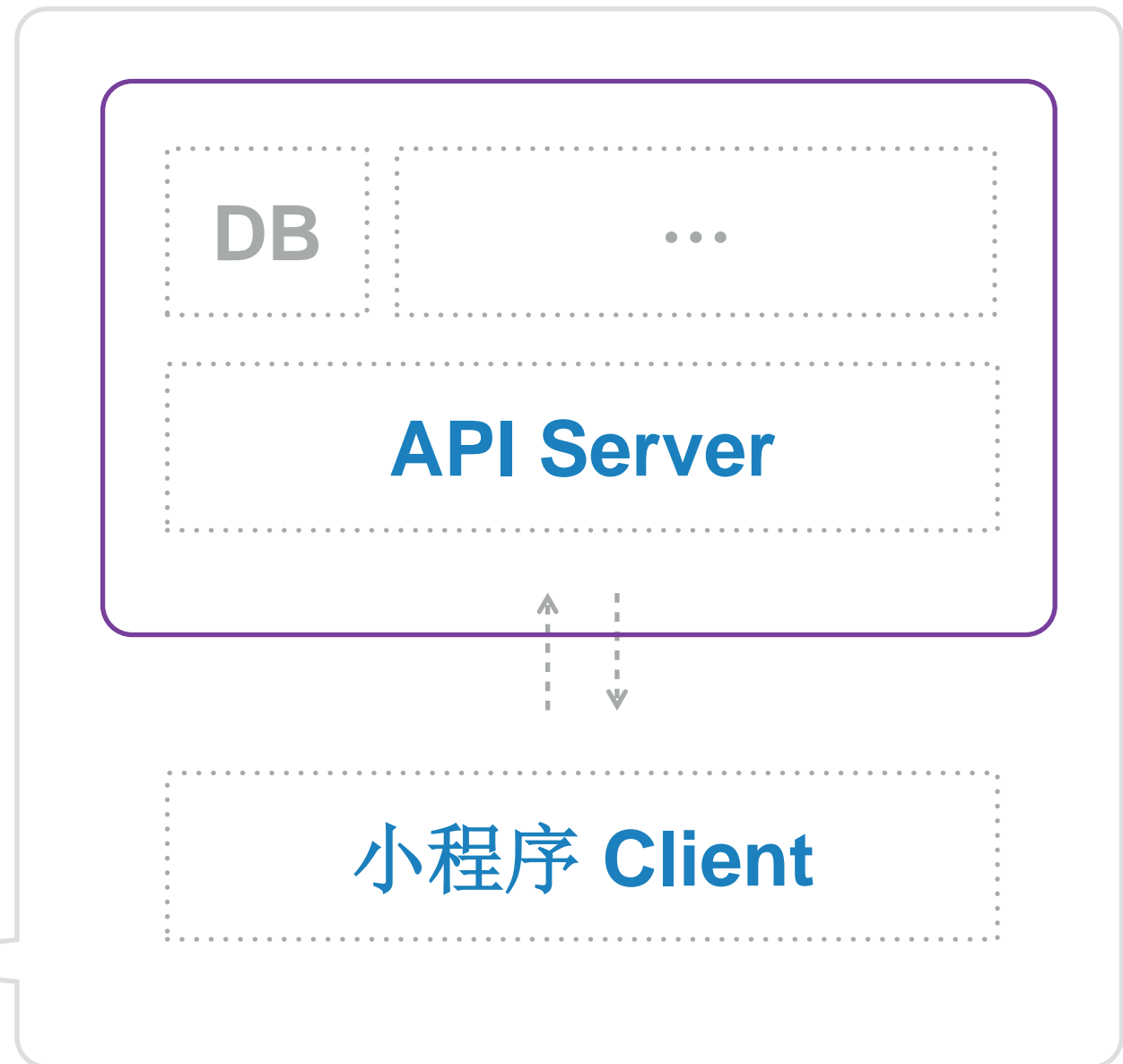


XpmJS

小程序后端开发工作的思考与实践

小程序开发





一、小程序与云计算

云端通信API

wx.request

文件上传下载

WebSocket

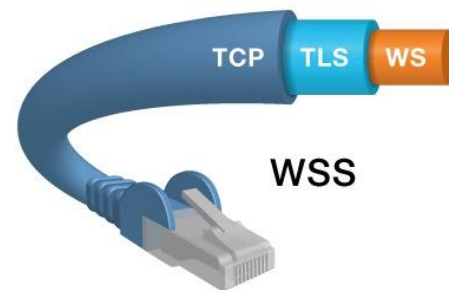
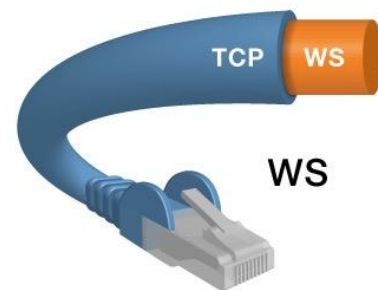
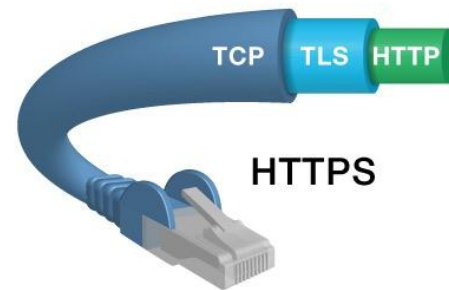
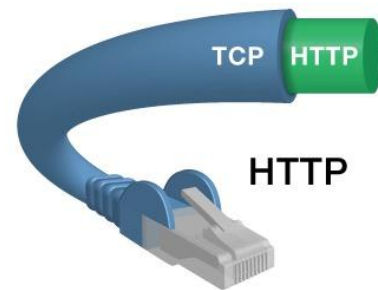


WSS & HTTPS



+ 购买 TLS/SSL 证书

+ 配置 Web 服务器



后端开发工作

搭建开发环境

实现 **Session** 鉴权

实现业务接口

微信接口

数据查询

实现管理后台



重复劳动应该交给机器去做！

- mayun

后端开发工作应该如何简化？



二、云计算的“进化”

