
ThingJS 物联网 3D 可视化平台

部署及更新文档

2018 年 09 月 07 日

目录

1	说明	1
2	模型库部署	2
2.1	初次部署	2
2.1.1	部署条件	2
2.1.2	部署流程	3
2.2	模型资源更新	12
2.2.1	更新条件	12
2.2.2	更新流程	12
3	CAD 导入服务部署	13
3.1	部署条件	13
3.1.1	CAD 导入服务部署资源包	13
3.1.2	Jdk 工具	13
3.2	部署流程	13
4	JS 调试环境部署	15
4.1	部署条件	15
4.1.1	JS 调试环境部署包	15
4.1.2	http-server	15
4.2	部署流程	15
5	3D 场景搭建工具部署	16
5.1	部署条件	16
5.2	部署流程	16

1 说明

为实现阿里物联网云平台部署优锘科技（www.uinnova.cn）ThingJS 3D 可视化应用，完成阿里物联网云平台的物联网应用与 ThingJS 3D 可视化应用整合，特编写本文档，以说明 ThingJS 部署的相关步骤。

此次 ThingJS 3D 可视化部署包含以下方面：

- ThingJS 模型库部署：这是 ThingJS 3D 场景搭建工具 CampusBuilder 所用云端模型库，也是 ThingJS 调试环境引用模型的来源；
- ThingJS CAD 导入服务部署：CampusBuilder 所用 CAD 导入功能云端服务器；
- ThingJS 在线开发调试环境部署：3D 可视化应用脚本开发在线调试环境。

在阿里云完成以上部署后，阿里物联网云用户就可完成从 3D 场景搭建、上传、物联网注册设备关联、3D 可视化应用开发、在线部署整个过程。

服务部署架构如图 1 所示：

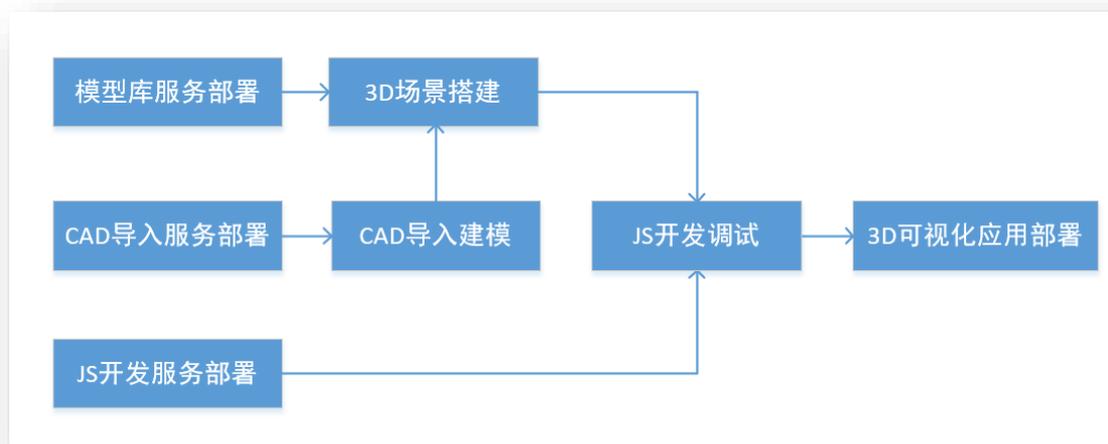


图 1 服务部署架构图

2 模型库部署

2.1 初次部署

2.1.1 部署条件

2.1.1.1 服务器

模型库部署前需准备能联网的服务器，最低参数如下：

数量：1 台

操作系统：Windows 2008 R2 及以上

内存：≥8G

2.1.1.2 模型库资源包

从优锆科技获取模型库资源包，模型库资源包内包括 4 类文件，分别是 models(模型文件)、textures(模型贴图)、5566.conf(配置文件)以及 BundleLib.json(索引文件)。



