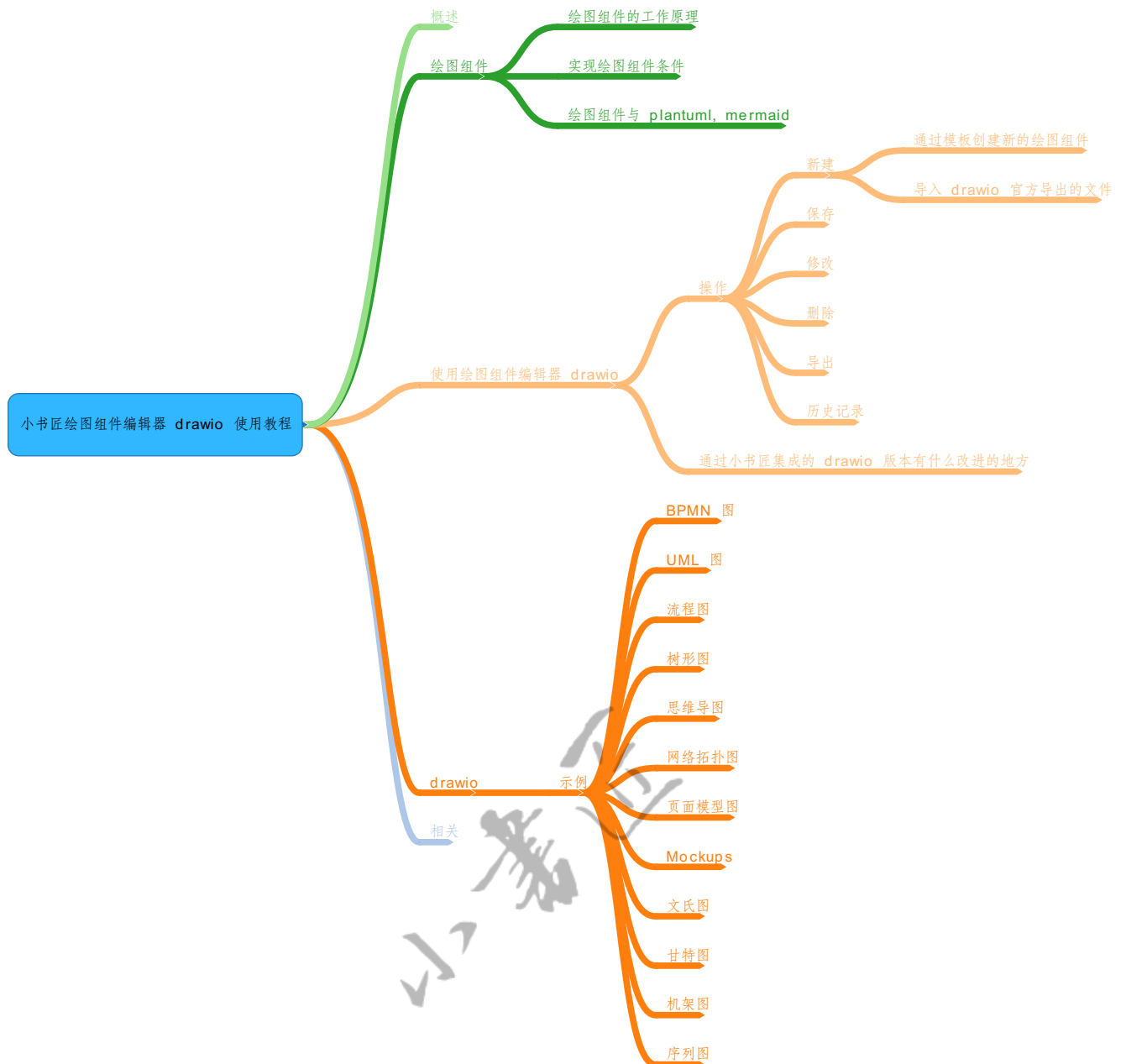


小书匠绘图组件编辑器 drawio 使用教程

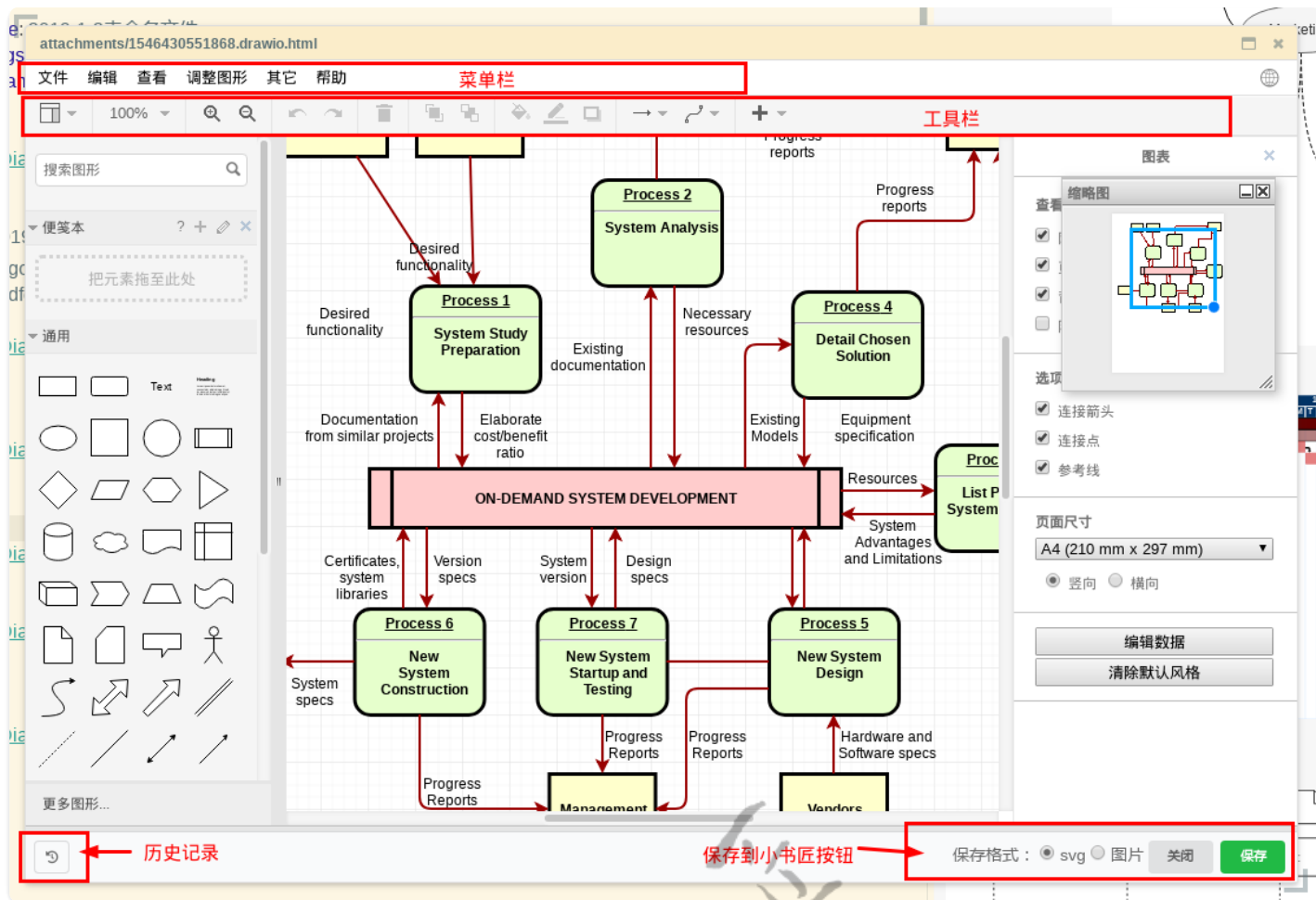
小書匠

目录

概述	1
绘图组件	2
绘图组件的工作原理	3
实现绘图组件条件	3
绘图组件与 plantuml, mermaid	3
使用绘图组件编辑器 drawio	3
操作	3
新建	3
通过模板创建新的绘图组件	4
导入 drawio 官方导出的文件	5
保存	6
修改	6
删除	7
导出	7
历史记录	8
通过小书匠集成的 drawio 版本有什么改进的地方	11
drawio	11
示例	11
BPMN 图	12
UML 图	12
流程图	13
树形图	14
思维导图	14
网络拓扑图	15
页面模型图	16
Mockups	17
文氏图	18
甘特图	18
机架图	19
序列图	20
相关	21



概述



绘图组件编辑器

[draw.io](#) 是一款在线图表编辑工具，可以用来编辑工作流、BPM、org charts、UML、ER图、网络拓扑图等。[draw.io](#) 也能导出各类图片（比如 png、jpg、svg）和 PDF，同时也能导出 HTML 格式和 VSDX 格式（Microsoft Visio 图形格式）。

小书匠编辑器直接内置集成了 [Draw.io](#)，用户可以非常方便的在小书匠编辑器内绘制各种图例。

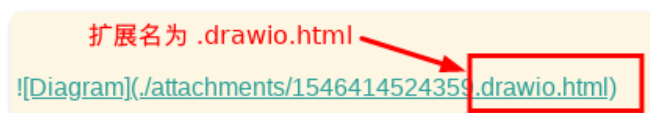
使用 [drawio](#) 可以创建 流程图，序列图，网络拓扑图，甘特图，思维导图，模型图，文氏图等

通过小书匠集成的 [drawio](#) 版本，您甚至可以使用到 只有在 [drawio](#) 付费版本里才有提供的历史记录功能。

更详细的 [drawio](#) 特性，可以查看 [drawio](#) 自己的[官方特性介绍](#)，以及 [官方特性示例](#)

绘图组件

小书匠的绘图组件是在现在的图片语法基础上集成的，只要在文章内使用了图片语法，在指定图片链接里包含了扩展名为 **.drawio.html** 文件，小书匠系统会将该图片当成绘图组件来处理。



