after or equal to start and before stop. If step is specified, then everystep'th date will be returned, based on the day of the month. For example, a step of 2 will return the 1st, 3rd, 5th etc. of the month.

等同于d3.time. [day](#wiki-day).[range](#wiki-interval_range).返回等于或晚于*start*时间点和早于*stop*时间点的精确到天的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为2，则返回这个月的1st ，3rd，5th…

[#](" \l "wiki-weeks) d3.time.**weeks**(*start*, *stop*[, *step*])   
[#](" \l "wiki-sundays) d3.time.**sundays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-mondays) d3.time.**mondays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-tuesdays) d3.time.**tuesdays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-wednesdays) d3.time.**wednesdays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-thursdays) d3.time.**thursdays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-fridays) d3.time.**fridays**(*start*, *stop*[, *step*])   
[#](" \l "wiki-saturdays) d3.time.**saturdays**(*start*, *stop*[, *step*])

Aliases for d3.time.*interval*.[range](#wiki-interval_range) etc. Returns the week boundaries (midnight Sunday) after or equal to start and before stop. If step is specified, then every step'th week will be returned, based on the week of the year. For example, a step of 4 will return January 2, January 30, February 27, etc.

等同于d3.time.*interval*.[range](#wiki-interval_range).返回等于或晚于*start*时间点和早于*stop*时间点的精确到周的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为4，则返回January 2, January 30, February 27…

[#](" \l "wiki-months) d3.time.**months**(*start*, *stop*[, *step*])

Alias for d3.time.[month](#wiki-month).[range](#wiki-interval_range). Returns the month boundaries (e.g., January 01) after or equal to start and before stop. If step is specified, then every step'th month will be returned, based on the month of the year. For example, a step of 3 will return January, April, July, etc.

等同于d3.time. [month](#wiki-month).[range](#wiki-interval_range).返回等于或晚于*start*时间点和早于*stop*时间点的精确到月的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为3，则返回January, April, July …

[#](" \l "wiki-years) d3.time.**years**(*start*, *stop*[, *step*])

Alias for d3.time.[year](#wiki-year).[range](#wiki-interval_range). Returns the year boundaries (midnight January 1st) after or equal to start and before stop. If step is specified, then every step'th year will be returned. For example, a step of 5 will return 2010, 2015, 2020, etc.

等同于d3.time. [year](#wiki-year).[range](#wiki-interval_range).返回等于或晚于*start*时间点和早于*stop*时间点的精确到年的所有时间。如果设置了*step*步长，则返回以每个步长为间隔的时间。比如：*step*为5，则返回2010, 2015, 2020…

## Counting(计数)

[#](" \l "wiki-dayOfYear) d3.time.**dayOfYear**(*date*)

Returns the day number for the given date. The first day of the year (January 1) is always the 0th day. Unlike the [d3.time.format](Time-Formatting.html)'s %j directive, dayOfYear is 0-based rather than 1-based.

返回参数*date*在一年里是第几日。每年的第一天（1月1日）是第0天，是从0开始计数，而非从1开始计数。

[#](" \l "wiki-weekOfYear) d3.time.**weekOfYear**(*date*)   
[#](" \l "wiki-sundayOfYear) d3.time.**sundayOfYear**(*date*)   
[#](" \l "wiki-mondayOfYear) d3.time.**mondayOfYear**(*date*)   
[#](" \l "wiki-tuesdayOfYear) d3.time.**tuesdayOfYear**(*date*)   
[#](" \l "wiki-wednesdayOfYear) d3.time.**wednesdayOfYear**(*date*)   
[#](" \l "wiki-thursdayOfYear) d3.time.**thursdayOfYear**(*date*)   
[#](" \l "wiki-fridayOfYear) d3.time.**fridayOfYear**(*date*)   
[#](" \l "wiki-saturdayOfYear) d3.time.**saturdayOfYear**(*date*)

Returns the week number for the given date, where weeks start with the given *day*. The first day of the year (January 1) is always the 0th week. weekOfYear is an alias for sundayOfYear, which is equivalent to [d3.time.format](Time-Formatting.html)'s %U directive. mondayOfYear is equivalent to[d3.time.format](Time-Formatting.html)'s %W directive.

返回*date*在一年里的第几周。一年的第一周永远都是第0周。weekOfYear等同于sundayOfYear，等同于[d3.time.format](Time-Formatting.html)的%U命令。mondayOfYear等同于[d3.time.format](Time-Formatting.html)的%W命令