

- 本文档由[VisualCrew小组](#)耗时两年翻译，保持与[D3 V3最后一版 \(3.5.17\)](#)一致。D3 V4最新版API请参考[d3.v4-API-Translation](#)。
- 如发现翻译不当或有其他问题可以通过以下方式联系译者：
 - 邮箱：zhang_tianxu@sina.com
 - QQ群：[D3.js](#):437278817, [大数据可视化](#): 436442115
- API使用方法可参考：<https://github.com/tianxuzhang/d3-api-demo>

D3库中所有函数都在“d3”命名空间内。

D3 使用[语义版本命名](#)。你可以使用“d3.version”查看D3的最新版本

D3 API总览

- [核心](#) - 包括选择器，过渡，数据处理，本地化，颜色等。
- [地理](#) - 球面坐标，经纬度运算。
- [几何](#) - 提供绘制2D几何图形的实用工具。
- [布局](#) - 推导定位元素的辅助数据。
- [比例尺](#) - 数据编码和视觉编码之间转换。
- [可缩放矢量图形](#) - 提供用于创建可伸缩矢量图形的实用工具。
- [时间](#) - 解析或格式化时间，计算日历的时间间隔等。
- [行为](#) - 可重用交互行为

[d3 \(核心函数\)](#)

[选择](#)

- [d3.event](#) - 访问用于交互的当前用户事件。
- [d3.mouse](#) - 获取相对于指定容器的鼠标位置。
- [d3.select](#) - 从当前文档中选择一个元素。
- [d3.selectAll](#) - 从当前文档中选择多个元素。
- [d3.selection](#) - 增强选择器原型，或测试实例类型。
- [d3.touch](#) - 获取相对于指定容器的单点触摸位置。
- [d3.touches](#) - 获取相对于指定容器的多点触摸位置。
- [selection.append](#) - 创建并追加一个新元素。
- [selection.attr](#) - 取得或设置属性的值。
- [selection.call](#) - 为当前选择调用一个函数。
- [selection.classed](#) - 添加或移除CSS类。
- [selection.data](#) - 在计算相关的连接时，取得或设置一组元素的数据。
- [selection.datum](#) - 取得或设置单个元素的数据，不必计算连接。
- [selection.each](#) - 为每个选中的元素调用一个函数。
- [selection.empty](#) - 如果选择是空则返回true。
- [selection.enter](#) - 为缺失的元素返回占位符。
- [selection.exit](#) - 返回不再需要的元素。
- [selection.filter](#) - 基于数据过滤选择。
- [selection.html](#) - 取得或设置innerHTML内容。
- [selection.insert](#) - 在已存在元素之前创建并插入一个元素。
- [selection.interrupt](#) - 如果有过渡的话，立即中断当前的过渡。
- [selection.node](#) - 返回选择中的第一个节点。
- [selection.on](#) - 为交互添加或移除事件监听器。
- [selection.order](#) - 重排列文档中的元素，以匹配选择。

- [selection.property](#) - 取得或设置行内属性。
- [selection.remove](#) - 从当前文档中移除当前元素。
- [selection.select](#) - 为每个选中元素的在选择一个后代元素。
- [selection.selectAll](#) - 为每个选中元素的在选择多个后代元素。
- [selection.size](#) - 返回选择中的元素数。
- [selection.sort](#) - 基于数据排列文档中的元素。
- [selection.style](#) - 取得或设置样式属性。
- [selection.text](#) - 取得或设置文本内容。
- [selection.transition](#) - 在选中元素上开启过渡。

过渡

- [d3.ease](#) - 自定义过渡时间。
- [d3.timer](#) - 开启一段自定义动画定时器。
- [d3.interpolate](#) - 插补两个值。
- [d3.interpolateArray](#) - 插补两个数组。
- [d3.interpolateHcl](#) - 插补两个HCL颜色值。
- [d3.interpolateHsl](#) - 插补两个HSL颜色值。
- [d3.interpolateLab](#) - 插补两个L*a*b*颜色值。
- [d3.interpolateNumber](#) - 插补两个数字值。
- [d3.interpolateObject](#) - 插补两个任意对象。
- [d3.interpolateRgb](#) - 插补两个RGB颜色值。
- [d3.interpolateRound](#) - 插补两个整数。
- [d3.interpolateString](#) - 插补两个字符串。
- [d3.interpolateTransform](#) - 插补两个2D矩阵变换。
- [d3.interpolateZoom](#) - 在两个点之间平滑地缩放平移。
- [d3.interpolators](#) - 注册一个自定义的插值器。
- [d3.timer.flush](#) - 立即执行一个0延迟的定时器。
- [d3.transition](#) - 开启一个动画过渡。
- [ease](#) - 一个参数化的缓动函数。
- [interpolate](#) - 一个参数化的插值器函数。
- [transition.attr](#) - 平滑地过渡到一个新的属性值。
- [transition.attrTween](#) - 在两个属性值之间平滑地过渡。
- [transition.call](#) - 为当前的过渡调用一个函数。
- [transition.delay](#) - 指定每个元素的延迟时间（以毫秒为单位）。
- [transition.duration](#) - 指定每个元素的持续时间（以毫秒为单位）。
- [transition.each](#) - 为过渡结束时间添加一个监听器。
- [transition.ease](#) - 指定一个过渡的缓动函数。
- [transition.empty](#) - 如果过渡是空则返回true。
- [transition.filter](#) - 基于数据过滤一个过渡。
- [transition.node](#) - 返回过渡中的第一个节点。
- [transition.remove](#) - 在过渡的最后移除选中的元素。
- [transition.select](#) - 为每个选中的元素在一个子元素开启一段过渡。
- [transition.selectAll](#) - 为每个选中的元素在多个子元素开启一段过渡。
- [transition.size](#) - 返回在选择中元素的数量。
- [transition.style](#) - 平滑地过渡到一个新的样式值。
- [transition.styleTween](#) - 在两个样式属性值之间平滑地过渡。
- [transition.text](#) - 在过渡开始时设置文本内容。
- [transition.transition](#) - 当这次过渡结束时，在相同的元素上开启另一段过渡。
- [transition.tween](#) - 指定一个自定义的补间操作符作为过渡的一部分运行。

