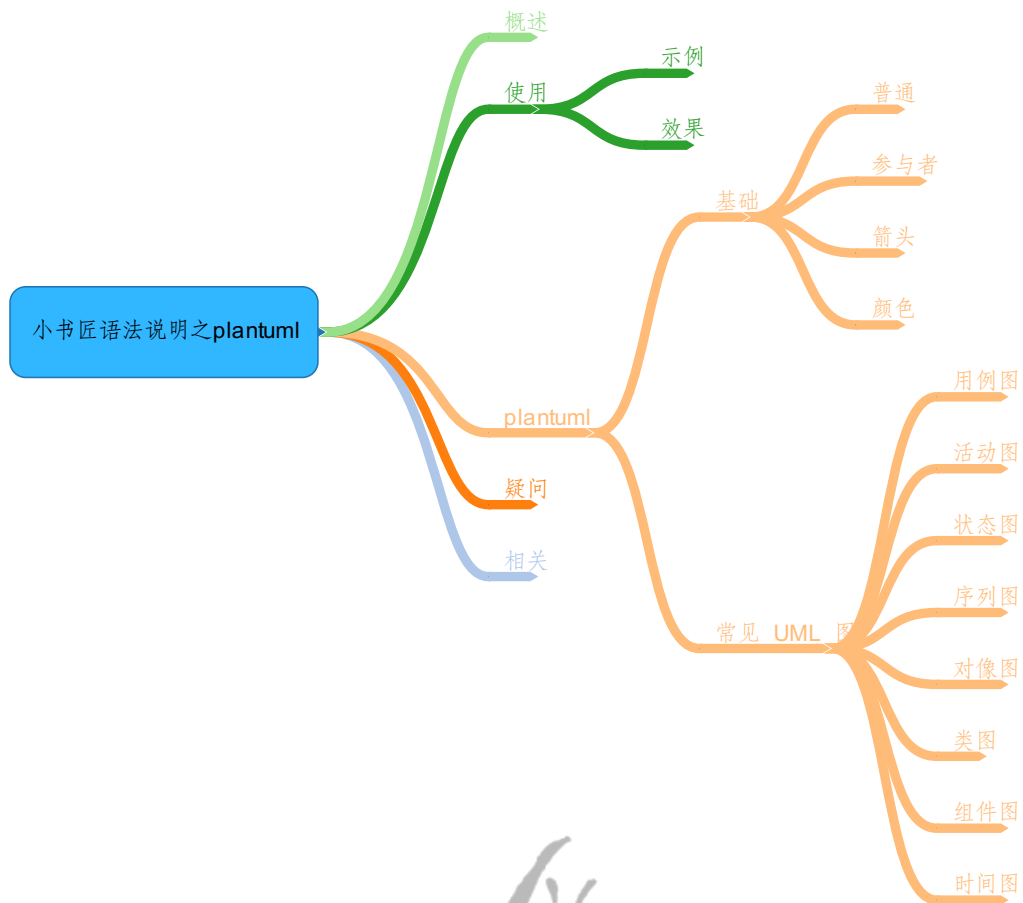


# 小书匠语法说明之 plantuml

小书匠

# 目录

概述	1
使用	1
示例	2
效果	2
plantuml	2
基础	2
普通	2
参与者	3
箭头	3
颜色	3
常见 UML 图	4
用例图	4
活动图	5
状态图	6
序列图	7
对象图	8
类图	9
组件图	9
时间图	10
疑问	11
相关	11



## 概述

plantuml是一个非常方便的uml工具，只要简单的编写文本，就可以得到对应的UML图形。这样的功能，对于写技术类的文章，特别有帮助。再加上自身是一种纯文本的格式，更加方便了文件版本的管理。plantuml除了可以绘制标准的uml图之外，还能绘制界面布局图、结构图、甘特图乃至数学公式等。可谓“plantuml在手，天下我有”。

plantuml 语法不是标准的 commonmark语法，不同编辑器或者博客对其支持的程度不一样，目前小书匠提供了对该语法的完整支持，并且实现了实时预览的效果。

PlantUML is an open-source tool allowing users to create UML diagrams from a plain text language. The language of PlantUML is an example of an Application Specific Language.[3] It uses Graphviz software to lay out its diagrams. It has been used to allow blind students to work with UML.[4][5] PlantUML also helps blind software engineers to design and read UML diagrams.

