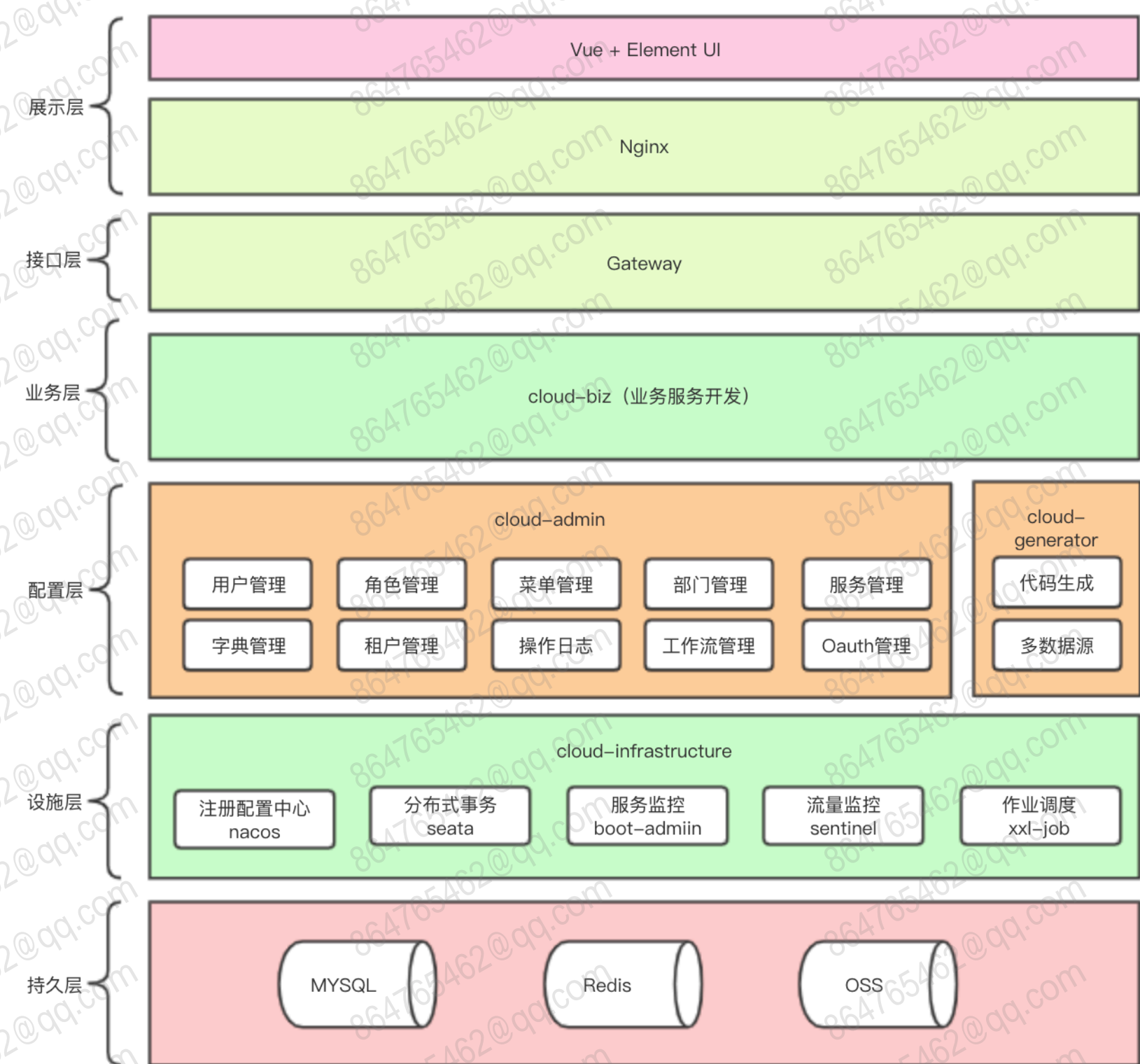


整体应用架构



基础准备

- 部署基础环境

```
mysql (其中dockr-compose已集成mysql)
docker
```

- 执行数据脚本

以下脚本已整合docker下的init.sql中



```
DB/cloud_admin.sql
DB/cloud_nacos.sql
```

后端镜像准备

- 编译基础环境

```
后端工程根目录: mvn clean install
```

- 构建基础镜像

1.修改根目录pom文件配置，true改为false
<docker.image.skip>>false</docker.image.skip>

```
cloud-admin: mvn clean package docker:build
cloud-gate: mvn clean package docker:build
cloud-monitor: mvn clean package docker:build
```

前端镜像准备

以下命令均在前端工程根目录下执行

- 打包命令

```
# 构建生产环境
npm run build:prod
```

- 构建前端镜像命令

```
docker build -t cloud/cloud-ui .
```

docker-compose 启动

- 后端工程路劲: /docker/docker-compose.yml
- 首次启动会下载大量镜像，勿慌
- 考虑代码生成器服务只在开发过程使用，没有集成到docker-compose

```
docker-compose up // 前台启动
docker-compose up -d // 后台启动
docker-compose down // 卸载容器
```

访问地址

```
核心应用地址: localhost:80
超级租户账户: super/super

Monitor应用: localhost:9999

Sentinel应用: localhost:8080
Sentinel账户: sentinel/sentinel

Nacos应用: localhost:8080/nacos
Nacos账户: naos/nacos
```