图片语法

绘图组件的工作原理

绘图组件的原理和表格组件的原理很像，不同的是附件内容不一样。

绘图组件其实就是一个包含了绘图数据的 `HTML` 文件。只要该 `HTML` 文件扩展名为 `.drawio.html` 或者 `.drawio`，并做为该 `markdown` 文章的附件，通过 `markdown` 的图片语法进行引用，系统就会将该 `markdown` 文章里原本应该显示一张图片的内容替换成 `HTML` 文件里 `svg` 标签范围内的数据。可以通过导出为 `zip` 功能，查看该 `HTML` 文件的实际内容。

实现绘图组件条件

1. 扩展名为 `.drawio` 或者 `.drawio.html` 的文件
2. `markdown` 文章的附件，也就是以 `./` 开头的链接引用
3. 通过小书匠自带的 `drawio` 编辑器生成的 `html` 文件
4. 必须是块级图片语法(也就是图片语法的上下各空一行，前后不能有其他文字)

绘图组件与 plantuml, mermaid

对于一般的流程图，序列图，类图等，建议使用 `plantuml`, `mermain` 等纯文本的绘图语法来生成。

当 `plantuml`, `mermaid` 创建的图例不能满足用户的需求时，可以考虑使用绘图组件。

`mermaid`, `plantuml` 等语法通过纯文本的方式生成图例，而 `drawio` 则是直接在一块画布上手动绘制。

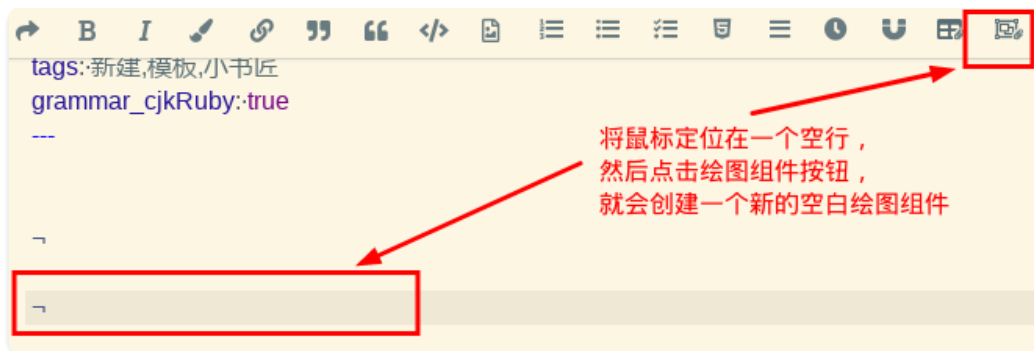
使用绘图组件编辑器 drawio

操作

新建

点击工具栏 **绘图组件** 按钮，系统就会创建一个扩展名为 `drawio.html` 的附件，并打开 `drawio` 编辑器操作界面。

用户在绘图编辑器操作界面的每次修改，都会把文件保存到这个扩展名为 `drawio.html` 文件上。

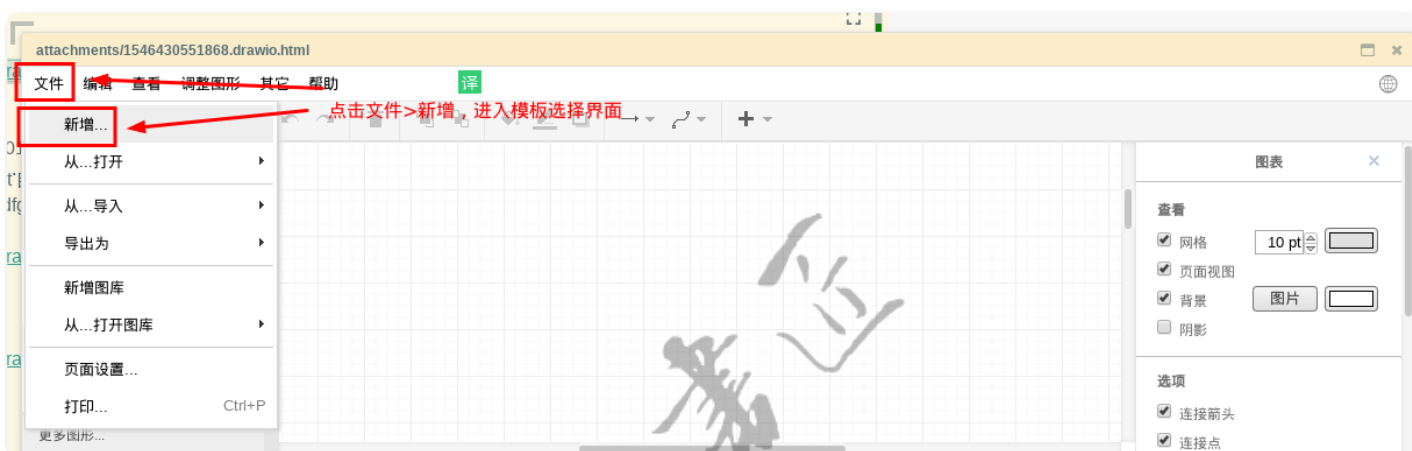


将鼠标定位在一个空行，
然后点击绘图组件按钮，
就会创建一个新的空白绘图组件

创建一个新的绘图组件