## 使用

元数据标识: **grammar\_plantuml**

想要使用该语法，需要在 **设置>扩展语法** 里把 **plantuml** 选项打开。或者在每篇文章的元数据里通过 **grammar\_plantuml** 进行控制。系统默认关闭 **plantuml** 语法功能

更详细的语法使用，可以参考这里 <http://plantuml.com/>

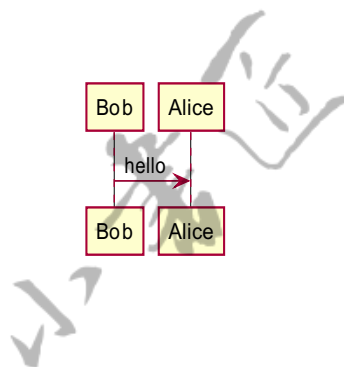
plantuml 语法需要使用第三方服务器，系统默认指定小书匠自定义的服务器，内网用户可以通过 **设置>扩展语法>plantuml** 里的服务器选项，修改成自己的 plantuml 服务器地址

代码段内的文字将被 **encode** 成 **base64** 并被压缩后做为 **.uml** 内容提交到服务器。

## 示例

```
1  ``plantuml!  
2    Bob->Alice : hello  
3  ````
```

## 效果



# plantuml

## 基础

### 普通

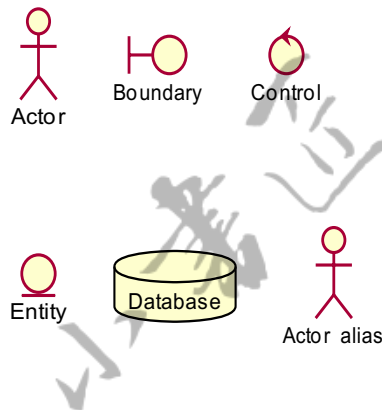
```
1  `` plantuml!  
2  title Title  
3  
4  ( )  
5  note left : Note  
6  
7  [ ]  
8  note right : Note  
9  
10 ' single-line comment  
11  
12 /'  
13 block comment
```

```
14 | '/'
15 | '''
```



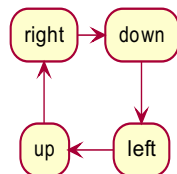
## 参与者

```
1 | ''' plantuml!
2 | actor Actor
3 | boundary Boundary
4 | control Control
5 | entity Entity
6 | database Database
7 | :Actor alias:
8 | '''
```



## 箭头

```
1 | ''' plantuml!
2 | up -up-> right
3 | -right-> down
4 | -down-> left
5 | -left-> up
6 | '''
```



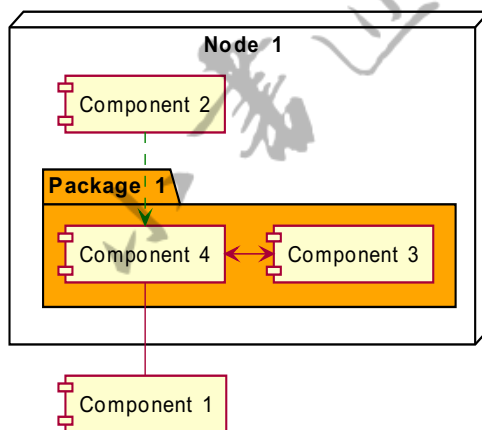
## 颜色

```
1 | ''' plantuml!
2 | ' =====
```

```

3  ' == Declaration ==
4  ' =====
5
6  [Component 1]
7
8  node "Node 1" {
9      package "Package 1" #Orange {
10         [Component 4]
11         [Component 3]
12     }
13     [Component 2]
14 }
15
16
17
18 ' =====
19 ' == Implementation ==
20 ' =====
21
22
23 node "Node 1" {
24     [Component 2] .[#Green]-> [Component 4]
25     [Component 3] <-left-> [Component 4]
26     [Component 4] -- [Component 1]
27 }
28 ```