云的作用

账号系统

数据存储

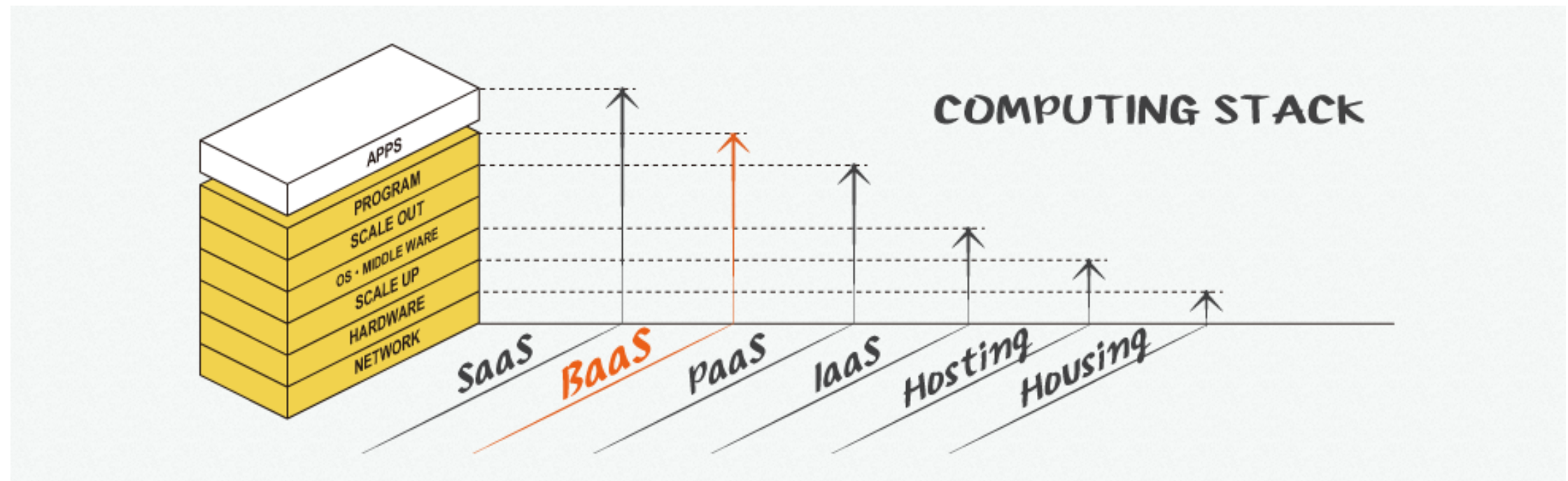
支付

即时通信

文件存储

...

从 IaaS 到 BaaS



	IaaS	PaaS	BaaS
后端编码	高	中	无需编码
交付产品	主机/网络/磁盘	服务	SDK + 后台
弹性伸缩	客户负责	厂商负责	厂商负责

国内外的BaaS平台



Parse Shutdown Reminder

The hosted Parse service is now shut down. To all the developers out there, it was a pleasure helping you build apps. Goodbye and good luck!

Parse Server

The open source version of the Parse backend, and related projects.

parse-server Parse-compatible API server module for Node/Express	JS	★ 12916	🔗 3499
parse-server-example Example server using Express and the parse-server module.	JS	★ 1347	🔗 3942
parse-server-simple-mailgun-adapter Used to send Parse Server password reset and email verific...	JS	★ 40	🔗 70
parse-server-modules Community Organization for non-core Parse Server modules and adapters			



“我们意识到 Parse 这件事会对开发者对于第三方后端服务的信心造成一些负面影响，但归根到底，成败在于我们自己是不是能提供优秀的产品和服务。我们最近正在落实一些提高服务稳定性、提高 SDK 品质的技术和流程上的改进。”

— 江宏 LeanCloud CEO

我们需要怎样的云服务？

1. 可以掌控数据，最好没有迁移成本。
2. 不用做 Web 服务器配置之类和业务无关的事儿。
3. 不用写后端代码，节省开发时间（成本）。
4. 立即上手，不用学习一堆新东西。

可独立部署、开通即用的轻量 **BaaS** 平台



三、搭建“云桥” – XpmJS的实践

XpmJS

XpmJS 可链接任何云计算资源，为小程序提供云端支撑。开发者无需编写后端代码，即可实现用户登录、WebSocket 通信、微信支付、云端数据表格、文件存储等功能。



小程序 **SDK**

XpmJS Server



小程序SDK

信道 **WSS**

支付 **Pay**

云端队列 **Que**

用户 **User**

会话 **Session**

工具 **Utils**

云表格 **Table**

本地存储 **Stor**

云端应用 **App**

小程序SDK | 用户 User

```
var user = app.xpm.require('User');

user.login().then( function( userInfo ) {

    console.log( '用户登录成功', userInfo );
    app.session.set('loginUser', userInfo );
})

.catch( function( excp ) {
    console.log('用户登录失败', excp );
});
```

wx.login

/login API

User Data

User.logout()

User.get()

...