通过模板创建新的绘图组件

1. 先创建一个空白组件，进入绘图编辑器界面
2. 选择绘图编辑器菜单栏 **文件>新增**



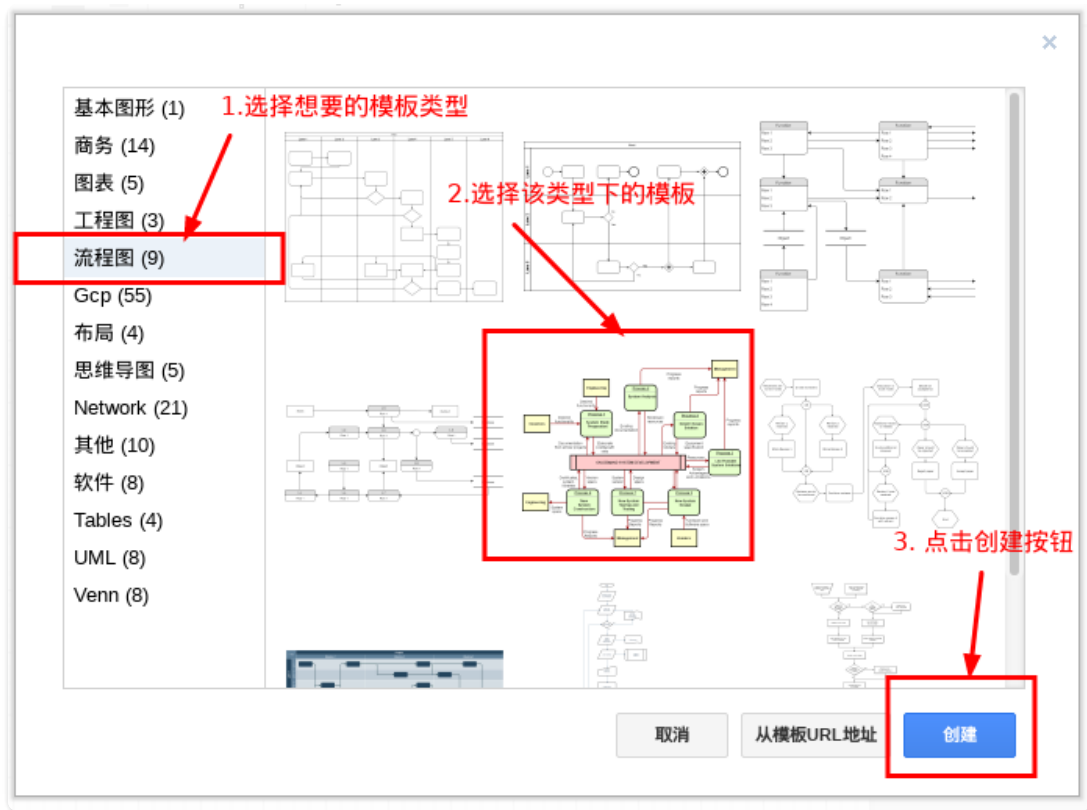
点击文件>新增，进入模板选择界面

进入模板选择界面

3. 选择一个模板。

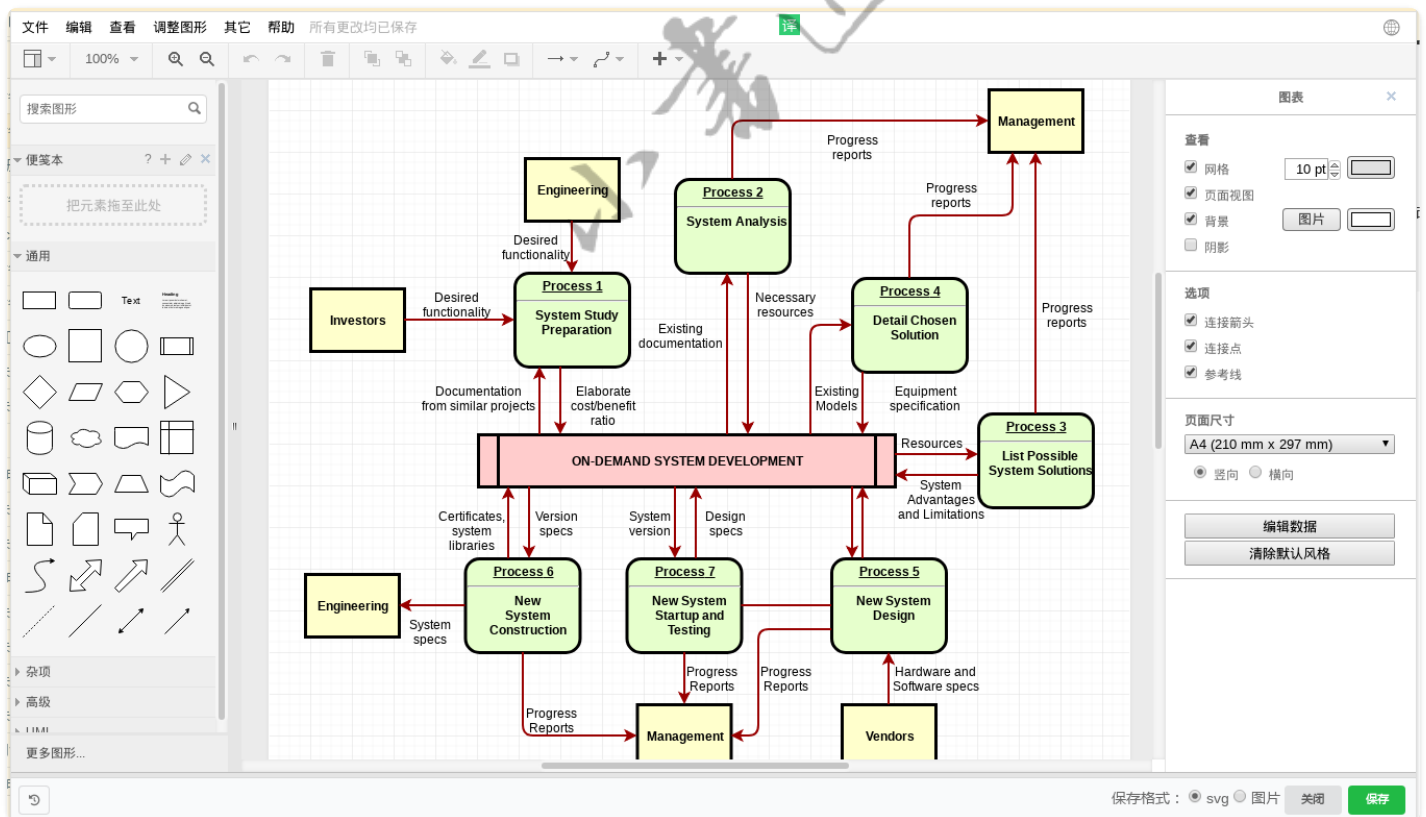
系统提供了丰富的模板，并根据不同的类型进行了分类，比如流程图，模型图，思维导图，网络拓扑图等，根据自己的需求，选择一个相应的类型，再选择一个具体的模板。

如果系统提供的模板不满足需求，用户还可以通过网络地址的模板，创建图例



选择模板

4. 根据流程模板创建的图例



成功创建

导入 drawio 官方导出的文件

1. 在 drawio 官方编辑器上将图例导出为 xml 格式
2. 在小书匠编辑器里点击 绘图组件按钮，进入绘图编辑界面

3. 在绘图编辑界面里选择菜单 **文件>从...打开**，将您刚才导出的 xml 文件打开

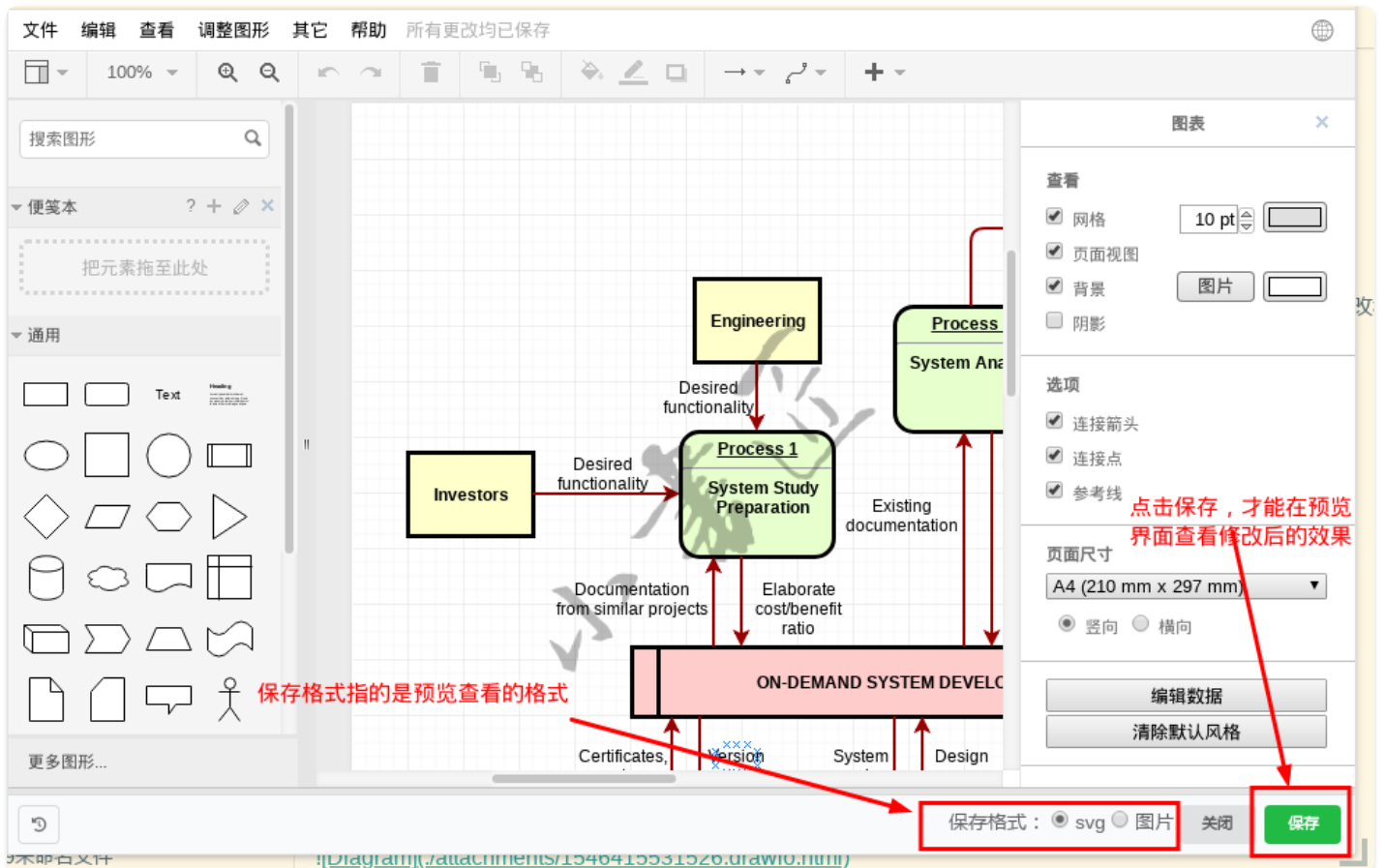
注：

如果选择 **文件>从...打开**，系统会提示你放弃当前文件的修改，并创建新的文件。如果选择 **文件>从...导入**，则是在当前文件的基础上进行添加新的图例上添加新的图例，并不会清空整个原有的图例。

保存

当用户完成了在绘图编辑器的修改时，需要点击保存按钮，将该绘图保存到小书匠内置数据库里。

只有点击了保存按钮，才会在预览界面查看该新修改的绘图效果。



保存

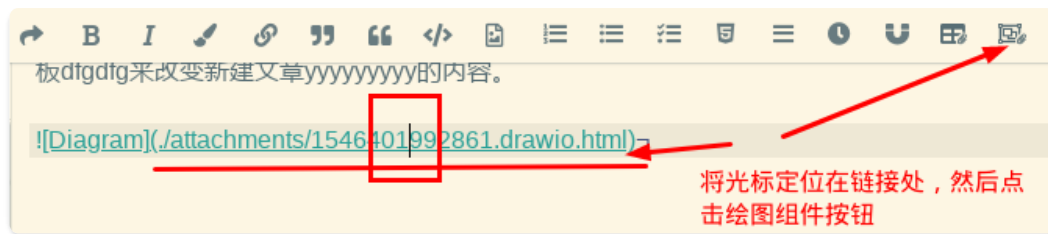
保存格式一般选择 **svg** 就可以，如果生成的 **svg** 不符合预期，也可以尝试保存成 **png** 格式。

修改

修改已有的绘图组件方法有两种

方法一：

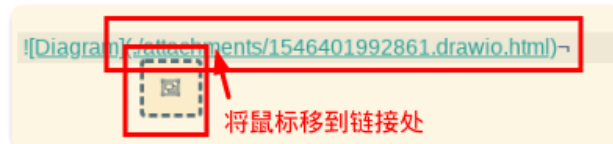
在 **markdown** 文章里，将光标定位到想要修改的绘图组件的链接里，然后点击工具栏上的**绘图组件**按钮。