数组

- [d3.ascending](#) - 为排序比较两个值。
- [d3.bisectLeft](#) - 在排序数组中检索值。
- [d3.bisector](#) - 二等分使用访问器或比较器。
- [d3.bisectRight](#) - 在排序数组中检索值。
- [d3.bisect](#) - 在排序数组中检索值。
- [d3.descending](#) - 为排序比较两个值。
- [d3.deviation](#) - 计算一组数据的标准差。
- [d3.entries](#) - 列出一个关联数组的键值对实体。
- [d3.extent](#) - 找出一个数组中的最大值和最小值。
- [d3.keys](#) - 列出一个关联数组中的键。
- [d3.map](#) - 构建一个新的map。
- [d3.max](#) - 找出一个数组中的最大值。
- [d3.mean](#) - 计算一组数据的算数平均值。
- [d3.median](#) - 计算一组数据的算数中值。
- [d3.merge](#) - 合并多个数组为一个数组。
- [d3.min](#) - 找出一个数组中的最小值。
- [d3.nest](#) - 分层地分组数组元素。
- [d3.pairs](#) - 返回一个元素的相邻对数组。
- [d3.permute](#) - 按照数组的索引重新排序数组元素。
- [d3.quantile](#) - 为一个排好序的数字数组的分位数。
- [d3.range](#) - 产生一系列的数值。
- [d3.set](#) - 构建一个新的集合。
- [d3.shuffle](#) - 随机化一个数组的顺序。
- [d3.sum](#) - 计算数字数组的和。
- [d3.transpose](#) - 转置一个数组的数组。
- [d3.values](#) - 列出关联数组的值。
- [d3.variance](#) - 计算数字数组的方差。
- [d3.zip](#) - 转置数组的可变数量。
- [map.empty](#) - 如果map不包含元素就返回true。
- [map.entries](#) - 返回map的实体数组。
- [map.forEach](#) - 为每个指定的实体调用一个函数。
- [map.get](#) - 为指定的键返回值。
- [map.has](#) - 如果map包含指定的值则返回true。
- [map.keys](#) - 返回map的键数组。
- [map.remove](#) - 为指定的键移除值。
- [map.set](#) - 为指定的键设置值。
- [map.size](#) - 返回map的实体数量。
- [map.values](#) - 返回map的值数组。
- [nest.entries](#) - 返回一组键-值元组
- [nest.key](#) - 在嵌套层级中添加一个级别。
- [nest.map](#) - 返回一个关联数组。
- [nest.rollup](#) - 为叶子值指定一个汇总函数。
- [nest.sortKeys](#) - 按照键排序叶子嵌套级别。
- [nest.sortValues](#) - 按照值排序叶子嵌套级别。
- [set.add](#) - 添加指定的值。
- [set.empty](#) - 如果集合不含元素的话返回true。
- [set.forEach](#) - 为集合中的每个元素调用指定的函数。
- [set.has](#) - 如果集合中包含指定值就返回true。
- [set.remove](#) - 移除指定的值。
- [set.size](#) - 返回集合中的元素数量。

- [set.values](#) - 返回集合中的值数组。

数学

- [d3.random.bates](#) - 生成具有贝茨分布规律的随机数。
- [d3.random.irwinHall](#) - 生成具有Irwin–Hall分布规律的随机数。
- [d3.random.logNormal](#) - 生成具有对数正态分布规律的随机数。
- [d3.random.normal](#) - 生成具有正态分布规律的随机数。
- [d3.transform](#) - 计算2D仿射变换的标准形式。

请求

- [d3.csv](#) - 请求一个CSV（逗号分隔值）的文件。
- [d3.html](#) - 请求一个HTML文档片段。
- [d3.json](#) - 请求一个JSON对象。
- [d3.text](#) - 请求一个text文件。
- [d3.tsv](#) - 请求一个TSV（制表符分隔值）的文件。
- [d3.xhr](#) - 使用XMLHttpRequest请求一个资源。
- [d3.xml](#) - 请求一个XML文档片段。
- [xhr.abort](#) - 终止未完成的请求。
- [xhr.get](#) - 发送一个GET请求。
- [xhr.header](#) - 设置一个请求头。
- [xhr.mimeType](#) - 设置一个接受请求头并覆盖响应的MIME类型。
- [xhr.on](#) - 为“progress”，“load”或“error”事件添加一个事件监听器。
- [xhr.post](#) - 发送一个POST请求。
- [xhr.response](#) - 设置一个响应映射函数。
- [xhr.send](#) - 使用指定的数据和函数发送一个请求。

格式化

- [d3.format](#) - 将一个数组格式化为字符串。
- [d3.formatPrefix](#) - 为指定的值和精度返回SI前缀。
- [d3.regquote](#) - 将字符串转义为正则表达式。
- [d3.round](#) - 将值四舍五入到指定小数位。

CSV格式化(d3.csv)

- [d3.csv.formatRows](#) - 格式化一组元组为CSV字符串。
- [d3.csv.format](#) - 格式化一组对象为CSV字符串。
- [d3.csv.parseRows](#) - 解析CSV字符串为元组，忽略首行。
- [d3.csv.parse](#) - 把首行数据CSV字符串解析为对象。
- [d3.csv](#) - 请求一个CSV文件。
- [d3.dsv](#) - 为指定的分隔符和mime类型创建一个解析器/格式化器。
- [d3.tsv.formatRows](#) - 格式化一组元组为TSV字符串。
- [d3.tsv.format](#) - 格式化一组对象为TSV字符串。
- [d3.tsv.parseRows](#) - 解析TSV字符串为元组，忽略首行。
- [d3.tsv.parse](#) - 把首行数据TSV字符串解析为对象。
- [d3.tsv](#) - 请求一个TSV文件。