小程序管理 数据管理 > DEMO_USER

demo_user 选择数据表

共找到 4 条符合条件的数据 + 插入

数据 ID	UNIONID	OPENID	NICKNAME	GENDER	LANGUAGE	CITY	AVATARURL
2	Ksa3ks83adkjsahgjsj22881jsaja	oJcP50F3aoEfSH6ggdAuVxP7-08U	测试用户	*1*	zh_CN	NULL	http://wx.qlogo.cn/mmopen/vi_32/cvicMAEjn3xy9WjqgtiasYyc
3	o7dWTuD6gzJMbWKFEnMQRakGPo2Q	onHK_luJxXJ_N0uF0QBT1QkF3BkQ	测试用户	NULL	zh_CN	NULL	http://wx.qlogo.cn/mmopen/RKf4zQJGBZcO6dibiaDEWcZzec
4	NULL	oJcP50H1BnH_Re6iXG0f5I-eGRN0	测试用户	*1*	zh_CN	Haidian	http://wx.qlogo.cn/mmopen/vi_32/Q3auHgzwzM7MDcQ9Iribia
5	NULL	o7h_r0M7OSVui_iOLw8LLY4Sc16o	测试用户	*1*	zh_CN	Haidian	http://wx.qlogo.cn/mmopen/vi_32/Q3auHgzwzM6ZegyY80fKq

小程序SDK | 支付 Pay

```
var pay = app.xpm.require('Pay');
pay.request({
  total_fee:500, // 单位分
  body:'算命、服务器开光',
  attach:'HELLO XpmJS.com',
  detail:'{id:888,desp:"算命,抽SSR,赠送服务器开光"}'
}).then(function( data ){
  console.log('Request Pay Success', data );
}).catch( function( excp){
  console.log('Request Pay Failure', excp );
});
```

/unifiedorder API

wx.requestPayment

/payreturn API

Order data

云端事件

before()

success()

...

小程序管理 数据管理 > DEMO_SYS_PAYLOG

demo_sys_paylog 选择数据表

共找到 40 条符合条件的数据

数据 ID	SN	UNIONID	OPENID	OUT_TRADE_NO	PREPAY_ID	ATTACH	RETURN_CODE	创建时间	修改时间
1	*1484565246947239*	NULL		*20170116191405027338*	wx2017011619140658b892fd7c0446761438	attach user is 1	SUCCESS	2017-01-16 19:14:06	
2	*1484565646412901*	NULL		*20170116192046021920*	wx201701161920461fc8c4c7310027004401	attach user is 1	SUCCESS	2017-01-16 19:20:46	
3	*1484565798274017*	NULL		*20170116192318019949*	wx20170116192318a9c304111e0720625400	attach user is 1	SUCCESS	2017-01-16 19:23:18	
4	*1484566029176336*	NULL		*20170116192708059752*	wx20170116192709cb5d482b340104985087	attach user is 1	SUCCESS	2017-01-16 19:27:09	
5	*1484566190542772*	NULL		*20170116192950016328*	wx2017011619295080d57e9de00255824062	attach user is 1	SUCCESS	2017-01-16 19:29:50	

小程序SDK | 云表格 Table

```
var table = app.xpm.require('Table', 'hello');
table._schema(
  [
    {name:"name", type:'string', option:{length:80, require:true }, acl:"rwd:r:-" },
    {name:"company", type:'string', option:{length:100}, acl:"w:-:-" }
  ],
  { record:"rwd:r:-", table:"rwd:-:-", field:'rwd:r:-', user:'admin', group:'member' }
, true ).then( function( data ) {
  console.log('数据表创建成功', data );
})
.catch( function( excp ) {
  console.log('数据表创建失败', excp );
});
```

```
var table = app.xpm.require('Table', 'hello');
// 创建
table.create(
  {name:'张艺谋', company:'中国电影制片厂'}
).then(function(data) { // 更新
  return table.update(data['_id'], {name:'冯小刚'});
}).then(function(data) { // 读取
  return table.get(data['_id']);
}).then(function(data) { // 删除
  return table.remove(data['name'], 'name' );
}).then(function(resp) {
  console.log( 'remove success', resp );
}).catch( function( excp ) {
  console.log('出错了', excp );
});
```

```
var table = app.xpm.require('Table', 'hello');
table.query()
  .where('name', '=', '冯小刚')
  .orderBy('name', 'asc')
  .limit(2) // 仅查询 2条

.fetch('name','company').then(function(data) {
  console.log( '查询结果', data );
})

table.query()
  .where('name', '=', '冯小刚')
  .orderBy('name', 'asc')
  .paginate(3, 2) // 分3页, 当前显示第 2页

.fetch('name','company').then(function(data) {
  console.log( '查询结果', data );
});
```