通过工具栏

方法二：

在 **codemirror** 内置编辑器里，将鼠标移到想要修改的绘图组件链接上，会在该链接的下方显示一个**绘图组件**的按钮,点击该按钮进入绘图操作界面。



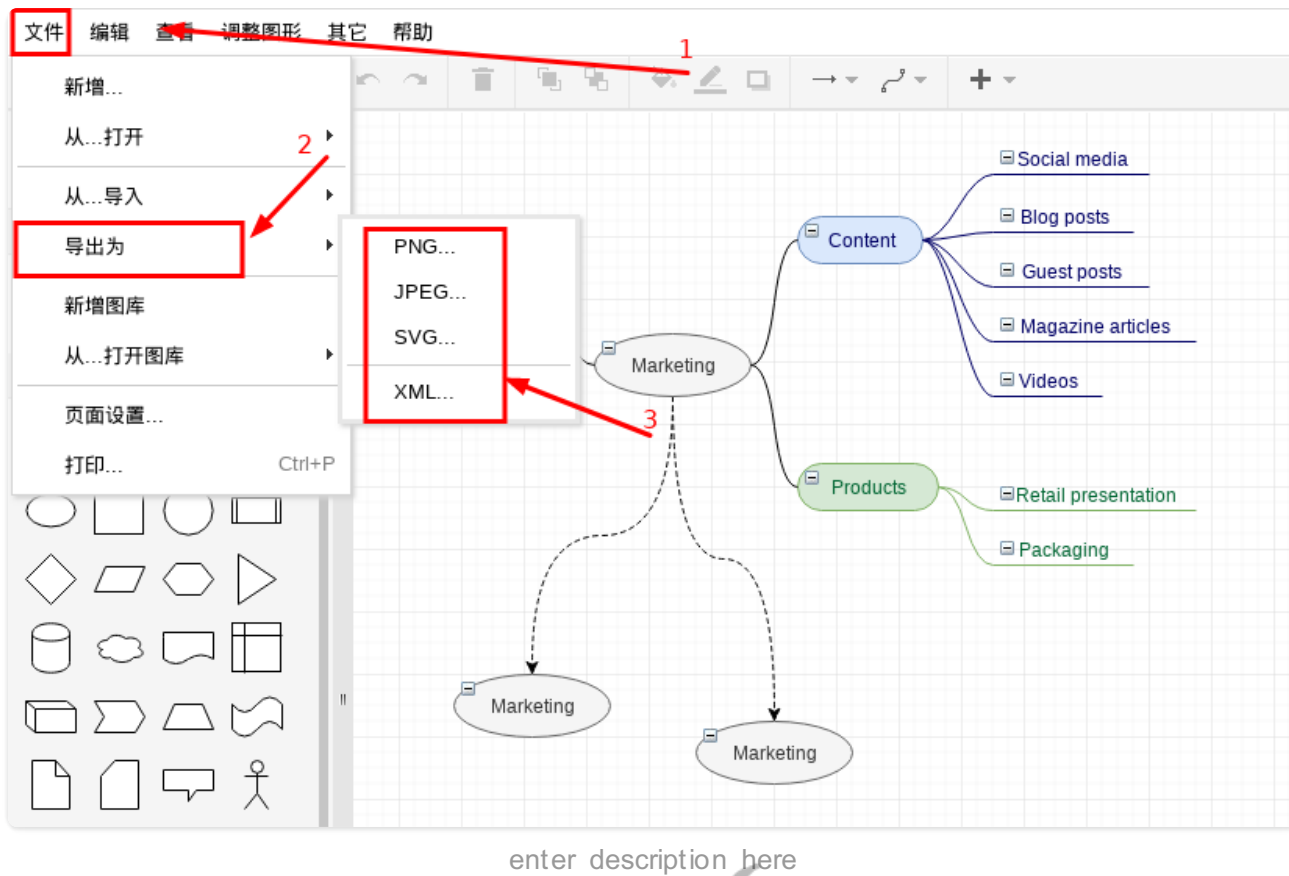
将鼠标移到绘图组件链接处

删除

1. 在 **markdown** 文章里将想要删除的绘图组件标识去掉。
2. 再通过文章的附件管理，把对应的附件删除。

导出


可以在绘图编辑器的菜单栏上选择 **文件>导出为**，小书匠集成的 **drawio** 提供 **png, jpg, svg, xml** 四种导出格式。



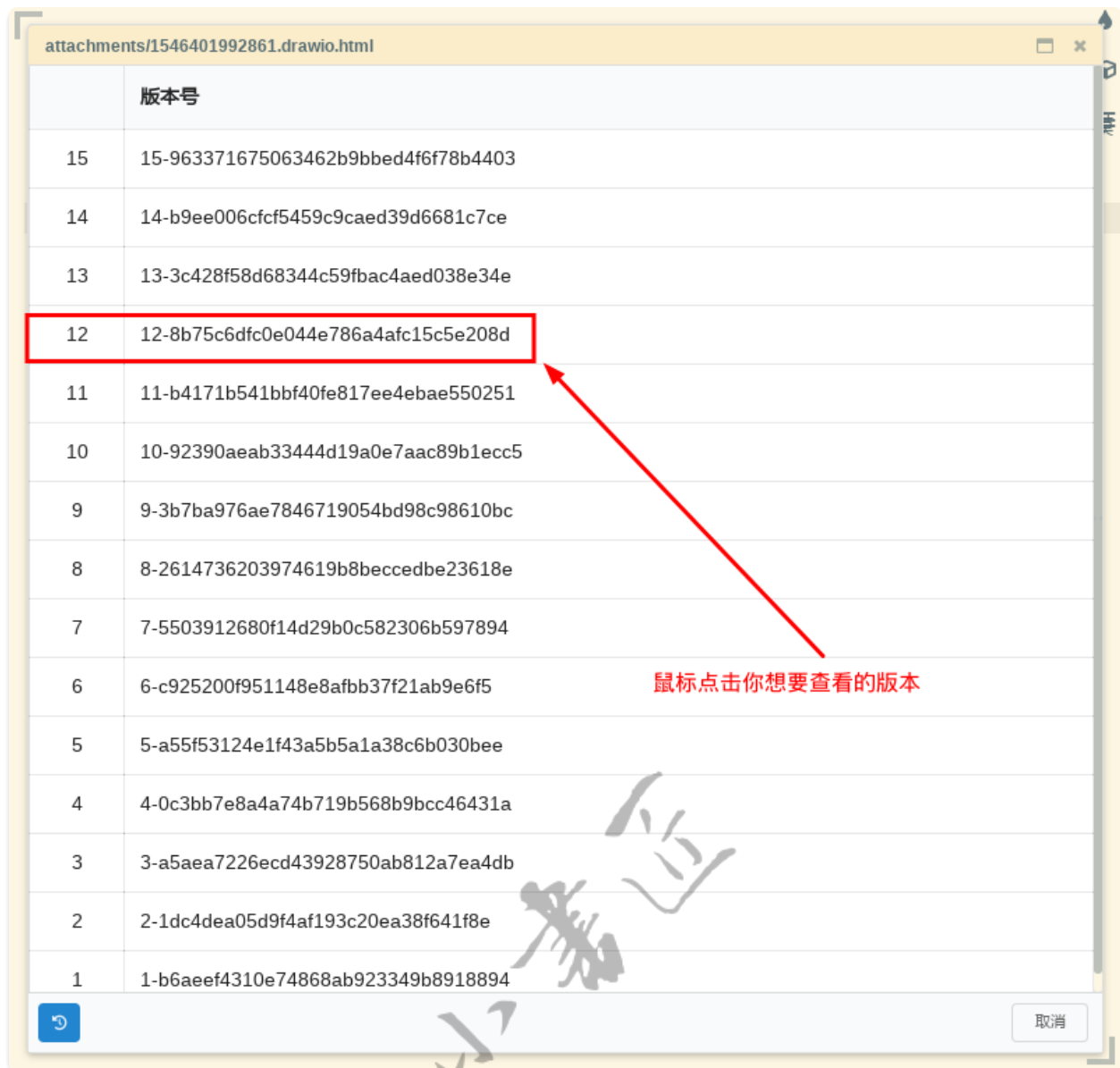
如果想把导出的文件在 drawio 官方平台上继续使用，可以选择导出为 xml 格式，然后再在 drawio 官方平台上导入刚才生成的 xml 文件。