本地化

- [d3.locale](#) - 使用指定的字符串创建一个本地化。
- [locale.numberFormat](#) - 创建一个新的数字格式化器。
- [locale.timeFormat](#) - 创建一个新的时间格式化器/解析器。

颜色

- [d3.hcl](#) - 指定一种颜色，创建一个HCL颜色对象。
- [d3.hsl](#) - 指定一种颜色，创建一个HSL颜色对象。
- [d3.lab](#) - 指定一种颜色，创建一个L*a*b*颜色对象。
- [d3.rgb](#) - 指定一种颜色，创建一个RGB颜色对象。
- [hcl.brighter](#) - 增强颜色的亮度，变化幅度由参数决定。
- [hcl.darker](#) - 减弱颜色的亮度，变化幅度由参数决定。
- [hcl.rgb](#) - 将HCL颜色对象转化成RGB颜色对象。
- [hcl.toString](#) - HCL颜色对象转化为字符串格式。
- [hsl.brighter](#) - 增强颜色的亮度，变化幅度由参数决定。
- [hsl.darker](#) - 减弱颜色的亮度，变化幅度由参数决定。
- [hsl.rgb](#) - 将HSL颜色对象转化成RGB颜色对象。
- [hsl.toString](#) - 将HSL颜色对象转化为字符串格式。
- [lab.brighter](#) - 增强颜色的亮度，变化幅度由参数决定。
- [lab.darker](#) - 减弱颜色的亮度，变化幅度由参数决定。
- [lab.rgb](#) - 将L*a*b*颜色对象转化成RGB颜色对象。
- [lab.toString](#) - 将L*a*b*颜色对象转化为字符串格式。
- [rgb.brighter](#) - 增强颜色的亮度，变化幅度由参数决定。
- [rgb.darker](#) - 减弱颜色的亮度，变化幅度由参数决定。
- [rgb.hsl](#) - 将RGB颜色对象转化成HSL颜色对象。
- [rgb.toString](#) - 将RGB颜色对象转化为字符串格式。

命名空间

- [d3.ns.prefix](#) - 访问或扩展已知的XML命名空间。
- [d3.ns.qualify](#) - 限定一个前缀名称，例如"xlink:href"。

内部

- [d3.dispatch](#) - 创建一个定制的事件分发器。
- [d3.functor](#) - 创建一个函数并返回一个常量。
- [d3.rebind](#) - 重新绑定get/set方法到一个子类上。
- [dispatch.on](#) - 注册或者解除注册事件监听器。
- [dispatch.type](#) - 为指定的监听器分发事件。

d3.scale (比例尺)

数值比例尺

- [d3.scale.identity](#) - 构建一个线性恒等比例尺。
- [d3.scale.linear](#) - 构建一个线性比例尺。
- [d3.scale.log](#) - 构建一个对数比例尺。
- [d3.scale.pow](#) - 构建一个指数比例尺。
- [d3.scale.quantile](#) - 构建一个分位数比例尺。
- [d3.scale.quantize](#) - 构建一个量化比例尺（值域离散）。
- [d3.scale.sqrt](#) - 构建一个平方根比例尺。
- [d3.scale.threshold](#) - 构建一个临界值比例尺（值域离散）。
- [identity.copy](#) - 复制比例尺。
- [identity.domain](#) - 取得或设置比例尺的定义域。
- [identity.invert](#) - 等价于恒等函数。
- [identity.range](#) - 等价于identity.domain。
- [identity.tickFormat](#) - 获取一个用来展示刻度值得格式化器。

- [identity.ticks](#) - 取得定义域中典型的值。
- [identity](#) - 恒等函数。
- [linear.clamp](#) - 启用或者关闭值域的闭合。
- [linear.copy](#) - 复制比例尺。
- [linear.domain](#) - 取得或设置比例尺的定义域。
- [linear.interpolate](#) - 取得或设置输出插值器。
- [linear.invert](#) - 取得输出值对应的输入值。
- [linear.nice](#) - 扩展比例尺的定义域为一个优化的定义域。
- [linear.rangeRound](#) - 设置比例尺的输出范围，并四舍五入。
- [linear.range](#) - 取得或设置比例尺的输出范围。
- [linear.tickFormat](#) - 获取一个用来展示刻度值得格式化器。
- [linear.ticks](#) - 取得定义域中典型的值。
- [linear](#) - 取得输入值对应的输出值。
- [log.clamp](#) - 启用或者关闭值域的闭合。
- [log.copy](#) - 复制比例尺。
- [log.domain](#) - 取得或设置比例尺的定义域。
- [log.interpolate](#) - 取得或设置输出插值器。
- [log.invert](#) - 取得输出值对应的输入值。
- [log.nice](#) - 扩展比例尺的定义域为一个优化的10的次方。
- [log.rangeRound](#) - 设置比例尺的输出范围，并四舍五入。
- [log.range](#) - 取得或设置比例尺的输出范围。
- [log.tickFormat](#) - 获取一个用来展示刻度值得格式化器。
- [log.ticks](#) - 取得定义域中典型的值。
- [log](#) - 取得输入值对应的输出值。
- [pow.clamp](#) - 启用或者关闭值域的闭合。
- [pow.copy](#) - 复制比例尺。
- [pow.domain](#) - 取得或设置比例尺的定义域。
- [pow.exponent](#) - 取得或设置指数。
- [pow.interpolate](#) - 取得或设置输出插值器。
- [pow.invert](#) - 取得输出值对应的输入值。
- [pow.nice](#) - 扩展比例尺的定义域为一个优化的定义域。
- [pow.rangeRound](#) - 设置scale的输出范围，并四舍五入。
- [pow.range](#) - 取得或设置比例尺的值域。
- [pow.tickFormat](#) - 获取一个用来展示刻度值得格式化器。
- [pow.ticks](#) - 取得定义域中典型的值。
- [pow](#) - 取得输出值对应的输入值。
- [quantile.copy](#) - 复制比例尺。
- [quantile.domain](#) - 取得或设置比例尺的定义域（离散的值）。
- [quantile.invertExtent](#) - 取得输出值对应的输入值。
- [quantile.quantiles](#) - 取得比例尺的分位数箱阈值。
- [quantile.range](#) - 取得或设置比例尺的值域（离散的值）。
- [quantile](#) - 取得输入值对应的输出值。
- [quantize.copy](#) - 复制比例尺。
- [quantize.domain](#) - 取得或设置比例尺的定义域。
- [quantize.invertExtent](#) - 取得输出值对应的输入值。
- [quantize.range](#) - 取得或设置比例尺的值域（离散的值）。
- [quantize](#) - 取得输入值对应的输出值。
- [threshold.copy](#) - 复制比例尺。
- [threshold.domain](#) - 取得或设置比例尺的定义域。
- [threshold.invertExtent](#) - 取得输出值对应的输入值。