小程序管理 数据管理 > DEMO_BOOKING

demo_booking 选择数据表

共找到 8 条符合条件的数据

数据ID	SN	UID	GOODS	WHEN	STATUS	STATUS_TIPS	REMARK	创建时间	修改时间
1	'1484565646412901'	1	{ "1": { "id": 1 } }	2017-02-12 12:00	DONE	*3333*	*3333*	2017-01-18 23:03:21	2017-01-18 23:27:34
2	'2484565646412902'	1	{ "2": { "id": 2 } }	2017-02-13 13:00	CANCEL	NULL	NULL	2017-01-18 23:03:21	2017-01-18 23:36:35
3	'3484565646412903'	1	{ "3": { "id": 3 } }	2017-02-10 10:00	SUCCESS	房间喝喝酒	还去上回的那个房间	2017-01-18 23:03:21	2017-01-18 23:40:20
4	'4484565646412904'	1	{ "8": { "id": 8, "price": 1, "amount": 1 } }	2017-2-18 22:00	WAIT_CFM	NULL	NULL	2017-01-18 23:03:21	
5	'5484565646412905'	1	{ "9": { "id": 9, "price": 1, "amount": 1 } }	2017-2-18 14:00	SUCCESS	OK	还去上回的那个房间	2017-01-18 23:03:21	
6	'6484565646412906'	1	{ "6": { "id": 6 } }	NaN-NaN-NaN NaN:NaN	FAIL	NULL	还去上回的那个房间	2017-01-18 23:03:21	2017-01-30 11:49:40
7	'7484565646412907'	1	{ "11": { "id": 11, "price": 1, "amount": 1 } }	2017-4-28 15:00	DONE	NULL	还去上回的那个房间	2017-01-18 23:03:22	
8	'8484565646412908'	1	{ "8": { "id": 8 } }	2017-05-16 16:00	FAIL	NULL	还去上回的那个房间	2017-01-18 23:03:22	2017-01-18 23:36:24

支持ACL鉴权，支持JSON字段，支持join, inWhere 等多种查询。

小程序SDK | 信道 WSS

```
var wss = app.xpm.require('Wss');
wss.open('/wxapp').then(function( res ) {
    console.log( '信道连接成功', res );
})
.catch( function( excp ) {
    console.log('信道连接失败', excp );
});
```

```
var wss = app.xpm.require('Wss');
wss.listen('payment', function( res, status ){
    // 当接收到 payment 指令后运行
    if ( status !== 'success') return ;
    console.log( res, status );
});
```

```
var wss = app.xpm.require('Wss');
wss.liveUsers().then(function( users ) {
    console.log( '读取在线用户成功', users );
    // 向第一个用户发送 payment 指令
    if ( users.length > 0 ) {
        return wss.send('payment', users[0], users[0]['id'] )
    } else {
        return {code:404, message:'no live user'};
    }
}).then( function( res ){
    console.log('发送完毕', res);
});
.catch( function( excp ) {
    console.log('出错了', excp );
});
```

The screenshot shows the '信道调试器' (Channel Debugger) interface for a WeChat Mini Program. The top navigation bar includes '小程序管理' (Mini Program Management), '信道管理' (Channel Management), and '信道调试器' (Channel Debugger). The main area displays the connection URL 'wss://wss.xpmjs.com/ws-server/wxapp'. Below the URL are several control buttons: '在线用户' (Online Users), 'SessionID', 'PING ALL', '指令 A' (Command A), '指令 B' (Command B), '指令 C' (Command C), 'Face 花' (Face Flower), 'Face 开心' (Face Happy), and 'Face 转' (Face Turn). The '输出日志' (Output Log) section shows a JSON response from the 'getConnections' command. The '在线用户' (Online Users) section shows a single user with a cartoon avatar and the text '未登录用户 (1)' (Unlogged User (1)) and '未知' (Unknown). The bottom right corner features a heart icon and the text '感谢采用团队猫' (Thank you for using Team Cat).