名称	修改日期	类型	大小
models	2018/9/5 17:08	文件夹	
textures	2018/9/5 17:08	文件夹	
5566.conf	2018/9/5 17:10	CONF 文件	3 KB
BundleLib.json	2018/8/31 10:51	JSON File	288 KB

图 2 模型库资源包内容

2.1.1.3 nginx 安装包

模型库部署前需下载 nginx 安装包，版本不低于 Windows-1.15.3，下载链接为：<http://nginx.org/en/download.html>

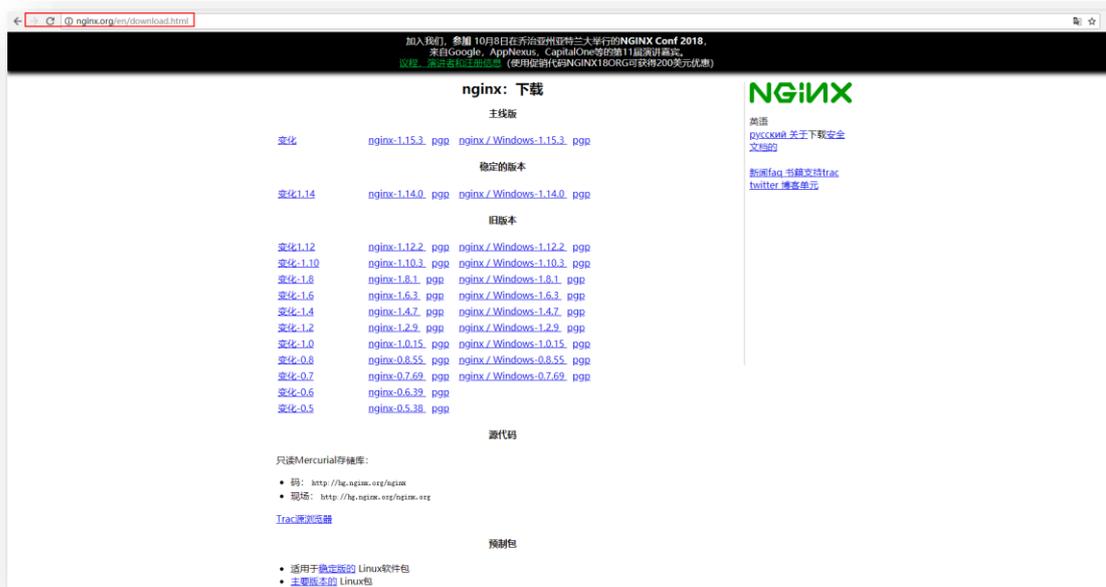


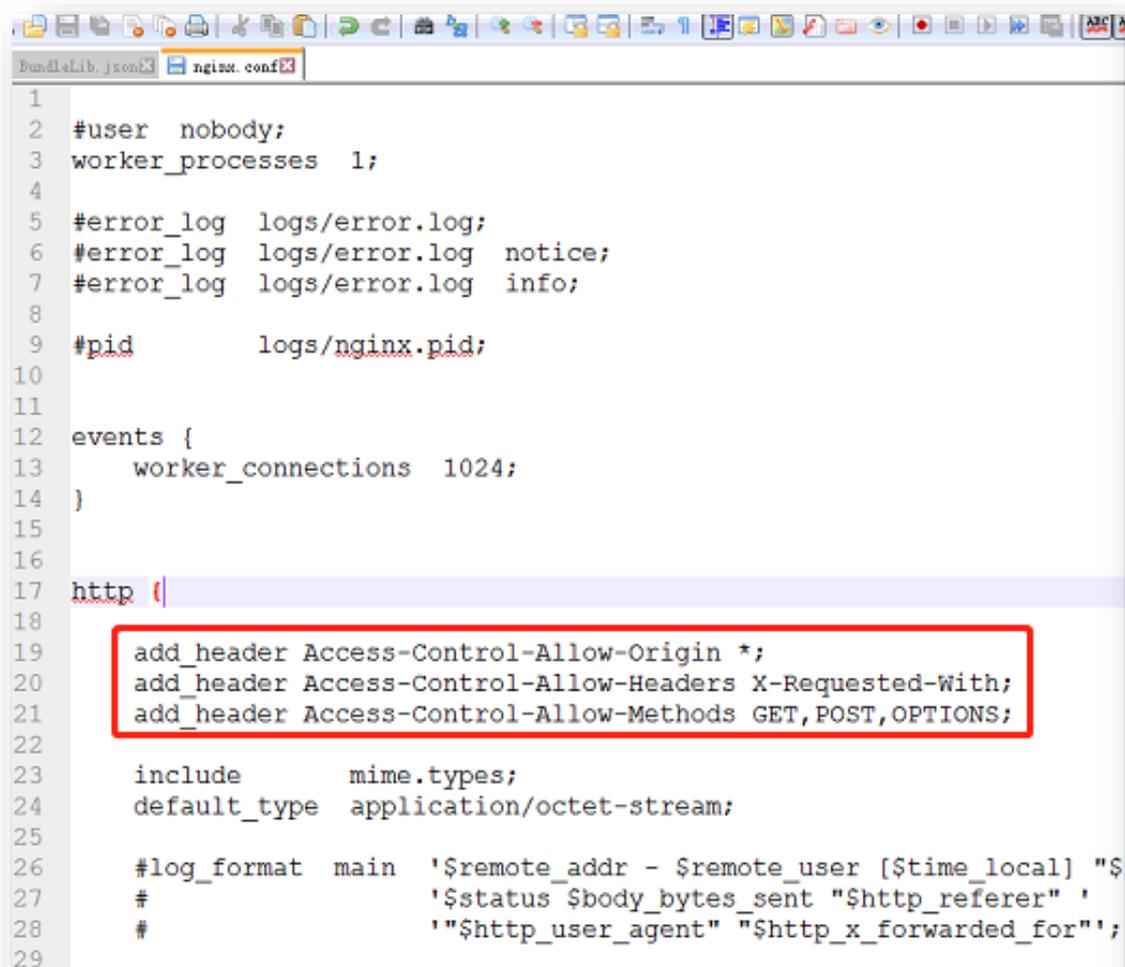
图 3nginx 安装包下载页面

2.1.2 部署流程

- 1) 在准备好的服务器硬盘下新建名称为 **BundleLib** 的文件夹，该文件夹不能存储在中文名称目录下；
- 2) 将准备好的模型库资源包解压到新建的 **BundleLib** 的文件夹目录下；
- 3) 将准备好的 **nginx** 安装包，解压在与 **BundleLib** 同级的目录下；
- 4) 打开解压后的 **nginx** 安装包，配置 **nginx** 服务，详细配置步骤如下：
 - a) 步骤一：打开 **nginx\conf\nginx.conf** 配置文件，配置跨域访问模型库，在 **http{}**中添加以下代码段：

```
add_header Access-Control-Allow-Origin *;
add_header Access-Control-Allow-Headers X-Requested-With;
add_header Access-Control-Allow-Methods GET,POST,OPTIONS;
```

详情如图 4 所示：



```
1
2 #user nobody;
3 worker_processes 1;
4
5 #error_log logs/error.log;
6 #error_log logs/error.log notice;
7 #error_log logs/error.log info;
8
9 #pid logs/nginx.pid;
10
11
12 events {
13     worker_connections 1024;
14 }
15
16
17 http {
18
19     add_header Access-Control-Allow-Origin *;
20     add_header Access-Control-Allow-Headers X-Requested-With;
21     add_header Access-Control-Allow-Methods GET,POST,OPTIONS;
22
23     include mime.types;
24     default_type application/octet-stream;
25
26     #log_format main '$remote_addr - $remote_user [$time_local] "$'
27     # '$status $body_bytes_sent "$http_referer" '
28     # '$http_user_agent' "$http_x_forwarded_for"';
29
```