- [threshold.range](#) - 取得或设置比例尺的值域（离散的值）。
- [threshold](#) - 取得输入值对应的输出值。

序数比例尺

- [d3.scale.ordinal](#) - 构造一个序数比例尺。
- [ordinal](#) - 获取输入值对应的输出值。
- [ordinal.domain](#) - 获取或指定比例尺的输入域。
- [ordinal.range](#) - 获取或指定比例尺的输出范围。
- [\[ordinal.rangePoints\]](#) (序数比例尺#ordinal_rangePoints) - 指定输出范围为连续区间。
- [ordinal.rangeRoundPoints](#) - 指定输出范围为连续区间，刻度点均为整数。
- [ordinal.rangeBands](#) - 指定输出范围为连续区间。
- [ordinal.rangeRoundBands](#) - 指定输出范围为连续区间，区间段的起点均为整数。
- [\[ordinal.rangeBand\]](#) (序数比例尺#ordinal_rangeBand) - 获取区间段的宽度。
- [\[ordinal.rangeExtent\]](#) (序数比例尺#ordinal_rangeExtent) - 获取当前比例尺的输出范围，未被切分的。
- [ordinal.copy](#) - 深度拷贝当前比例尺对象。
- [d3.scale.category10](#) - 构造一个有10种颜色的序数比例尺。
- [d3.scale.category20](#) - 构造一个有20种颜色的序数比例尺。
- [d3.scale.category20b](#) - 构造一个另外20种颜色的序数比例尺。
- [d3.scale.category20c](#) - 构造一个另外20种颜色的序数比例尺。

SVG函数

形状

- [arc.centroid](#) - 计算弧中心。
- [arc.cornerRadius](#) - 获取或设置拐角（corner）半径访问器。
- [arc.endAngle](#) - 获取或设置结束角度访问器。
- [arc.innerRadius](#) - 获取或设置内半径访问器。
- [arc.outerRadius](#) - 获取或设置外半径访问器。
- [arc.padAngle](#) - 获取或设置填补（pad）角度访问器。
- [arc.padRadius](#) - 获取或设置填补（pad）半径访问器。
- [arc.startAngle](#) - 获取或设置开始角度访问器。
- [arc](#) - 生成一个像饼图或圆环图中的固定弧度。
- [area.angle](#) - 获取或设置角度访问器。
- [area.defined](#) - 控制面积在给定点是否有定义的。
- [area.defined](#) - 控制径向面积在给定点是否有定义的。
- [area.endAngle](#) - 获取或设置角度（顶线）访问器。
- [area.innerRadius](#) - 获取或设置内半径（基线）访问器。
- [area.interpolate](#) - 获取或设置插值模式。
- [area.outerRadius](#) - 获取或设置外半径（顶线）访问器。
- [area.radius](#) - 获取或设置半径访问器。
- [area.startAngle](#) - 获取或设置角度（基线）访问器。
- [area.tension](#) - 获取或设置基本样条线的张力。
- [area.x0](#) - 获取或设置x0-坐标（基线）访问器。
- [area.x1](#) - 获取或设置x1-坐标（顶线）访问器。
- [area.x](#) - 获取或设置x-坐标访问器。
- [area.y0](#) - 获取或设置y0-坐标（基线）访问器。
- [area.y1](#) - 获取或设置y1-坐标（顶线）访问器。
- [area.y](#) - 获取或设置y-坐标访问器。
- [area](#) - 生成一个像面积图中的分段线性面积。
- [area](#) - 生成一个像极坐标面积图中的分段线性面积。