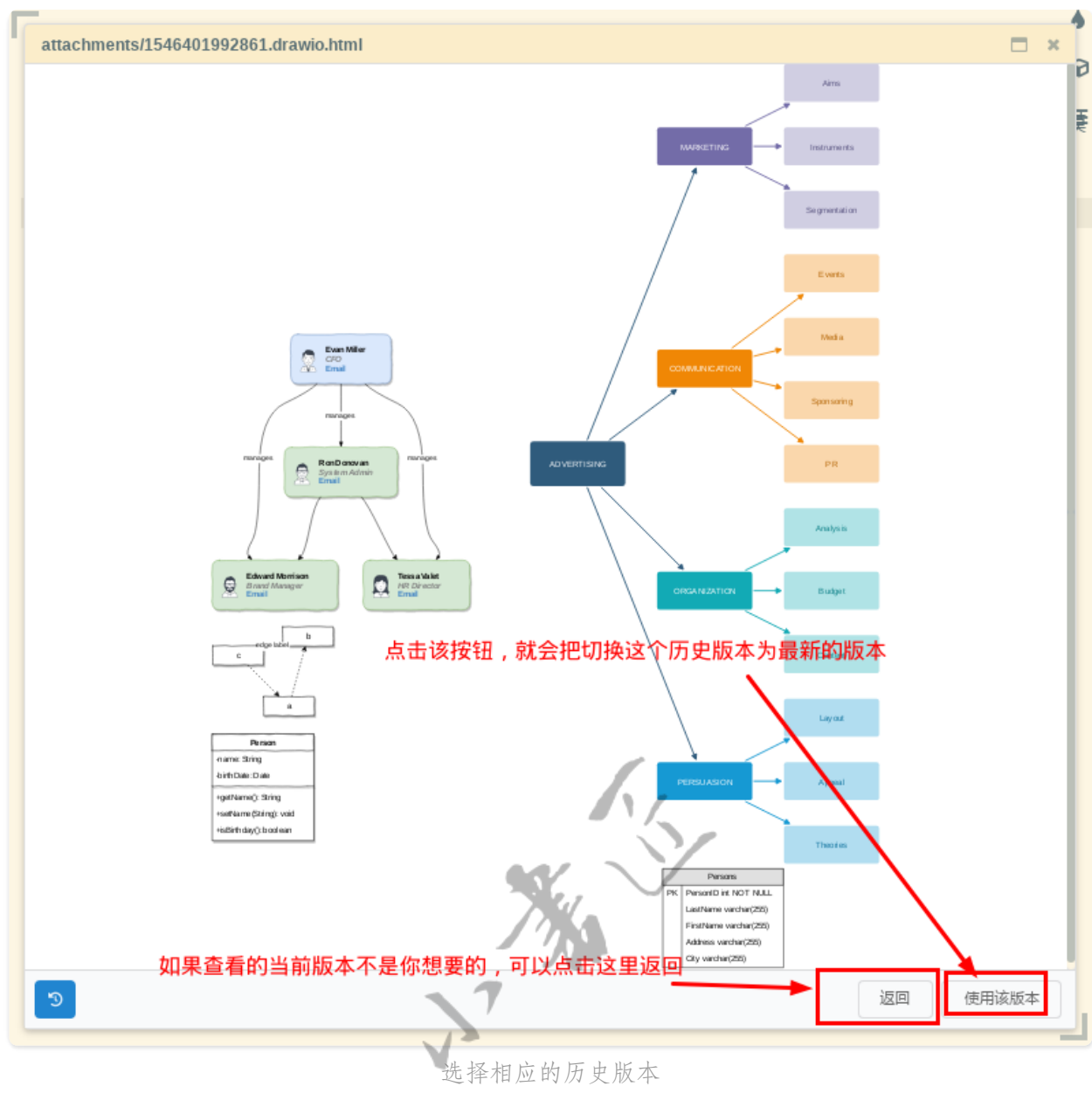
历史记录

用户在绘图编辑器的修改都会通过小书匠自动保存功能，保存到小书匠内置数据库内，因此用户可以通过历史记录恢复到自己想要的历史版本。

通过小书匠集成的 drawio 绘图编辑器，实现了本地历史记录的功能，用户只要点击编辑器左下方的  按钮，系统就会切换到该绘图文件的所有历史记录。







通过小书匠集成的 drawio 版本有什么改进的地方

提供了绘图版本管理，
数据直接通过小书匠系统保存

drawio

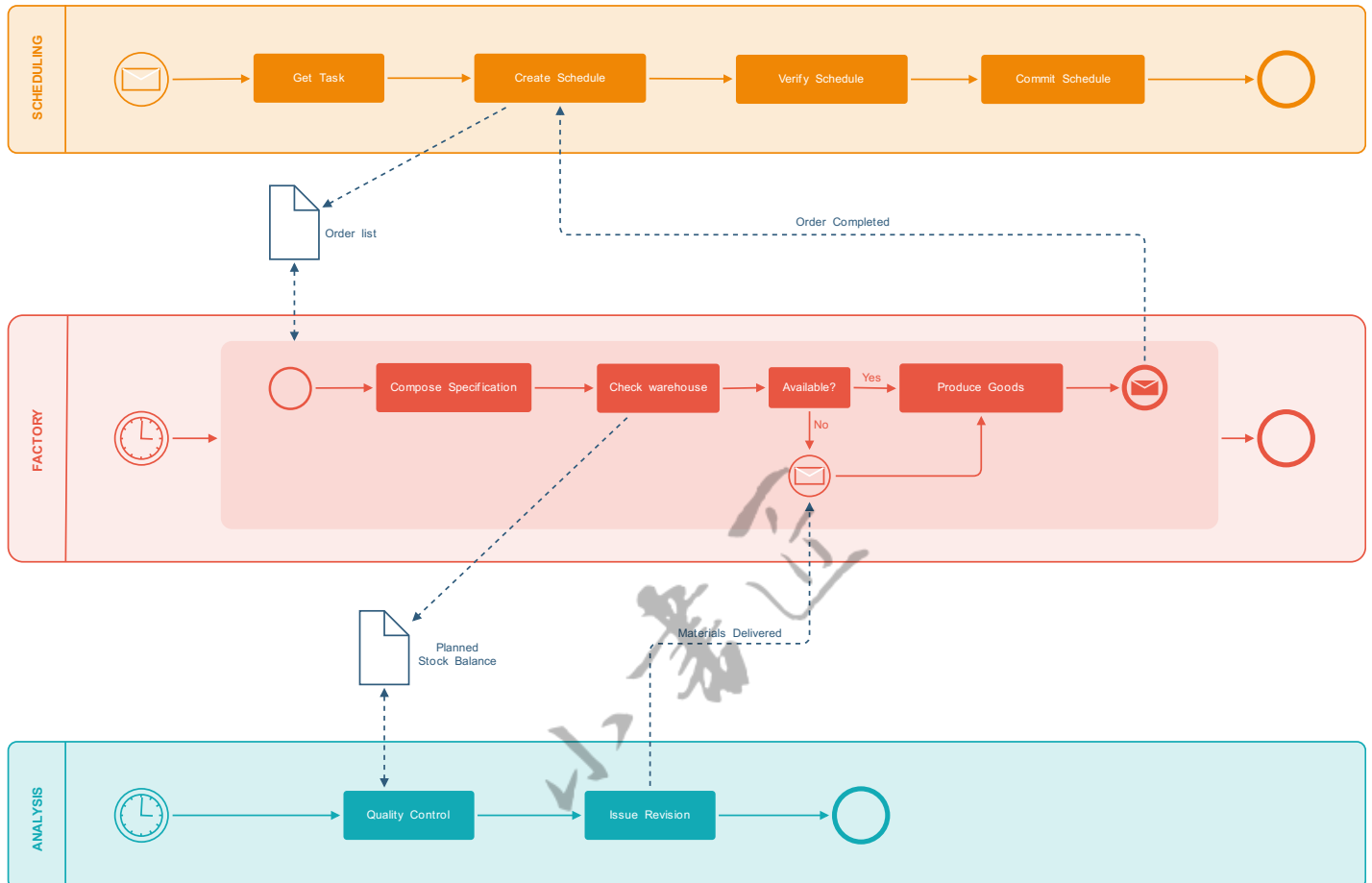
可以到这里查看drawio官方的[教程](#)

示例

BPMN 图

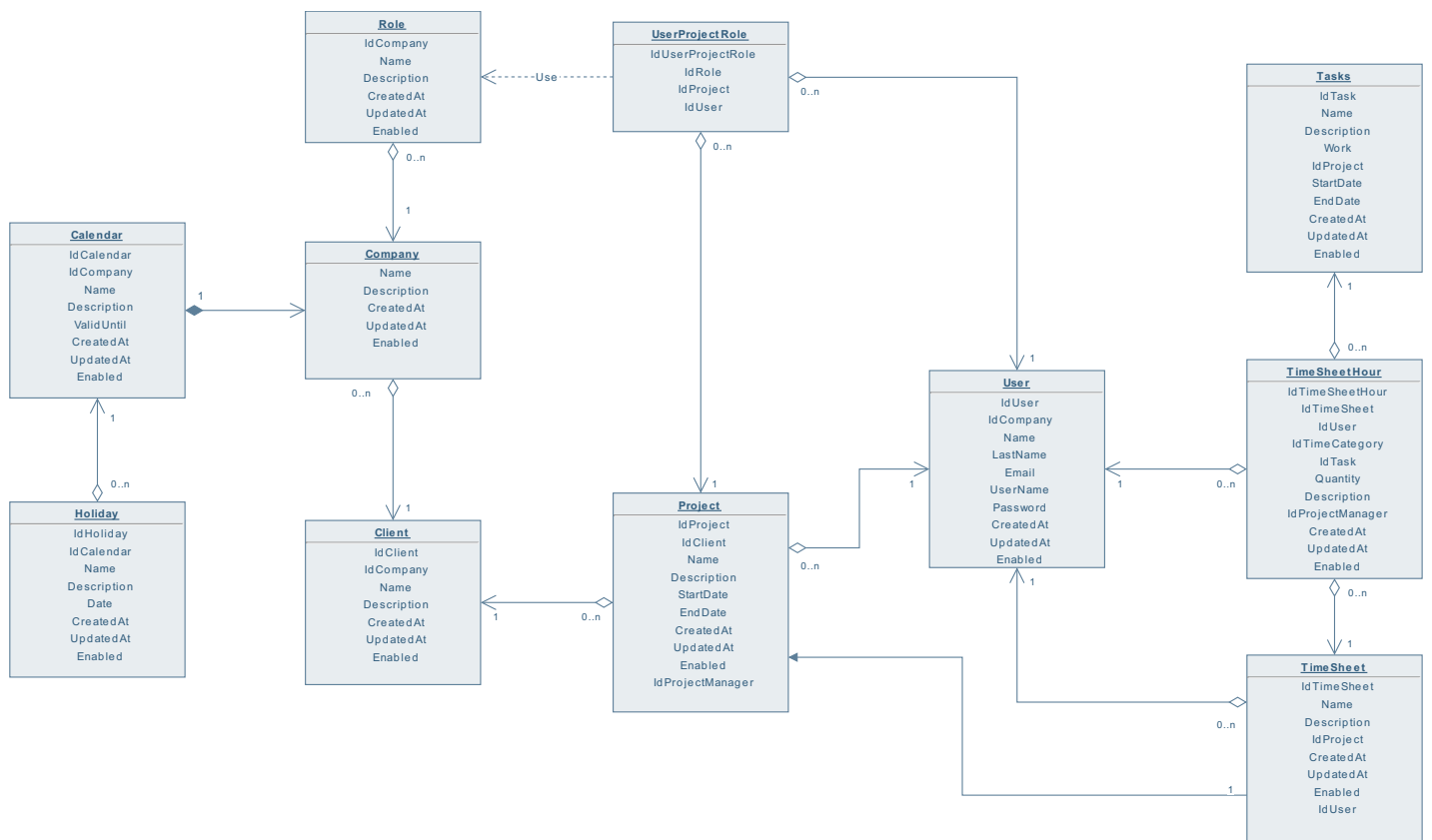
Complex business processes and workflows are much easier to understand when presented in diagram form. The shape library includes all the components you need to build your BPMN diagrams.

These diagrams are especially useful for when your business processes need to be audited or for onboarding new employees.