小程序管理 信道管理 > 信道调试器

wss://wss.xpmjs.com/ws-server/wxapp

在线用户 SessionID PING ALL 指令 A 指令 B 指令 C Face 花 Face 开心 Face 转

输出日志:

```
{
  "c": "getConnections",
  "b": []
}
```

在线用户

未登录用户 (1)
未知

感谢采用团队猫

XpmJS Server



```
docker run -d --name=xpmjs-server \
  -e "HOST=yourdomain.com" \
  -v /host/data:/data \
  -v /host/apps:/apps \
  -v /host/config:/config \
  -p 80:80 -p 443:443 \
  tuanduimao/xpmjs-server:1.0rc5
```

Nginx 1.9.7

PHP 7.0.9

MariaDB 10.1.14

Redis 3.2.2

团队猫 XPM 版 1.0rc5



团队猫 XpmJS-Server 1.0RC5 独立数据库版

提供商: 北京云道天成科技有限公司

最新版本: V1.0RC5-NODB

操作系统: Ubuntu S...

功能简介: XpmJS后端服务器, 采用团队猫企业应用框架构建, 本镜像为 XpmJS 服务端系统。XpmJS为微信小程序、微信公众号开发提供云端能力。无需编写后端代码, 即可实现用户登录、WebSocket 通信、微信支付、云端数据表格、文件存储等功能。

优惠价: 免费



团队猫 XpmJS-Server 1.0RC5

提供商: 北京云道天成科技有限公司

最新版本: V1.0RC5

操作系统: Ubuntu S...

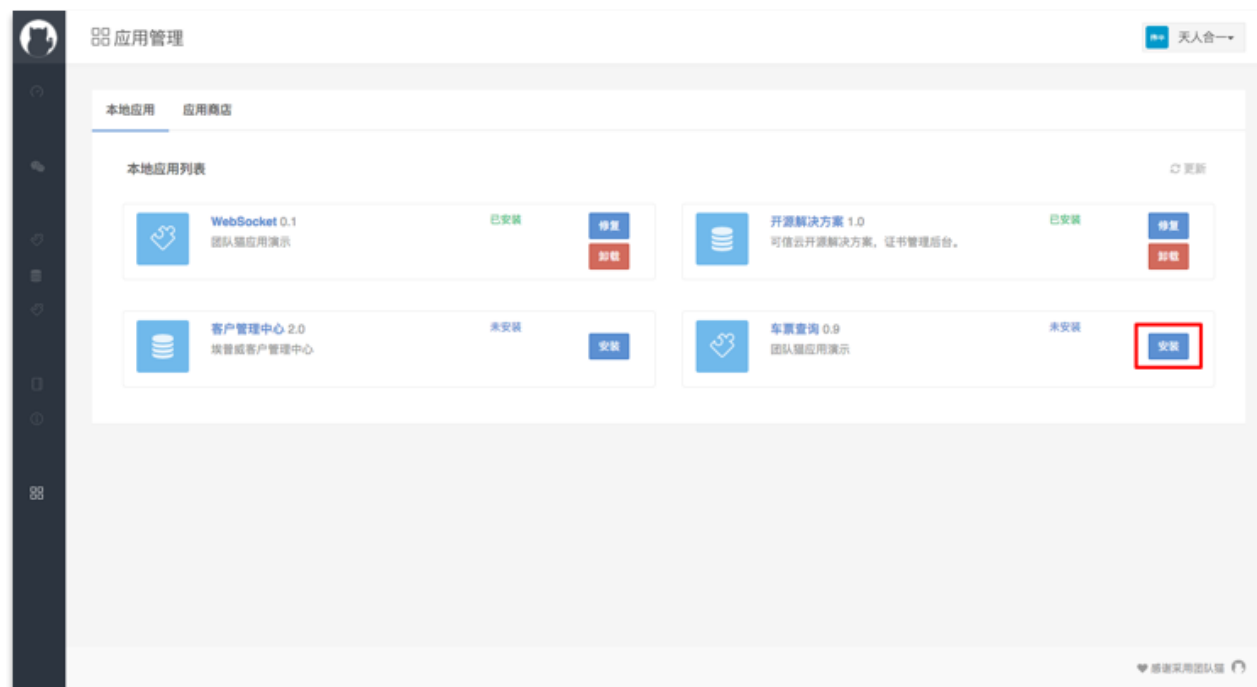
功能简介: XpmJS后端服务器, 采用团队猫企业应用框架构建, 本镜像为 XpmJS 服务端系统。XpmJS为微信小程序、微信公众号开发提供云端能力。无需编写后端代码, 即可实现用户登录、WebSocket 通信、微信支付、云端数据表格、文件存储等功能。