- [chord.endAngle](#) - 获取或设置圆弧结束角访问器。
- [chord.radius](#) - 获取或设置圆弧半径访问器。
- [chord.source](#) - 获取或设置圆弧来源圆弧访问器。
- [chord.startAngle](#) - 获取或设置圆弧开始角访问器。
- [chord.target](#) - 获取或设置目标圆弧访问器。
- [chord](#) - 在弦图中生成一个二次贝塞尔曲线连接两个弧。
- [d3.svg.arc](#) - 新建一个弧度生成器。
- [d3.svg.area.radial](#) - 新建一个径向面积生成器。
- [d3.svg.area](#) - 新建一个面积生成器。
- [d3.svg.chord](#) - 新建一个弦生成器。
- [d3.svg.diagonal.radial](#) - 新建一个径向对角线生成器。
- [d3.svg.diagonal](#) - 新建一个对角线生成器。
- [d3.svg.line.radial](#) - 新建一个径向线生成器。
- [d3.svg.line](#) - 新建一个线生成器。
- [d3.svg.symbolTypes](#) - 一组符号类型。
- [d3.svg.symbol](#) - 新建一个符号生成器。
- [diagonal.projection](#) - 设置或获取一个可选的点转换。
- [diagonal.source](#) - 设置或获取源点访问器。
- [diagonal.target](#) - 设置或获取目标点访问器。
- [diagonal](#) - 生成一个像节点链接图中的二维贝塞尔连接器。
- [diagonal](#) - 生成一个像节点链接图中的二维径向贝塞尔连接器。
- [line.angle](#) - 设置或获取角度 accessor。
- [line.defined](#) - 控制线在给定点是否是定义的。
- [line.defined](#) - 控制径向线在给定点是否是定义的。
- [line.interpolate](#) - 设置或获取插值模式。
- [line.interpolate](#) - 设置或获取径向弦的插值模式。
- [line.radius](#) - 设置或获取半径访问器。
- [line.tension](#) - 设置或获取径向基本样条线的张力。
- [line.tension](#) - 设置或获取基本样条线的张力。
- [line.x](#) - 设置或获取x-坐标访问器。
- [line.y](#) - 设置或获取y-坐标访问器。
- [line](#) - 生成一个像线图的分段线段。
- [line](#) - 生成一个像极线图的分段线段。
- [symbol.size](#) - 设置或获取符号尺寸（平方像素）访问器。
- [symbol.type](#) - 设置或获取符号类型访问器。
- [symbol](#) - 生成一个像散点图中的符号。

轴

- [axis.innerTickSize](#) - 指定内刻度大小。
- [axis.orient](#) - 设置或者取得轴的方向。
- [axis.outerTickSize](#) - 指定外刻度大小。
- [axis.scale](#) - 设置或者取得比例尺。
- [axis.tickFormat](#) - 重载标签的刻度格式化。
- [axis.tickPadding](#) - 指定刻度和刻度标签之间的间距。
- [axis.tickSize](#) - 指定主要的次要的和尾部的刻度。
- [axis.ticks](#) - 控制轴的刻度如何生成。
- [axis.tickValues](#) - 明确地指定刻度值。
- [axis](#) - 为给定的选择器或过渡创建或者更新轴。
- [d3.svg.axis](#) - 创建一个新的轴生成器。

刷子

- [brush.clear](#) - 重置拖选范围。
- [brush.empty](#) - 拖选是否为空。
- [brush.event](#) - 在设置范围之后分发拖选事件。
- [brush.extent](#) - 拖选范围可以是0,1, 2维的。
- [brush.on](#) - 监听拖选何时改变。
- [brush.x](#) - 拖选的x-比例, 用于水平拖选。
- [brush.y](#) - 拖选的y-比例, 用于垂直拖选。
- [brush](#) - 将拖选应用在指定的选择器和过渡上。
- [d3.svg.brush](#) - 点击和拖曳来选择1个1维或2维区域。

[d3.time \(时间\)](#)

[时间格式化](#)

- [d3.time.format.iso](#) - ISO 8601 UTC时间格式化器。
- [d3.time.format.multi](#) - 创建一个新的本地多功能时间格式化器。
- [d3.time.format.utc](#) - 由指定的限定符创建一个新的UTC时间格式化器。
- [d3.time.format](#) - 由指定的限定符创建一个新的本地时间格式化器。
- [format.parse](#) - 将字符串解析为时间对象。
- [format](#) - 将一个时间对象格式化为一个字符串。

[时间比例尺](#)

- [d3.time.scale](#) - 构造一个线性时间比例尺。
- [scale.clamp](#) - 指定输出范围是否闭合。
- [scale.copy](#) - 创建比例尺的副本。
- [scale.domain](#) - 取得或设置比例尺的定义域。
- [scale.interpolate](#) - 取得或设置比例尺的输出插值器。
- [scale.invert](#) - 取得给定输出值对应定义域中的值。
- [scale.nice](#) - 扩展比例尺的定义域为一个优化的整数值。
- [scale.rangeRound](#) - 设置比例尺的四舍五入输出范围。
- [scale.range](#) - 取得或设置比例尺的输出范围。
- [scale.tickFormat](#) - 取得用于展示刻度值的格式化器。
- [scale.ticks](#) - 取得定义域中有代表性的值。
- [scale](#) - 取得给定定义域中值对应的输出范围中的值。

[时间间隔](#)

- [d3.time.dayOfYear](#) - 计算天数。
- [d3.time.days](#) - day.range的别名。
- [d3.time.day](#) - 每天 (12:00 AM) 。
- [d3.time.fridayOfYear](#) - 计算基于周五的星期数。
- [d3.time.fridays](#) - friday.range的别名。
- [d3.time.friday](#) - 每周五 (例如February 5, 12:00 AM) 。
- [d3.time.hours](#) - hour.range的别名。
- [d3.time.hour](#) - 每小时 (例如, 1:00 AM) 。
- [d3.time.interval](#) - 一个基于本地时间的时间间隔。
- [d3.time.minutes](#) - minute.range的别名。
- [d3.time.minute](#) - 每分钟 (例如, 1:02 AM) 。
- [d3.time.mondayOfYear](#) - 计算基于周一的星期数。
- [d3.time.mondays](#) - monday.range的别名。
- [d3.time.monday](#) - 每周一 (例如, February 5, 12:00 AM) 。
- [d3.time.months](#) - month.range的别名。