图 4 配置跨域

- b) 步骤二：打开 `nginx\conf\nginx.conf` 文件进行编辑，在文档末尾最后一个大括号之前增加语句“`include vhost/*.conf;`”，并保存该文件，如图 5 所示：

```
92     #     }
93     #}
94
95
96     # HTTPS server
97     #
98     #server {
99     #     listen      443 ssl;
100    #     server_name localhost;
101
102    #     ssl_certificate      cert.pem;
103    #     ssl_certificate_key  cert.key;
104
105    #     ssl_session_cache    shared:SSL:1m;
106    #     ssl_session_timeout  5m;
107
108    #     ssl_ciphers  HIGH:!aNULL:!MD5;
109    #     ssl_prefer_server_ciphers  on;
110
111    #     location / {
112    #         root   html;
113    #         index index.html index.htm;
114    #     }
115    #}
116
117    include vhost/*.conf;
118
119 }
120
```

图 5 配置 nginx 服务

- c) 步骤三：在 nginx\conf\目录下新建名称为 vhost 的文件夹，如图 6 所示：

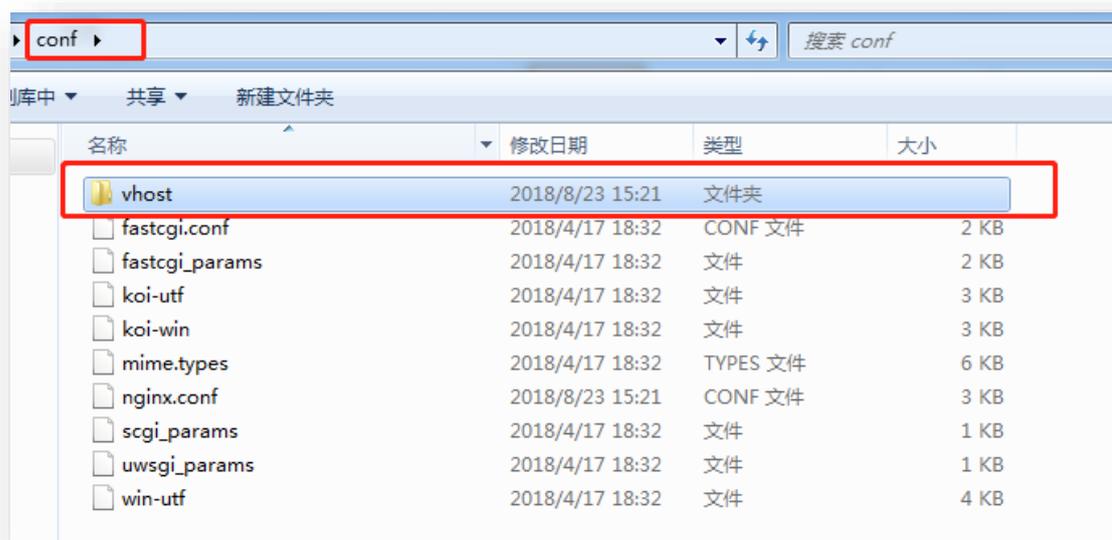


图 6 新建 vhost 文件夹

- d) 步骤四：将模型库资源包下的配置文件 5566.conf 剪切到 vhost 目录下，如图 7 所示：

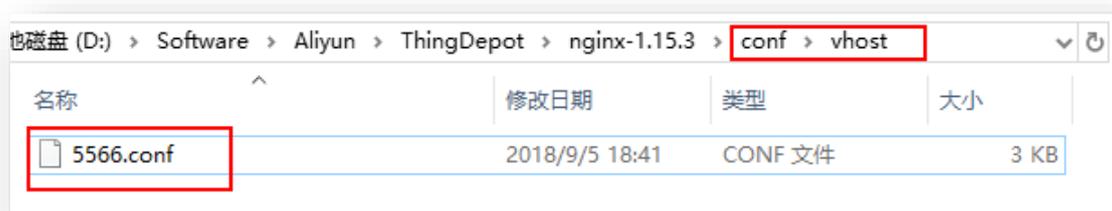


图 7 剪切配置文件

- e) 步骤五：打开“vhost”目录下配置文件 5566.conf 进行编辑，配置文件的内容如图 8 所示：


```
1 server{
2     #此处修改要监听的端口号
3     listen      5566;
4     #此处修改要监听的域名（localhost或者域名）
5     server_name localhost;
6     #控制全局nginx所有请求报文大小
7     client_max_body_size      50m;
8     location / {
9         # 禁用缓存
10        add_header Cache-Control no-store;
11        if ($request_method = 'OPTIONS') {
12            add_header 'Access-Control-Allow-Origin' '*';
13            add_header 'Access-Control-Allow-Credentials'
14            add_header 'Access-Control-Allow-Methods' 'GET
15            add_header 'Access-Control-Allow-Headers' 'DN
16            add_header 'Content-Type' 'text/plain charset
17            add_header 'Content-Length' 0;
18            add_header Content-disposition "attachment";
19            return 200;
```

图 10 域名配置界面

iii. 模型库 models 文件夹路径