优惠价: 免费

腾讯云服务市场已上架

XpmJS Server | 自由扩展

```
sudo docker exec -it xpmjs-server /bin/bash # 切换到容器 shell 环境
mkdir -p /apps/myapp
cd /apps/myapp # 进入应用目录
tdm app init # 初始化, 生成 MVC 框架, 程序配置文件等
exit; # 退出容器
```



```
/**
 * 上传文件接口
 * @return [type] [description]
 */
function upload() {

    $file = !empty($_FILES['wxfile']) ? $_FILES['wxfile'] : [];
    $content = !empty($file['content']) ? base64_decode($file['content']) : NULL;
    if ( $content == null ) {
        echo (new Excp("未接收到文件", 502, ['_FILES'=>$_FILES]))->toJSON();
        return;
    }

    $config = $GLOBALS['_QC']['cos'];
    $cos = App::M('Cos', $config);
    try {
        $resp = $cos->upload( $content, ['region'=>$config['region'], 'filetype'=>'png'] );
    } catch( Excp $e ){
        $extra = $e->getExtra();
        echo (new Excp("COS API 错误", 502, ['resp'=>$resp, 'e'=>$extra]))->toJSON();
        return;
    }
    if ( $resp['code'] == 0 ) {
        Utils::out($resp['data']);
        return;
    }
    echo (new Excp("COS API 错误", 502, ['resp'=>$resp]))->toJSON();
    return;
}
```