- [d3.time.month](#) - 每个月（例如, February 1, 12:00 AM）
- [d3.time.saturdayOfYear](#) - 计算基于周六的星期数。
- [d3.time.saturdays](#) - saturday.range的别名。
- [d3.time.saturday](#) - every Saturday（例如, February 5, 12:00 AM）。
- [d3.time.seconds](#) - second.range的别名。
- [d3.time.second](#) - 每秒（例如, 1:02:03 AM）。
- [d3.time.sundayOfYear](#) - 计算基于周日的星期数。
- [d3.time.sundays](#) - sunday.range的别名。
- [d3.time.sunday](#) - 每周日（例如February 5, 12:00 AM）。
- [d3.time.thursdayOfYear](#) - 计算基于周四的星期数。
- [d3.time.thursdays](#) - thursday.range的别名。
- [d3.time.thursday](#) - 每周四（例如February 5, 12:00 AM）。
- [d3.time.tuesdayOfYear](#) - 计算基于周二的星期数。
- [d3.time.tuesdays](#) - tuesday.range的别名。
- [d3.time.tuesday](#) - 每周二（例如February 5, 12:00 AM）。
- [d3.time.wednesdayOfYear](#) - 计算基于周三的星期数。
- [d3.time.wednesdays](#) - wednesday.range的别名。
- [d3.time.wednesday](#) - 每周三（例如February 5, 12:00 AM）。
- [d3.time.weekOfYear](#) - sundayOfYear的别名。
- [d3.time.weeks](#) - sunday.range的别名。
- [d3.time.week](#) - sunday的别名。
- [d3.time.years](#) - year.range的别名。
- [d3.time.year](#) - 每年（例如January 1, 12:00 AM）。
- [interval.ceil](#) - 上取整到最近的时间间隔。
- [interval.floor](#) - 下取整到最近的时间间隔。
- [interval.offset](#) - 基于一些间隔返回时间偏移。
- [interval.range](#) - 返回指定范围中的日期。
- [interval.round](#) - 四舍五入到最近的时间间隔。
- [interval.utc](#) - 返回UTC时间间隔。
- [interval](#) - interval.floor的别名。

[d3.layout \(布局\)](#)

[捆布局](#)

- [bundle](#) - 对边使用Holten *层次捆绑* 算法。
- [d3.layout.bundle](#) - 构造一个新的默认的捆绑布局。

[弦布局](#)

- [chord.chords](#) - 取回计算的弦角度。
- [chord.groups](#) - 取回计算的分组角度。
- [chord.matrix](#) - 取得或设置布局需要的矩阵数据。
- [chord.padding](#) - 取得或设置弦片段间的角填充。
- [chord.sortChords](#) - 取得或设置用于弦的比较器（Z轴顺序）。
- [chord.sortGroups](#) - 取得或设置用于分组的比较器。
- [chord.sortSubgroups](#) - 取得或设置用于子分组的比较器。
- [d3.layout.chord](#) - 从关系矩阵生成一个弦图。

[簇布局](#)

- [cluster.children](#) - 取得或者设置子节点的访问器函数。
- [cluster.links](#) - 技术树节点之间的父子连接。

- [cluster.nodeSize](#) - 为每个节点指定固定的尺寸。
- [cluster.nodes](#) - 计算簇布局并返回节点数组。
- [cluster.separation](#) - 取得或设置邻接节点的分隔函数。
- [cluster.size](#) - 取得或设置布局的尺寸。
- [cluster.sort](#) - 取得或设置兄弟节点的比较器函数。
- [cluster](#) - cluster.nodes的别名。
- [d3.layout.cluster](#) - 将实体聚集成树状图。

力布局

- [d3.layout.force](#) - 使用物理模拟排放链接节点的位置。
- [force.alpha](#) - 取得或者设置力布局的冷却参数。
- [force.chargeDistance](#) - 取得或者设置最大电荷距离。
- [force.charge](#) - 取得或者设置电荷强度。
- [force.drag](#) - 给节点绑定拖动行为。
- [force.friction](#) - 取得或者设置摩擦系数。
- [force.gravity](#) - 取得或者设置重力强度。
- [force.linkDistance](#) - 取得或者设置链接距离。
- [force.linkStrength](#) - 取得或者设置链接强度。
- [force.links](#) - 取得或者设置节点间的链接数组。
- [force.nodes](#) - 取得或者设置布局的节点数组。
- [force.on](#) - 监听在计算布局位置时的更新。
- [force.resume](#) - 重新加热冷却参数，并重启模拟。
- [force.size](#) - 取得或者设置布局大小。
- [force.start](#) - 当节点变化时启动或者重启模拟。
- [force.stop](#) - 立即停止模拟。
- [force.theta](#) - 取得或者设置电荷作用的精度。
- [force.tick](#) - 运行布局模拟的一步。

层次布局

- [d3.layout.hierarchy](#) - 派生一个定制的层次布局实现。
- [hierarchy.children](#) - 取得或设置子节点的访问器。
- [hierarchy.links](#) - 计算树节点中的父子链接。
- [hierarchy.nodes](#) - 计算层次布局并返回节点数组。
- [hierarchy.revalue](#) - 重新计算层次值。
- [hierarchy.sort](#) - 取得或设置兄弟节点的比较器函数。
- [hierarchy.value](#) - 取得或设置值访问器函数。
- [hierarchy](#) - hierarchy.nodes的别名。

直方图布局

- [d3.layout.histogram](#) - 构造一个新的默认的直方图布局。
- [histogram.bins](#) - 指定值是如何组织到箱中的。
- [histogram.frequency](#) - 按频数或者频率计算分布。
- [histogram.range](#) - 取得或设置值得范围。
- [histogram.value](#) - 取得或设置值访问器。
- [histogram](#) - 使用量化的箱计算数据的分布。

包布局

- [d3.layout.pack](#) - 用递归的圆-包生成一个层次布局。
- [pack.children](#) - 取得或设置子节点的访问器。
- [pack.links](#) - 计算树节点中的父子链接。

- [pack.nodes](#) - 计算包布局并返回节点数组。
- [pack.padding](#) - 指定布局间距（以像素为单位）
- [pack.radius](#) - 指定节点半径（不是由值派生来的）
- [pack.size](#) - 指定布局尺寸。
- [pack.sort](#) - 控制兄弟节点的遍历顺序。
- [pack.value](#) - 取得或设置用于圆尺寸的值访问器。
- [pack](#) - pack.nodes的别名。