（注意：此处的 alias 不能被删除；路径中的斜杠应为“/”而非“\”；末尾处必须有封号“;”）

```
31         add_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS';
32         add_header 'Access-Control-Allow-Headers' 'DNT,X-CustomHeader,Keep-Alive,User-Agent,
33
34     }
35     #nginx超时配置-链接
36     proxy_connect_timeout 30s;
37     #nginx超时配置-发请求
38     proxy_send_timeout 300s;
39     #nginx超时配置-读取
40     proxy_read_timeout 300s;
41 }
42 #错误请求返回页面
43 error_page 500 502 503 504 /50x.html;
44 location = /50x.html {
45     root html;
46 }
47 #配置模型文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中models文件夹！
48 location /BundleLib/model/ {
49     alias C:/modelslib/models/;
50     expires 7d;
51 }
52 #配置贴图文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中textures文件夹！
53 location /BundleLib/texture/ {
54     alias C:/modelslib/textures/;
55     expires 7d;
56 }
57 #配置模型库索引文件的正确访问，路径要换成服务器文件具体的存放路径！
58 location /BundleLib/ {
59     alias C:/modelslib/;
60     expires 7d;
61 }
62 }
63 }
```

图 11models 文件夹路径配置

iv. 模型库 textures 文件夹路径

（注意：此处的 alias 不能被删除；路径中的斜杠应为 “/” 而非 “\”；末尾处必须有封号 “;”）

```
34     }
35     #nginx超时配置-链接
36     proxy_connect_timeout 30s;
37     #nginx超时配置-发请求
38     proxy_send_timeout 300s;
39     #nginx超时配置-读取
40     proxy_read_timeout 300s;
41 }
42 #错误请求返回页面
43 error_page 500 502 503 504 /50x.html;
44 location = /50x.html {
45     root html;
46 }
47 #配置模型文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中models文件夹！
48 location /BundleLib/model/ {
49     alias C:/modelslib/models/;
50     expires 7d;
51 }
52 #配置贴图文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中textures文件夹！
53 location /BundleLib/texture/ {
54     alias C:/modelslib/textures/;
55     expires 7d;
56 }
57 #配置模型库索引文件的正确访问，路径要换成服务器文件具体的存放路径！
58 location /BundleLib/ {
59     alias C:/modelslib/;
60     expires 7d;
61 }
62 }
63 }
```

