

# 产融云第三方发货申请接口

## 1.功能说明

第三方发货接口

请求方式：POST

## 2.请求路径

环境	地址
正式环境	https://[需替换]工行API网关地址/api/ecp/order/delivery/third/apply/V1

## 3.通用请求参数

参数名	类型	是否非空	最大长度	描述	示例值
app_id	str	true	20	APP的编号,应用在API开放平台注册时生成	10000000000000005402
msg_id	str	true	40	消息通讯唯一编号,每次调用独立生成,APP级唯一	1506504665696
format	str	false	5	请求参数格式,仅支持json	json
charset	str	false	10	字符集,缺省为UTF-8	UTF-8
sign_type	str	false	5	签名类型,CA-工行颁发的证书认证,RSA-RSAWithSha1,RSA2-RSAWithSha256,缺省为RSA	RSA
sign	str	true	300	报文签名	-
timestamp	str	true	19	交易发生时间戳,yyyy-MM-dd HH:mm:ss格式	2022-03-29 20:44:38
biz_content	str	true	-	请求参数的集合	参看#8 请求示例#

## 4.请求参数 (biz\_content节点内容)

参数名	类型	是否必输	最大长度	描述	示例值
appId	String	true	20	APP的编号,应用在API开放平台注册时生成	100000000000000005402
vendorId	String	true	30	商户编号	SH00001
userId	String	true	30	用户编号	020230930000679803
userName	String	false	30	用户名	XY230918
roleId	String	true	20	角色ID, 该场景固定为超管-ROLE2000	固定ROLE2000
expressType	String	true	2	发货类型 1-全部 2-部分, 3-自主发货	1
fromFlag	String	true	2	操作员类型, 该场景固定为0-第三方	0
clientIp	String	false	30		
list	List	true	20	订单配送信息	0

订单配送信息 (OrderDeliveryDto内容)

字段名	类型	是否必输	最大长度	描述	备注
orderId	String	是	20	订单编号	
deliveryId	String	否	18	配送单号	
merchant2memberOrderMemo	String	否	100	给买家备注	
deliveryNotes	String	否	100	运单备注/申请原因	自主发货时必填
expressProdList	List	是	10	商品发货数量信息	
skuProdStr	String	否		发货分组编号	
grossWeight	String	否	15	发货重量	数字格式, 例 15.5
deliveryMerchantId	String	否		物流公司编号	
consigneeName	String	否		收件人名称	
consigneeMobile	String	否		收件人电话	
consigneeProvince	String	否		收件人省份	
consigneeCity	String	否		收件人城市	
consigneeCounty	String	否		收件人区县	
consigneeProvinceId	String	否		收件人省份编号	
consigneeCityId	String	否		收件人城市编号	
consigneeCountyId	String	否		收件人区县编号	
deliveryImageUrl	String	否		收件图片URL	
consigneeAddressDetail	String	否		收件人详细地址	

字段名	类型	是否必输	最大长度	描述	备注
deliveryFee	String	否		修改后运费金额	
applyItem	String	否		申请项	自主发货时必填
applyAmount	BigDecimal	否		自主发货-申请金额	自主发货时必填

expressProdDto包含以下字段

字段名	类型	最大长度	描述	是否必填	备注
productId	String		商品编号	是	
prodSkuld	String		商品SKU编号	是	
productNum	String		商品数量	是	
batchNum	String		商品批次号	否	

## 5.通用响应参数

参数名	类型	是否非空	最大长度	描述	示例值
response_biz_content	String	true	-	响应参数集合,包含公共和业务参数	-
sign	String	true	300	针对返回参数集合的签名	

## 6.响应参数说明 (response\_biz\_content节点内容)

参数名	类型	是否非空	最大长度	描述	示例值
return_code	String	true	-	返回码	
return_msg	String	true	300	返回码说明	
data	Map	false	-	结果信息	

data包含:

参数名	类型	是否非空	最大长度	描述	示例值
orderId	String	true	20	订单编号, map 的key	O0001
result	String	true		发货结果, map的value	success