```



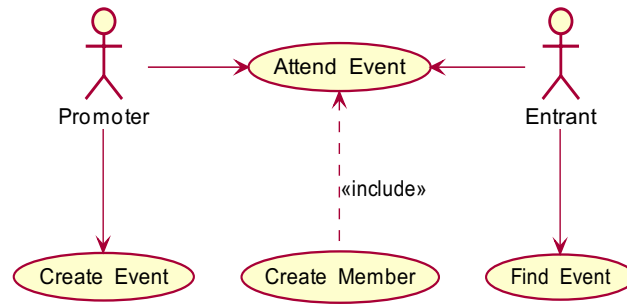
## 常见 UML 图

### 用例图

```

1  ``` plantuml!
2  actor Promoter
3  actor Entrant
4
5  Promoter --> (Create Event)
6  Promoter -> (Attend Event)
7
8  Entrant --> (Find Event)
9  (Attend Event) <- Entrant
10
11 (Attend Event) <.. (Create Member) : <<include>>

```

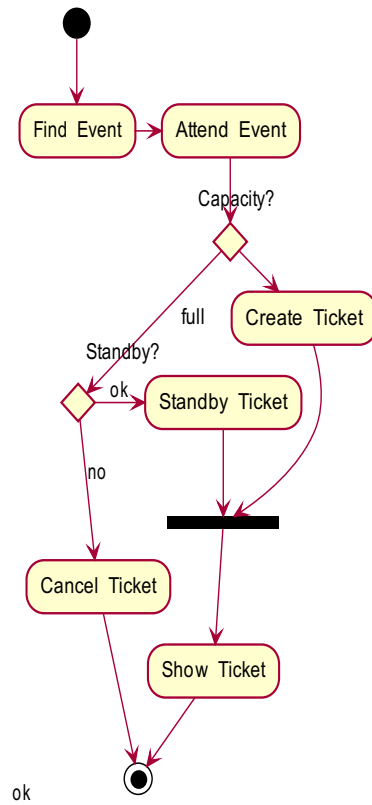


## 活动图

```

1  ``` plantuml!
2  (*) --> "Find Event"
3  "Find Event" -> "Attend Event"
4
5  if "Capacity?" then
6    ->[ok] "Create Ticket"
7  else
8    -->[full] if "Standby?" then
9      ->[ok] "Standby Ticket"
10     else
11       -->[no] "Cancel Ticket"
12       "Cancel Ticket" --> (*)
13     endif
14   endif
15
16   "Create Ticket" --> ==show==
17   "Standby Ticket" --> ==show==
18   ==show== --> "Show Ticket"
19   "Show Ticket" --> (*)
20   ```

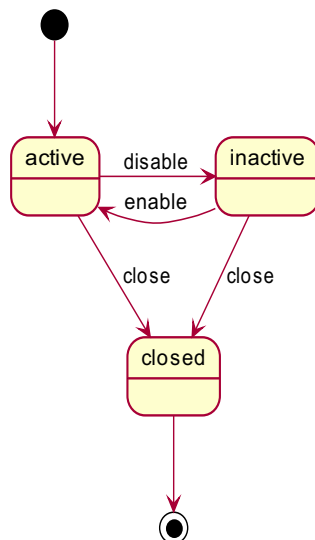
```



## 状态图

```

1  `` plantuml!
2  [*] --> active
3
4  active -right-> inactive : disable
5  inactive -left-> active : enable
6
7  inactive --> closed : close
8  active --> closed : close
9
10 closed --> [*]
11 ``
  