图 12textures 文件夹路径配置

v. 模型库索引文件位置

（注意：此处的 alias 不能被删除；路径中的斜杠应为 “/” 而非 “\”；末尾处必须有封号 “;”）

```
34     }
35     #nginx超时配置-链接
36     proxy_connect_timeout    30s;
37     #nginx超时配置-发请求
38     proxy_send_timeout       300s;
39     #nginx超时配置-读取
40     proxy_read_timeout       300s;
41 }
42 #错误请求返回页面
43 error_page 500 502 503 504 /50x.html;
44 location = /50x.html {
45     root    html;
46 }
47 #配置模型文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中models文件夹！
48 location /BundleLib/model/ {
49     alias C:/modelslib/models/;
50     expires 7d;
51 }
52 #配置贴图文件的正确访问，路径要换成服务器文件具体的存放路径，要到模型库中textures文件夹！
53 location /BundleLib/texture/ {
54     alias C:/modelslib/textures/;
55     expires 7d;
56 }
57 #配置模型库索引文件的正确访问，路径要换成服务器文件具体的存放路径！
58 location /BundleLib/ {
59     alias C:/modelslib/;
60     expires 7d;
61 }
62 }
63 }
```

图 13 索引文件路径配置

- 5) 双击 nginx 文件夹下的 nginx.exe 程序，启动 nginx 服务，如启动失败请返回检查；
- 6) 在浏览器中输入地址：<http://localhost:5566/bundleLib/BundleLib.json>（IP 及端口如和本文档设置不同，则修改成对应 IP 及端口号），如能正确显示索引目录即部署成功，索引目录如下图 14 所示：



图 14 访问模型库索引表

- 7) 打开 BundleLib 目录下的索引文件 BundleLib.json 进入编辑状态，修改配置 model_url 和 texture_url 路径为该服务器的相对路径，可使 CampusBulider 客户端的模型列表加载本次部署的模型库。