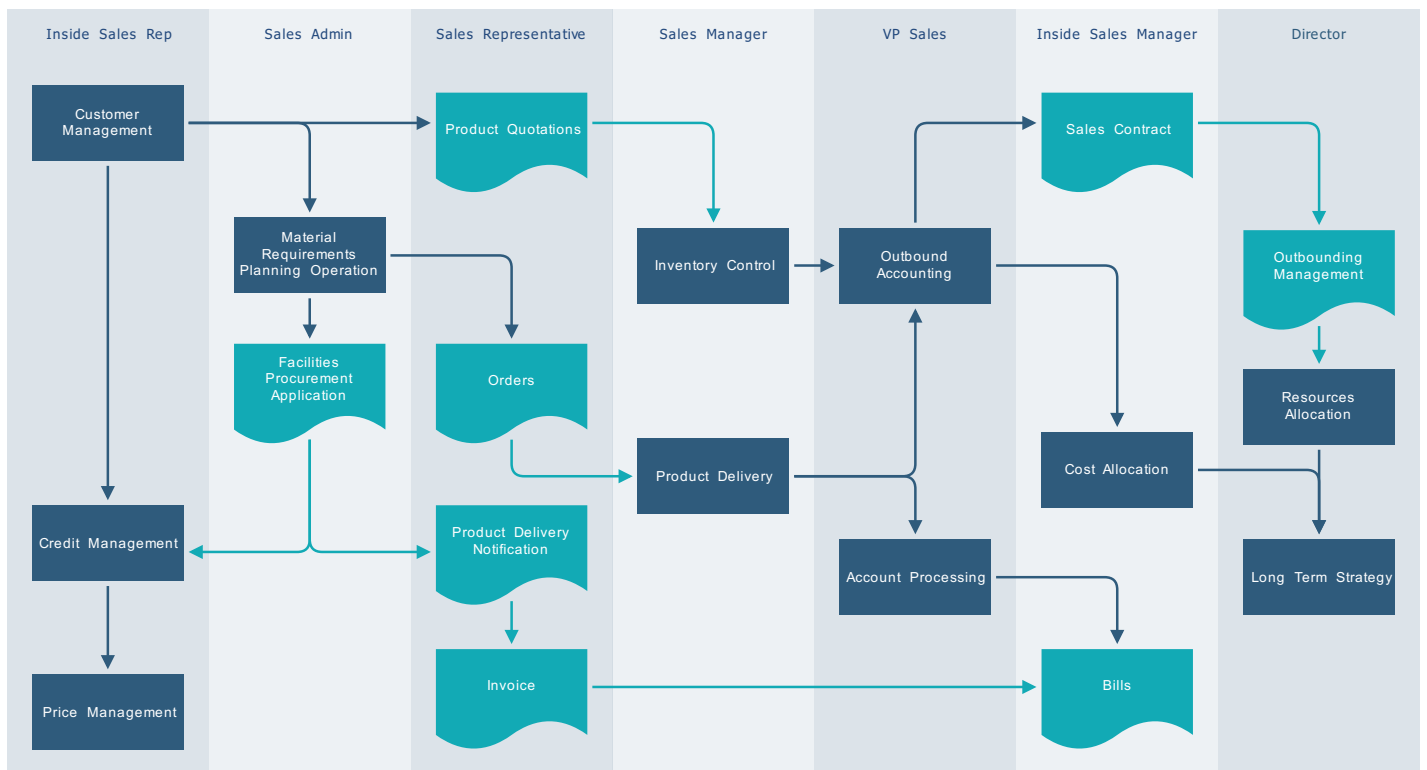
UML 图

You can't avoid documentation when programming in teams, and UML diagrams make it easy to document all aspects of your software engineering projects – both behavioural and structural. Create use case, communication, sequence, interaction, class, object and package diagrams with ease with the UML components in the extensive shape library.



流程图

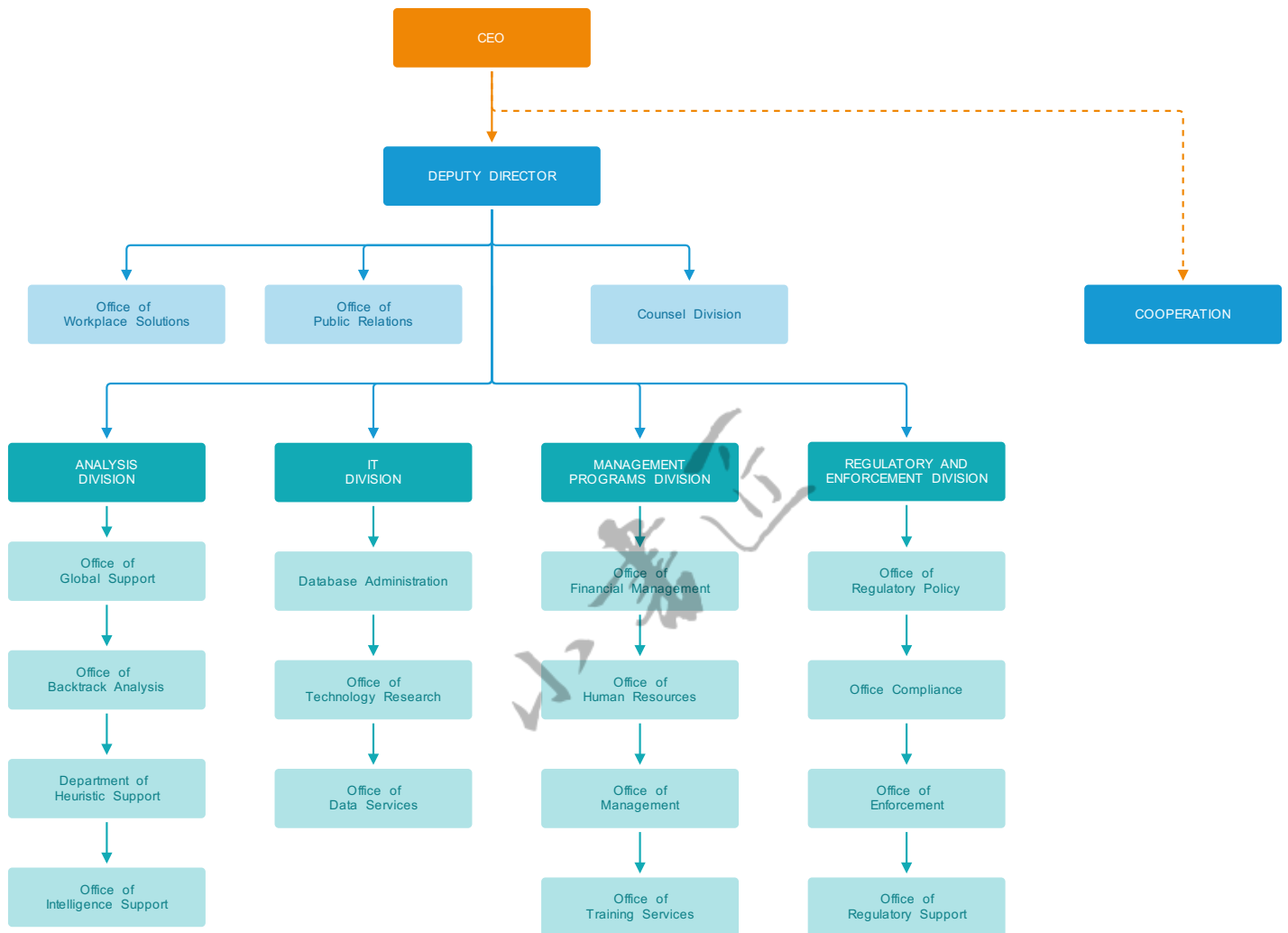
Do you have a workflow that you want to improve or document, but you don't want to use the strict BPMN notation? No problems! [draw.io](#) supports all of the shapes used in flow charts of various types. Because we learn visually, complex workflows are ideally presented as diagrams. Use [draw.io](#)'s drag & drop interface or the automatic layout function to create your flow charts quickly and easily.



树形图

Tree diagrams are one of the most commonly used diagrams to illustrate any situation where there is a hierarchy of elements. Organization charts, the content structure of a training document or book, a breakdown of components within components – you'll find many uses for tree diagrams in your organization.

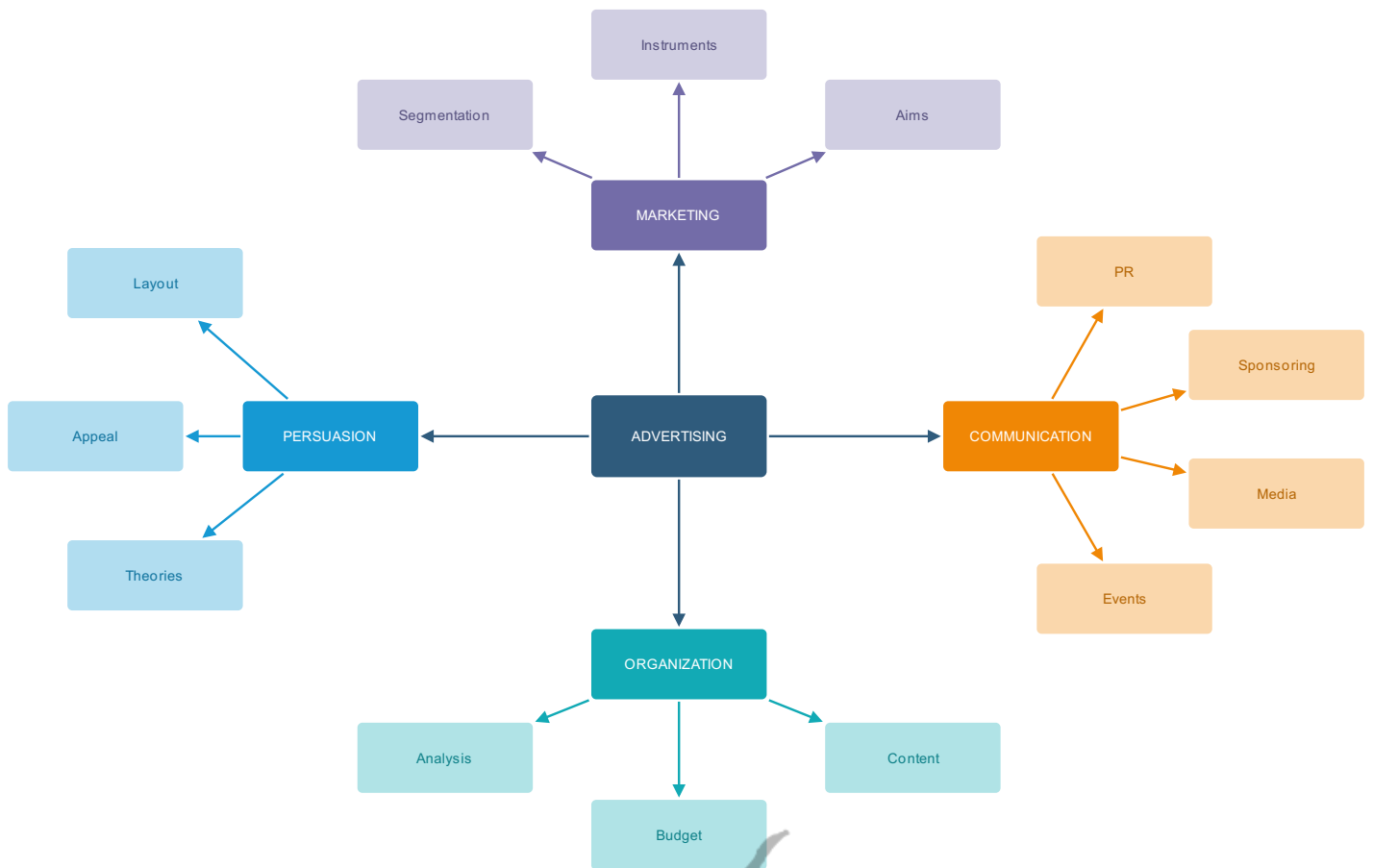
Take advantage of the automatic layout feature and keyboard shortcuts to quickly create an org chart or any other tree diagram.