```



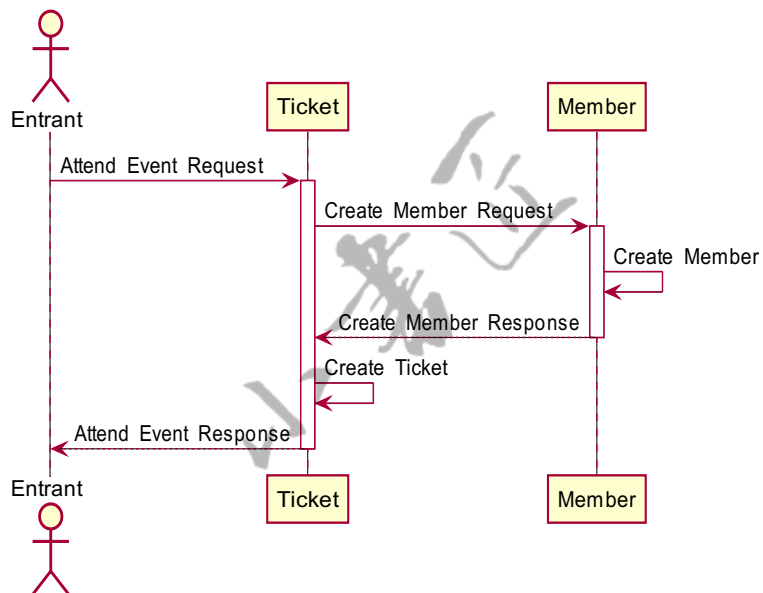


# 序列图

```

1  `` plantuml!
2  actor Entrant
3
4  Entrant -> Ticket : Attend Event Request
5
6  activate Ticket
7  Ticket -> Member : Create Member Request
8
9  activate Member
10 Member -> Member : Create Member
11 Ticket <-- Member : Create Member Response
12 deactivate Member
13
14 Ticket -> Ticket : Create Ticket
15 Entrant <-- Ticket : Attend Event Response
16 deactivate Ticket
17 ``

```



```

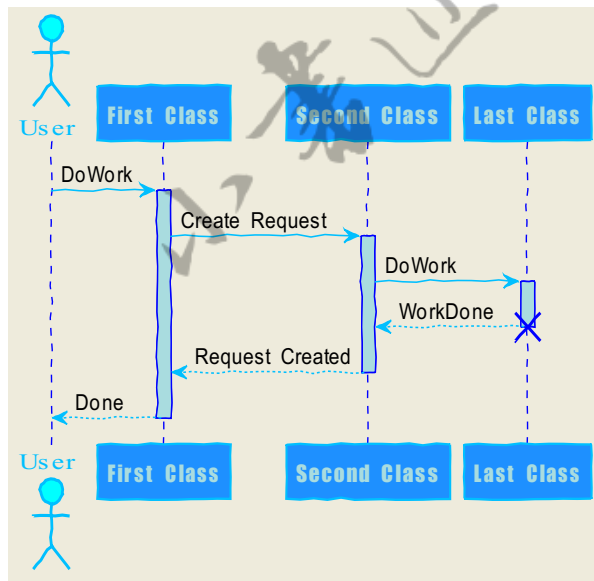
1  `` plantuml!
2
3  @startuml
4  skinparam backgroundColor #EEEBDC
5  skinparam handwritten true
6
7  skinparam sequence {
8      ArrowColor DeepSkyBlue
9      ActorBorderColor DeepSkyBlue
10     LifeLineBorderColor blue
11     LifeLineBackgroundColor #A9DCDF
12
13     ParticipantBorderColor DeepSkyBlue
14     ParticipantBackgroundColor DodgerBlue
15     ParticipantFontName Impact
16     ParticipantFontSize 17
17     ParticipantFontColor #A9DCDF
18
19     ActorBackgroundColor aqua
20     ActorFontColor DeepSkyBlue
21     ActorFontSize 17

```

```

22 ActorFontName Apex
23 }
24
25 actor User
26 participant "First Class" as A
27 participant "Second Class" as B
28 participant "Last Class" as C
29
30 User -> A: DoWork
31 activate A
32
33 A -> B: Create Request
34 activate B
35
36 B -> C: DoWork
37 activate C
38 C --> B: WorkDone
39 destroy C
40
41 B --> A: Request Created
42 deactivate B
43
44 A --> User: Done
45 deactivate A
46
47 @enduml
48 ```