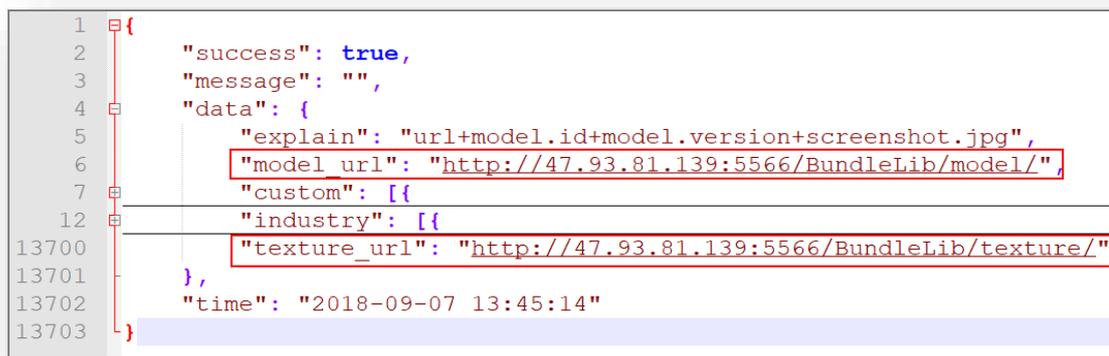


图 15 编辑索引文件

- 8) 完成模型库的初次部署。

2.2 模型资源更新

2.2.1 更新条件

在更新模型库资源包前，需要确保模型库资源包首次部署成功，预更新的模型库资源包格式同首次部署的模型库资源包一致，预更新的模型库资源包由优锘科技提供。

2.2.2 更新流程

更新模型库资源包前，请备份原模型库资源包。更新流程如下：

1) 替换 models 文件夹

将准备替换的 models 文件复制到 BundleLib 目录下，替换原 models 文件。

2) 替换 textures 文件夹

将准备替换的 textures 文件复制到 BundleLib 目录下，替换原 textures 文件。

3) 替换 BundleLib.json 配置文件

将准备替换的 BundleLib.json 文件复制到 BundleLib 目录下，替换原 BundleLib.json 文件。

3 CAD 导入服务部署

3.1 部署条件

3.1.1 CAD 导入服务部署资源包

获取 CAD 导入服务部署资源包 blackmirror.cad-0.0.1-SNAPSHOT，该服务部署资源包由优锘科技提供，资源包中的文件内容如下：

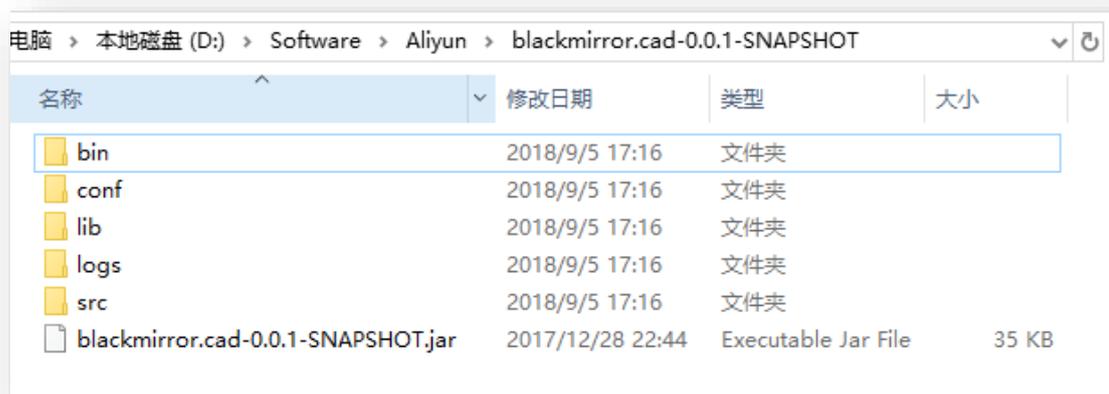


图 16CAD 导入服务部署包内容

3.1.2 Jdk 工具

安装 jdk，jdk 版本不低于 V1.7，jdk 的下载地址如下：

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

3.2 部署流程

- 1) 解压 CAD 部署服务中的 blackmirror.cad-0.0.1-SNAPSHOT.zip;
- 2) 修改/bin/start_console.sh 配置文件中 JAVA_HOME 变量值为 jdk 路径;

- 3) 修改 `/conf/blackmirror-cad.properties` 配置文件中的端口号“`jetty.ServerConnector.port`”，默认值为 58082；
- 4) 运行 `/bin/start.sh` 程序，启动 CAD 后台服务；
- 5) 启动成功后，可页面访问路径 <http://ip:port/pages/index.html>，例如：<http://192.168.1.179:58082/pages/index.html>

4 JS 调试环境部署

4.1 部署条件

4.1.1 JS 调试环境部署包

获取 JS 调试环境部署包，该部署包由优锆科技提供。

4.1.2 http-server

调试环境需要 http-server 支持。

4.2 部署流程

- 1) 将 JS 调试环境部署包解压；
- 2) 将解压后的 `dist` 和 `guide` 文件夹和 `edit.html` 文件，分别拷贝到 `http-server` 部署目录下或者 `OSS` 目录下（拷贝前需确保该目录下无同名目录）；
- 3) 输入 `OSS` 域名或启动 `http-server` 服务，浏览器访问 `edit.html`，如能正常打开即部署成功。

5 3D 场景搭建工具部署

5.1 部署条件

获取 3D 场景搭建工具 CampusBuilder 安装包，该安装包由优锘科技提供。

5.2 部署流程

将 3D 场景搭建工具 CampusBuilder 安装包部署在云端，生成下载链接，普通用户可通过该链接下载 3D 场景搭建工具 CampusBuilder。

3D 场景搭建工具 CampusBuilder 使用方法另见《CampusBuilder 用户手册》详细说明。