思维导图

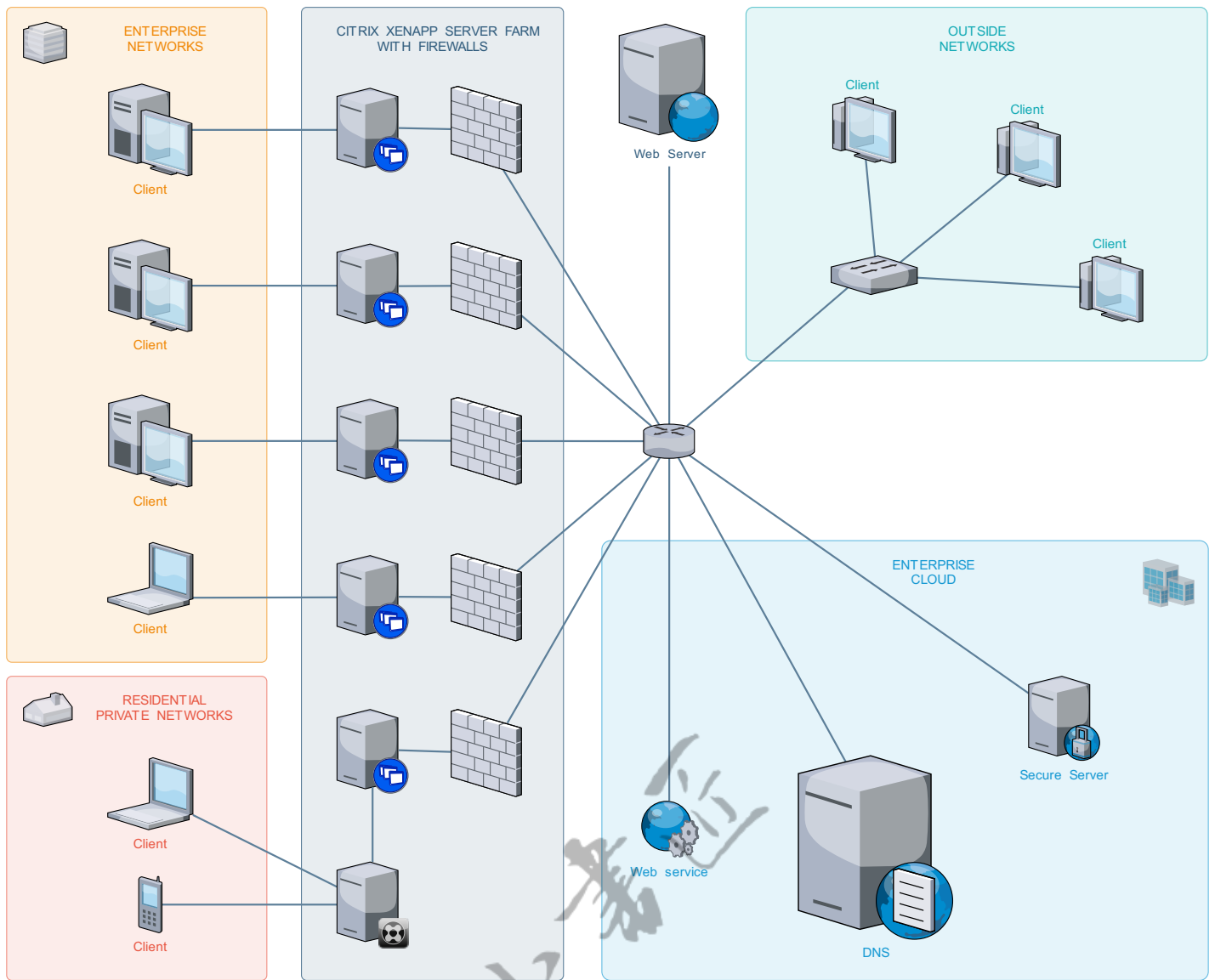
If you are like the majority, you prefer to brainstorm visually. Mind maps are ideal for quickly jotting down your thoughts about a concept or problem, and organizing the information into related topics. Use mind maps to help you remember information that you are studying, solve problems, make decisions and prepare for writing a complex document.

Use keyboard shortcuts to navigate around, connect and add new elements to your mind map quickly.



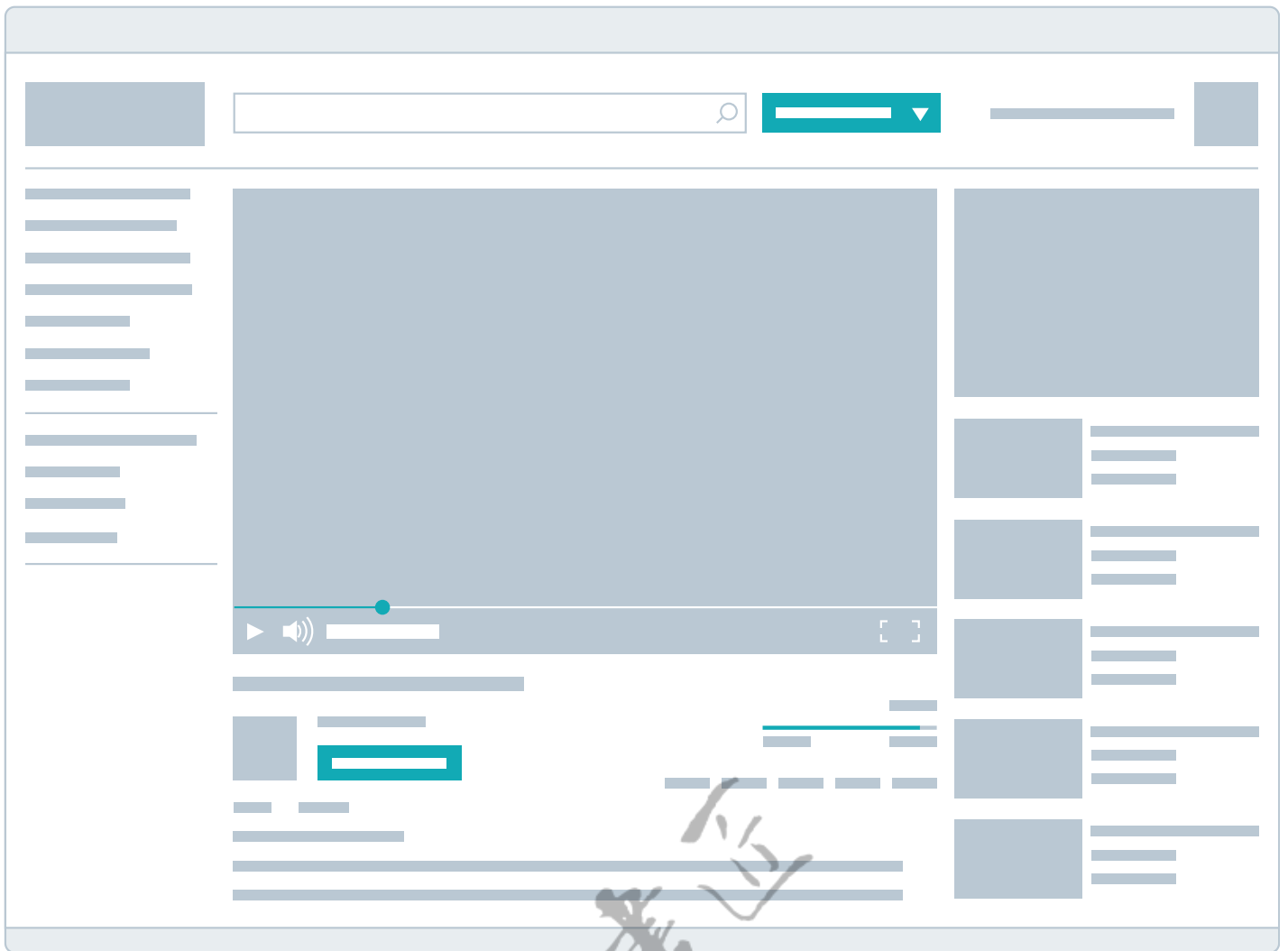
网络拓扑图

The network of connected devices in companies are becoming increasingly complex. With a wide variety of both generic and specific diagram elements commonly used for visualizing your IT infrastructure, you can create a detailed illustration of your network with all of its devices quickly and easily with draw.io, making network administration much less frustrating.



页面模型图

You can use wireframe models, also known as page schematics, to plan the layout of your website or user interface. A wide variety of professions work with wireframes on a daily basis: developers, user experience designers, business analysts, visual designers and graphic artists. They have become more important with the rise of mobile interfaces – wireframe models help you simplify your interface and make it easy for users to use your application or website.



Mockups

In a similar way to wireframe models, mockups are used to quickly prototype a user interface during application development. These days, they are most often used when creating mobile websites and app interfaces, and include more of the graphic design elements than wireframe models. They are also used to visualize physical products before committing to a final design.