分区布局

- [d3.layout.partition](#) - 递归地将节点树分区为旭日图或者冰柱图。
- [partition.children](#) - 取得或设置孩子访问器。
- [partition.links](#) - 计算树节点中的父子链接。
- [partition.nodes](#) - 计算分区布局并返回节点数组。
- [partition.size](#) - 指定布局的尺寸。
- [partition.sort](#) - 控制兄弟节点的遍历顺序。
- [partition.value](#) - 取得或设置用来指定圆尺寸的值访问器。
- [partition](#) - partition.nodes的别名。

饼布局

- [d3.layout.pie](#) - 构造一个新的默认的饼布局。
- [pie.endAngle](#) - 取得或设置饼布局整体的结束角度。
- [pie.padAngle](#) - 取得或设置饼布局填充角度。
- [pie.sort](#) - 控制饼片段的顺时针方向的顺序。
- [pie.startAngle](#) - 取得或设置饼布局整体的开始角度。
- [pie.value](#) - 取得或设置值访问器函数。
- [pie](#) - 计算饼图或圆环图中弧的开始和结束角度。

堆叠布局

- [d3.layout.stack](#) - 构造一个新的默认的堆叠布局。
- [stack.offset](#) - 指定整体的基线算法。
- [stack.order](#) - 控制每个系列的顺序。
- [stack.out](#) - 取得或设置用于存储基线的输出函数。
- [stack.values](#) - 取得或设置每个系列的值访问器函数。
- [stack.x](#) - 取得或设置x-维访问器函数。
- [stack.y](#) - 取得或设置y-维访问器函数。
- [stack](#) - 计算堆叠图或者面积图的基线。

树布局

- [d3.layout.tree](#) - 整齐地排列树节点。
- [tree.children](#) - 取得或设置孩子访问器。
- [tree.links](#) - 计算树节点的父-子连接。
- [tree.nodeSize](#) - 为每个节点指定一个固定的尺寸。
- [tree.nodes](#) - 计算父布局并返回一组节点。
- [tree.separation](#) - 取得或设置相邻节点的间隔函数。
- [tree.size](#) - 用x和y指定树的尺寸。
- [tree.sort](#) - 控制遍历顺序中兄弟节点的顺序。
- [tree](#) - tree.nodes的别名。

矩形树布局

- [d3.layout.treemap](#) - 使用空间递归分区算法展示树的节点。

- [treemap.children](#) - 取得或设置孩子访问器。
- [treemap.links](#) - 计算树节点中的父子链接。
- [treemap.mode](#) - 改变布局的算法。
- [treemap.nodes](#) - 计算矩形树布局并返回节点数组。
- [treemap.padding](#) - 指定父子之间的间距。
- [treemap.round](#) - 启用或者禁用四舍五入像素值。
- [treemap.size](#) - 指定布局的尺寸。
- [treemap.sort](#) - 控制兄弟节点的遍历顺序。
- [treemap.sticky](#) - 让布局对稳定的更新是粘滞的 (sticky) 。
- [treemap.value](#) - 取得或设置用来指定矩形树中矩形单元尺寸的值访问器。
- [treemap](#) - treemap.nodes的别名。

d3.geo (地理)

地理路径

- [circle.angle](#) - 指定角半径 (以度为单位) 。
- [circle.origin](#) - 指定经纬度原点。
- [circle.precision](#) - 指定分段圆的精度。
- [circle](#) - 生成一个分段圆。
- [d3.geo.area](#) - 计算给定要素的球体面积。
- [d3.geo.bounds](#) - 计算给定要素的经纬度边界框。
- [d3.geo.centroid](#) - 计算给定要素的球体中心。
- [d3.geo.circle](#) - 创建一个圆生成器。
- [d3.geo.distance](#) - 计算两点之间的大弧距离。
- [d3.geo.graticule](#) - 创建一个经纬网生成器。
- [d3.geo.interpolate](#) - 两个点之间插入一个大弧。
- [d3.geo.length](#) - 计算线的长度或多边形的面积。
- [d3.geo.path](#) - 创建一个地理路径生成器。
- [d3.geo.rotation](#) - 为指定的角度 $[\lambda, \phi, \gamma]$ 创建一个旋转角度。
- [graticule.extent](#) - 取得或设置major & minor范围。
- [graticule.lines](#) - 为经线和纬线生成线数组。
- [graticule.majorExtent](#) - 取得或设置major范围。
- [graticule.majorStep](#) - 取得或设置major步长间隔。
- [graticule.minorExtent](#) - 取得或设置minor范围。
- [graticule.minorStep](#) - 取得或设置minor步长间隔。
- [graticule.outline](#) - 生成格子线范围的一个多边形。
- [graticule.precision](#) - 取得或设置纬度精度。
- [graticule.step](#) - 取得或设置major & minor步长间隔。
- [graticule](#) - 生成经纬线的多线要素。
- [path.area](#) - 计算给定要素的投影面积。
- [path.bounds](#) - 计算给定要素的投影边界。
- [path.centroid](#) - 计算给定要素的投影中心。
- [path.context](#) - 取得或设置渲染上下文。
- [path.pointRadius](#) - 取得或设置点要素的半径。
- [path.projection](#) - 取得或设置地理投影。
- [path](#) - 投影指定的要素并渲染上下文。
- [rotation.invert](#) - 反旋转球体周围的给定位置。
- [rotation](#) - 旋转球体周围的给定位置。