## 7.使用示例

```
public class TestRun {
    private static final ObjectMapper OBJECT_MAPPER = new ObjectMapper();

    protected static final String MY_PRIVATE_KEY = "[需替换]应用方私钥，由应用方生成公
私钥对，并将公钥上传至工行API开放平台";

    protected static final String APIGW_PUBLIC_KEY = "[需替换]工行API网关公钥，请向工
行联系获取";

    protected static final String APP_ID = "*****";

    @Test
    public void test_orderDelivery() throws IcbcApiException,
JsonGenerationException, JsonMappingException, IOException {
        // 公私钥与AES
        DefaultIcbcClient client = new DefaultIcbcClient(APP_ID, "RSA",
MY_PRIVATE_KEY, "UTF-8",
            "json", APIGW_PUBLIC_KEY, "AES", "*****", "", "");
        OrderDeliveryApplyRequestV1 request = new OrderDeliveryApplyRequestV1();
        // APIP路由入口
        request.setServiceUrl("[需替换]工行API网关地址 +
/api/ecp/order/delivery/third/apply/v1");
        // 组织参数
        OrderDeliveryApplyRequestV1Biz bizContent = new
OrderDeliveryApplyRequestV1Biz();

        bizContent.setAppId("10000000000000005402");
        bizContent.setVendorId("xxx");
        bizContent.setUserId("020230930000679803");
        bizContent.setExpresstype("2");
        OrderDeliveryDto orderDeliveryDto = new OrderDeliveryDto();
        orderDeliveryDto.setOrderId("210825900000008536");
        orderDeliveryDto.setConsigneeName("zhangsan");
        orderDeliveryDto.setConsigneeMobile("18888888888");
        orderDeliveryDto.setConsigneeProvince("beijing");
        orderDeliveryDto.setConsigneeProvinceId("000000");
        orderDeliveryDto.setConsigneeCity("beijing");
        orderDeliveryDto.setConsigneeCityId("001000");
        orderDeliveryDto.setConsigneeCounty("haidian");
        orderDeliveryDto.setConsigneeCountyId("001010");
        orderDeliveryDto.setConsigneeAddressDetail("xx街道xx号");
        ExpressProdDto expressProdDto = new ExpressProdDto();
        expressProdDto.setProductId("P001");
        expressProdDto.setProductSkuId("S001");
        expressProdDto.setProductNum("10");
        orderDeliveryDto.setExpressProdList(Arrays.asList(expressProdDto));
        List<OrderDeliveryDto> list = Arrays.asList(orderDeliveryDto);
        bizContent.setList(list);
        //将bean转为String类型json串的工具，可替换任意该工具
        String jsonStr = OBJECT_MAPPER.writeValueAsString(bizContent);
        request.setBizContent(jsonStr);
    }
}
```

```

OrderDeliveryApplyResponseV1 response;
try {
    response = client.execute(request, System.currentTimeMillis() + "");
    System.out.println(JSONObject.toJSONString(response));
    if (response.isSuccess()) {
        // 业务成功处理
        System.out.println(response.getReturnCode());
    } else {
        // 失败
    }
} catch (IcbcApiException e) {
    e.printStackTrace();
} catch (Exception e) {
    System.out.println(e);
}
}
}

```

## 8.请求示例

```

POST HTTP/1.1
Content-Type: application/x-www-form-urlencoded; charset=UTF-8

https://gw.icloud.icbc.com.cn/api/ecp/order/delivery/third/apply/v1?
app_id=0ikec1o001&msg_id=urcn124ciutr9&format=json&charset=utf-
8&sign_type=RSA&sign=TRFEWHYUFCEW&timestamp=2017-10-29 20:44:38&biz_content=
{
  "appId": "1000000000000005402",
  "vendorId": "xxx",
  "userId": "020230930000679803",
  "expressType": "2",
  "list": [{
    "orderId": "21082590000008536",
    "consineeName": "zhansan",
    "consigneeMobile": "18888888888",
    "consigneeProvince": "beijing",
    "consigneeProvinceId": "000000",
    "consigneeCity": "beijing",
    "consigneeCityId": "001000",
    "consigneeCounty": "haidain",
    "consigneeCountyId": "001010",
    "consigneeAddressDetail": "xx街道xx号",
    "expressProdList": [
      {
        "productId": "P001",
        "productSkuId": "S001",
        "productNum": "10"
      }
    ]
  }
}
}

```

## 9.响应示例

---

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
{
  "response_biz_content": {
    "return_code": 0,
    "return_msg": "success",
    "data": {
      "21082590000008536": "success"
    }
  }
}
```

## 10.异常示例

---

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
{
  "response_biz_content": {
    "return_code": 10308250,
    "return_msg": "上送参数为空"
  }
}
```

## 11.返回码解释

返回码	返回说明
-1	请求异常，请联系工行管理员
-31	响应请求转换失败
-30	服务方响应状态错误
400011	参数非法，，原因可能为app id为空、app id非法、签名为空、应用系统时间与API平台系统时间不在限定差值以内、时间戳非法
500018	访问的API不存在
500020	非法调用
400016	app公钥未维护
400017	签名验证失败
400019	授权验证失败
500031	速率超限
500032	并发超限
-500041	代理异常
-500042	代理超时
-500044	网关签名失败
500043	网关配置文件错误，无法从配置文件中读取配置
500091	没有管理权限
-500099	网关内部异常
0	调用成功
10308512	上送参数为空
10308512	当前订单不支持退款
10308516	无法再次申请退款
10308516	本次退款申请金额不合法