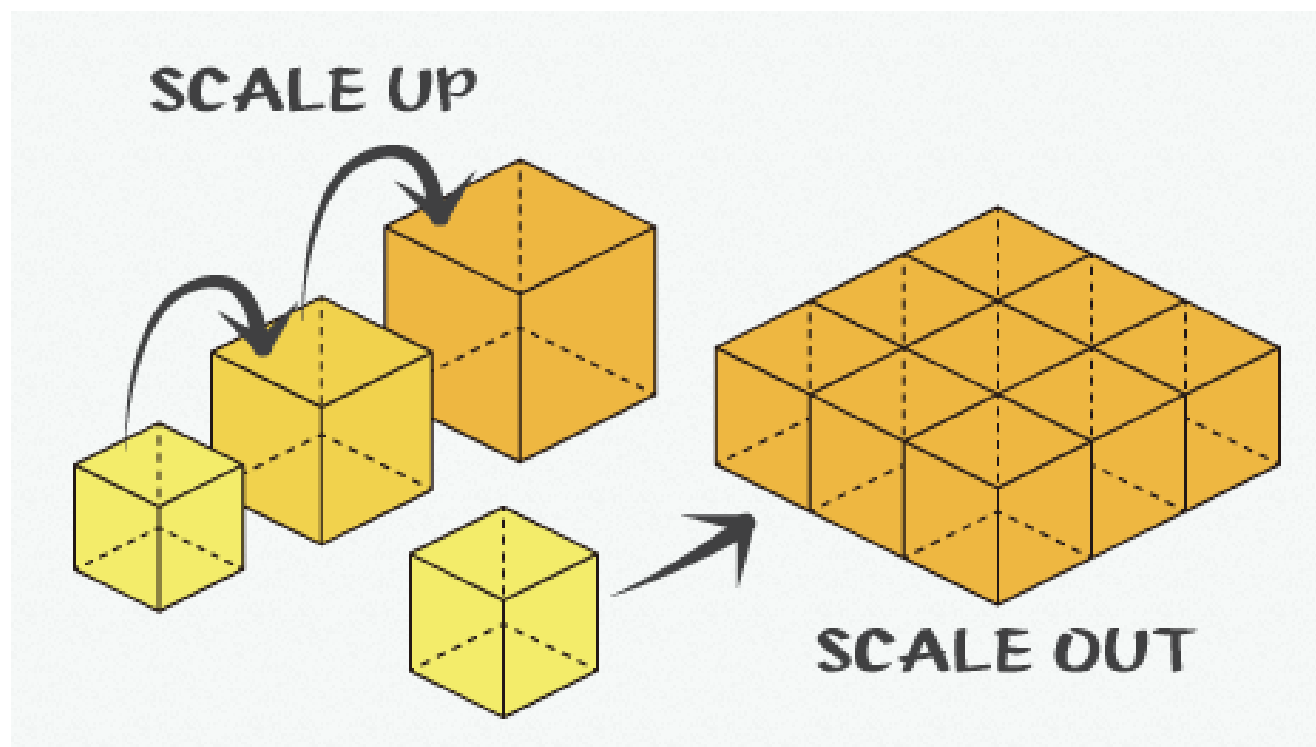
```
success: function (res) {
    var tempFilePaths = res.tempFilePaths;
    var qcloud = app.xpm.require('app', 'xqcloud');

    qcloud.api("cos", 'upload')
        .upload( tempFilePaths[0] )

        .then(function(data){
            that.setData({
                'rs.corver': data.access_url,
                'rs.images': [data.access_url]
            });
        })
        .catch( function(excp){
            console.log('Upload Fail', expc );
        });
}
```

对于一些机密数据, 比如API的 Secret KEY, 需要保存在云端。对于这部分需求, 可以通过编写 XpmJS 云端应用来实现。

XpmJS Server | 支持分布式部署



```
"storage": {
  "engine": "Mysql",
  "prefix": "{auto}",
  "option": {
    "db_name": "tuanduimao",
    "master": [{
      "host": "127.0.0.1",
      "port": "3307",
      "user": "root",
      "pass": "123456"
    }],
    "slave": [{
      "host": "127.0.0.1",
      "port": "3307",
      "user": "root",
      "pass": "123456"
    }]
  }
},
"search": {
  "engine": "Elasticsearch",
  "index": "devtuan"
```

XpmJS Server 支持分布式部署，通过修改配置文件即可添加多台前端，提升后端服务能力。



四、XpmJS RoadMap

XpmJS SDK



小程序



+ JS SDK



+ Android



+ iOS

XpmJS Server

+ NoSQL 数据存储服务

+ 消息推送服务

+ 直播流媒体服务

+ 统计分析

+ XpmJS Server Hosting

+ 快速开发移动应用培训

谢谢大家

欢迎加入XpmJS 开源项目，共同探索移动应用快速开发之路



腾讯云 小程序培训



我的微信