```



## 对像图

```

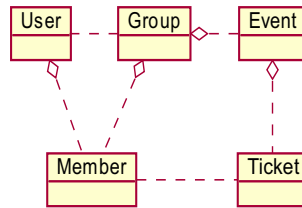
1  ``` plantuml!
2  object User
3  object Group
4  object Member
5
6  object Event
7  object Ticket
8
9  User . Group
10 User o.. Member
11 Group o.. Member
12

```

```

13 Group o.. Event
14 Event o.. Ticket
15 Member . Ticket
16 ```

```

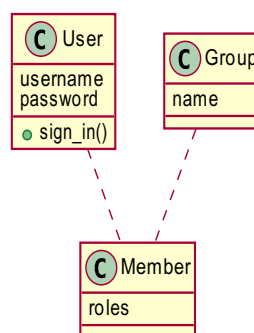


## 类图

```

1  ``` plantuml!
2  class User {
3      username
4      password
5      +sign_in()
6  }
7
8  class Group {
9      name
10 }
11
12 class Member {
13     roles
14 }
15
16 User .. Member
17 Group .. Member
18 ```

```



## 组件图

```

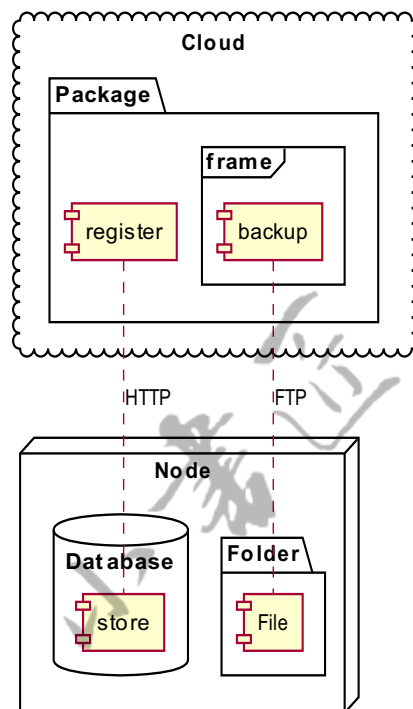
1  ``` plantuml!
2  cloud "Cloud" {
3      package "Package" {
4          [register]
5          frame "frame" {
6              [backup]
7          }
8      }

```

```

9  }
10
11 node "Node" {
12     database "Database" {
13         [store]
14     }
15     folder "Folder" {
16         [File]
17     }
18 }
19
20 [register] .. [store] : HTTP
21 [backup] .. [File] : FTP
22 ```

```



## 时间图

```

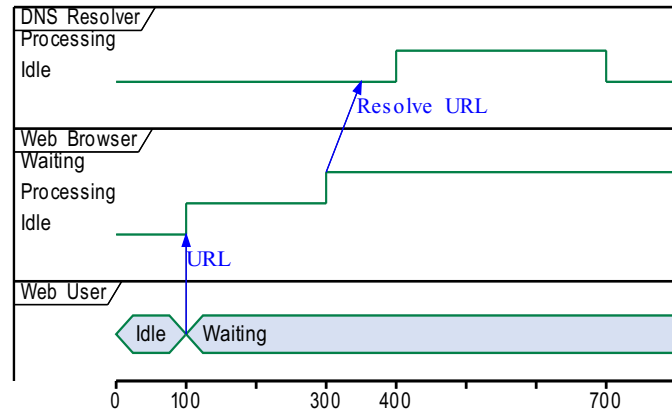
1  ``` plantuml!
2  @startuml
3  robust "DNS Resolver" as DNS
4  robust "Web Browser" as WB
5  concise "Web User" as WU
6
7  @0
8  WU is Idle
9  WB is Idle
10 DNS is Idle
11
12 @+100
13 WU -> WB : URL
14 WU is Waiting
15 WB is Processing
16
17 @+200
18 WB is Waiting
19 WB -> DNS@+50 : Resolve URL

```

```

20
21 @+100
22 DNS is Processing
23
24 @+300
25 DNS is Idle
26 @enduml
27 ` ` `

```



疑问

相关

1. [plantuml 官网](#)
2. [plantuml使用参考文档\(PDF格式\)](#)
3. [plantuml 官方语法](#)