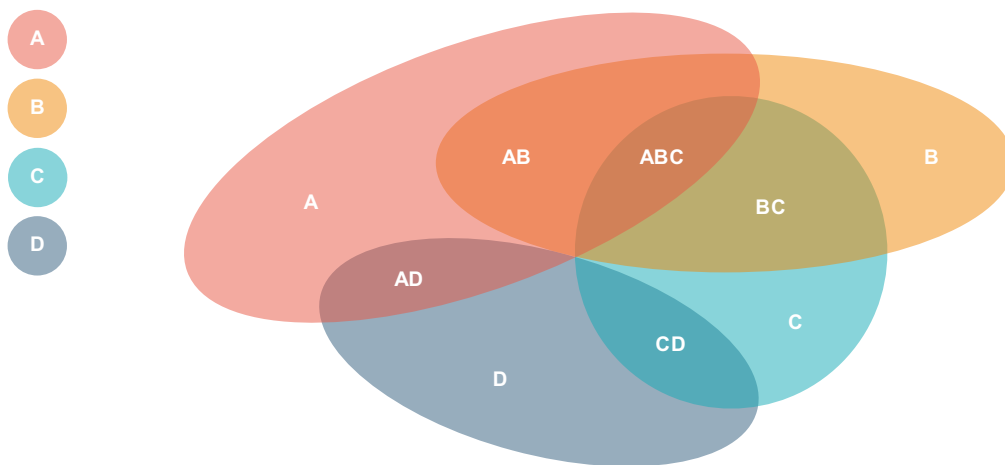
Use draw.io to create mockups of your application interfaces and mobile websites to optimize their usability, and keep your customers happy.



文氏图

Venn diagrams show the possible logical relationships between two or more groups of 'things' or sets. They are often used to organize groups of elements and show how they are similar or different. You will use them in scientific fields especially when classifying things, but they are also widely used in teaching, mathematics, linguistics, computer science and business.

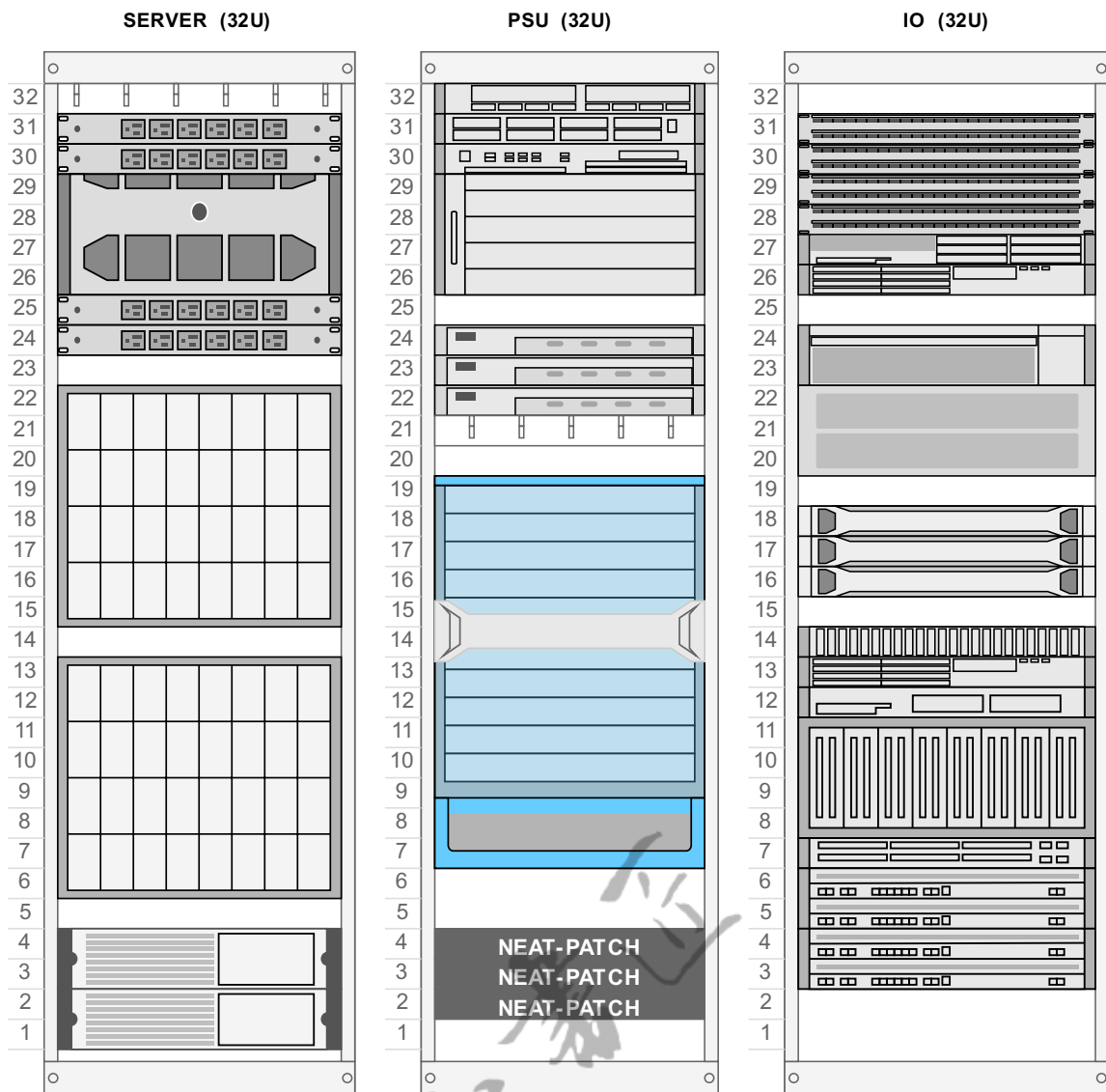
Use [draw.io](#) to help you classify your business plans, for illustrating the grouping of use cases, or for visually showing user access permissions in your IT infrastructure.



甘特图

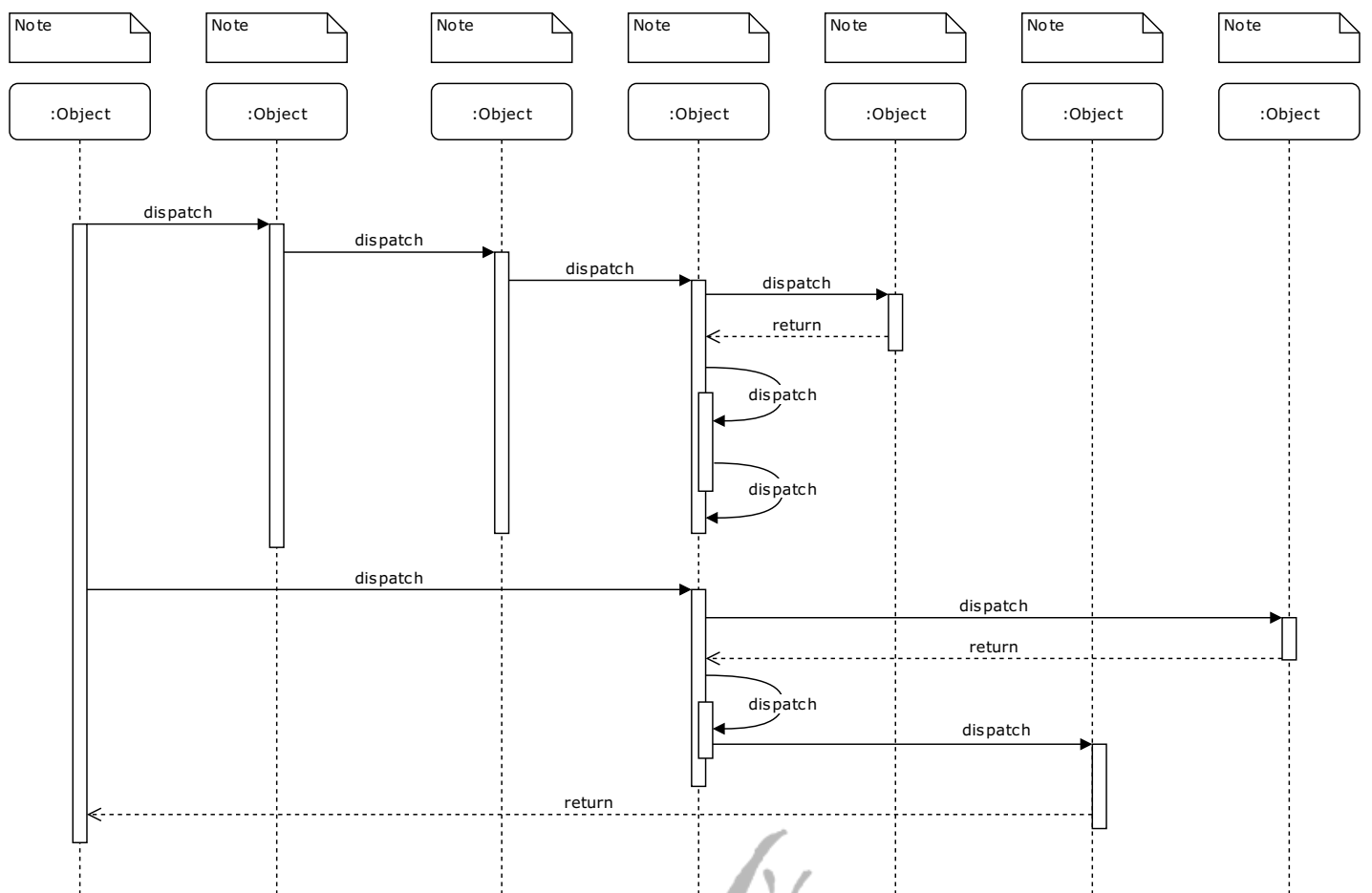


Rack diagrams are created by IT support personnel to visualize the rack mounting of computer and network equipment so they can plan and organize the equipment's arrangement before they actually buy or install anything. They show what the front (or rear) of the rack will look like and are drawn to scale. Make sure you have enough room for cabling, equipment and airflow by planning ahead with a rack diagram in draw.io.



序列图

You can use a sequence diagram to show how objects interact with each other and the order in which they act. Also known as event diagrams or event scenario, they are commonly used in documenting processes and in software engineering. Dotted lines (lifelines) indicate objects or roles, and arrows indicate messages passed between them. UML extends sequence diagrams to include components for conditional branches, optional processes and parallel processing.



相关

- ## 1. drawio 官网