地理投影

- [albers.parallels](#) - 取得或者设置投影的两条标准平行线。
- [d3.geo.albersUsa](#) - 用于展示美国地图的Albers复合投影。
- [d3.geo.albers](#) - Albers等面积圆锥投影。
- [d3.geo.azimuthalEqualArea.raw](#) - 原始方位角等面积投影。
- [d3.geo.azimuthalEqualArea](#) - 方位角等面积投影。
- [d3.geo.azimuthalEquidistant.raw](#) - 原始方位角等距投影。
- [d3.geo.azimuthalEquidistant](#) - 方位角等距投影。
- [d3.geo.conicConformal.raw](#) - 原始圆锥正形投影。
- [d3.geo.conicConformal](#) - 圆锥正形投影。
- [d3.geo.conicEqualArea.raw](#) 原始圆锥等面积投影 (a.k.a. Albers)。
- [d3.geo.conicEqualArea](#) 圆锥等面积投影 (a.k.a. Albers)。
- [d3.geo.conicEquidistant.raw](#) - 原始圆锥等距投影。
- [d3.geo.conicEquidistant](#) - 圆锥等距投影。
- [d3.geo.equirectangular.raw](#) - 原始等角投影（普通圆柱投影）。
- [d3.geo.equirectangular](#) - 等角投影（普通圆柱投影）。
- [d3.geo.gnomonic.raw](#) - 原始球心投影。
- [d3.geo.gnomonic](#) - 球心投影。
- [d3.geo.mercator.raw](#) - 原始墨卡托投影。
- [d3.geo.mercator](#) - 球形墨卡托投影。
- [d3.geo.orthographic.raw](#) - 原始方位角直角投影。
- [d3.geo.orthographic](#) - 方位角直角投影。
- [d3.geo.projectionMutator](#) - 从可变的原始投影创建一个标准投影。
- [d3.geo.projection](#) - 从原始投影创建一个标准投影。
- [d3.geo.stereographic.raw](#) - 原始方位角立体投影。
- [d3.geo.stereographic](#) - 方位角立体投影。
- [d3.geo.transverseMercator.raw](#) - 原始横向墨卡托投影。
- [projection.center](#) - 取得或设置投影的中心位置。
- [projection.clipAngle](#) - get or set the radius of the projection's clip circle.
- [projection.clipExtent](#) - 取得或设置投影的视口剪切范围（以像素为单位）。
- [projection.invert](#) - 为指定的位置反转投影。
- [projection.precision](#) - 取得或设置自适应重采样的精度阈值。
- [projection.rotate](#) - 取得或设置投影的三轴旋转角。
- [projection.scale](#) - 取得或设置投影的缩放系数。
- [projection.stream](#) - 包装指定的流监听器，投影输入的几何图形。
- [projection.translate](#) - 取得或设置投影的平移位置。
- [projection](#) - 投影指定的位置。

流

- [clipExtent.extent](#) - 设置剪裁范围。
- [d3.geo.clipExtent](#) - 流转换剪切几何图形为给定的轴对齐矩形。
- [d3.geo.stream](#) - 将GeoJSON对象转换为几何流。
- [d3.geo.transform](#) - 转换流几何图形。
- [stream.lineEnd](#) - 表示线或者环的终点。
- [stream.lineStart](#) - 表示线或者环的起点。
- [stream.point](#) - 表面x, y (可选的 z) 坐标。
- [stream.polygonEnd](#) - 表明多边形的终点。
- [stream.polygonStart](#) - 表明多边形的起点。
- [stream.sphere](#) - 表明一个球体。
- [transform.stream](#) - 包装指定的流。

d3.geom (几何)

泰森多边形

- [d3.geom.voronoi](#) - 用默认的访问器创建一个泰森多边形布局。
- [voronoi.clipExtent](#) - 取得或者设置铺嵌的剪切范围。
- [voronoi.links](#) - 计算Delaunay mesh为一个链接网络。
- [voronoi.triangles](#) - 计算Delaunay mesh为一个三角形密铺。
- [voronoi.x](#) - 取得或者设置每个点的x-坐标访问器。
- [voronoi.y](#) - 取得或者设置每个点的y-坐标访问器。
- [voronoi](#) - 为每个指定的点计算泰森多边形密铺。

四叉树

- [d3.geom.quadtree](#) - 为一个点数组创建一个四叉树。
- [quadtree.add](#) - 添加点到四叉树中。
- [quadtree.find](#) - 找到四叉树中最近的点。
- [quadtree.visit](#) - 递归地遍历四叉树中的点。

多边形

- [d3.geom.polygon](#) - 由指定的点数组创建多边形。
- [polygon.area](#) - 计算多边形逆时针方向的面积。
- [polygon.centroid](#) - 计算多边形的面积中心。
- [polygon.clip](#) - 对这个多边形进行执行的多边形剪切。

凸包

- [d3.geom.hull](#) - 使用默认访问器创建一个convex hull布局。
- [hull](#) - 为给定的点数组计算convex hull。
- [hull.x](#) - 取得或设置x-坐标访问器。
- [hull.y](#) - 取得或设置y-坐标访问器。

d3.behavior (行为)

拖动

- [d3.behavior.drag](#) - 创建拖动行为。
- [drag.on](#) - 监听拖动事件。
- [drag.origin](#) - 设置拖动行为的原点。

缩放

- [d3.behavior.zoom](#) - 创建缩放行为。
 - [zoom.center](#) - 鼠标滚轮缩放的焦点。
 - [zoom.duration](#) - 取得或设置双击事件的过渡持续的时间。
 - [zoom.event](#) - 设置完缩放比例或平移之后分发缩放事件。
 - [zoom.on](#) - 事件监听器。
 - [zoom.scaleExtent](#) - 可选参数，比例因子范围。
 - [zoom.scale](#) - 当前的比例因子。
 - [zoom.size](#) - 视口的大小。
 - [zoom.translate](#) - 当前的平移偏移量。
 - [zoom.x](#) - 可选比例尺，其定义域绑定到视口的_x_范围。
 - [zoom.y](#) - 可选比例尺，其定义域绑定到视口的_y_范围。
 - [zoom](#) - 给指定的元素应用缩放行为。
-

本文参与	人员	组织	时间
翻译	太傻	VisualCrew/小组	20150801
校对/排版	太傻	VisualCrew/小组	20160331
	WeiFei365	VisualCrew/小组	2015-12-01 15:12:24
	liang42hao	VisualCrew/小组	2016-01-11 19:27:12