# Antflow.net快速上手手册

本手册为快速上手手册,主要介绍如何快速将antflow使用起来,以及如何快速快速扩展常见功能，是一个精简的操作手册。

## 常见基本概念和术语

### antflow流程分类

#### 从接入形式上，antflow的流程分为内嵌应用接入和外部SAAS接入。

##### 内嵌接入。即为将antflow做为一个模块嵌入到已有的应用中，通过进程内方式调用antflow工作流引擎的方法实现流程流转。

##### 外部SAAS接入。即将antflow做为一个独立的应用部署，其它需要使用工作流的项目通过http请求antflow接口，实现流程的流转。

补充说明：用户可以根据自己公司项目现状和特点综合评估，自行决定是使用嵌入式还是SAAS式。如果公司项目是微服务形式，SAAS形式接入可能是一种比较理想的选择。如果公司项目是单体应用，嵌入应用内可能更适合一些，嵌入应用内架构相对简单，可以使用数据库事务，缺点就是需要集成到已有应用中，需要花些时间来做集成。

关于集成，其实笔者并没有做过将antflow集成到其它项目里，也没有集成经验。这里提供一些基本的建议。首先是看自己项目的.net版本是多少，然后修改antflowcore.csproj里面的TargetFramework（需要注意的是antflow使用的是.net 9.0，但是并没有使用较新的语法，理论上也是能兼容或者仅需要做少数修改即可兼容较旧一些的.net。此外项目的依赖及版本都定义在了Directory.Packages.props里面。如果和自己项目里版本冲突，可以做相应调整。最后将antflowcore整个模块加载到自己的应用里。Antflownet是一个空的应用出口项目，用户可以将Program.cs里的配置酌情复制到自己的项目中。（有些是运行antflow必须的）

* + 1. 以上是通过接入形式区分的，从流程特点上则可以分为低代码流程LF(lowcodeflow)流程和DIY流程（自定义流程）
       1. LF流程，LF流程为低代码表单流程，用户在项目中看到LF开头的内容大都是和低代码流程有关的。低代码表单流程适用于表单业务不是特别复杂（当然复杂的表单也能做得了，antflow低代码表单集成的是vform，能做到多复杂得看用户自己对vform的掌握程度。一般而言，没有特殊的定制，还是很容易上手的。）。  
          1.1.2.2 DIY流程，DIY流程即完全自定义流程。包括前端表单以及后端逻辑控制。

需要说明的是，lf流程和diy流程的主要核心差别在于表单处理上。即LF流程不需要前端写表单，只需要后端写控制逻辑即可。很多用户误以为LF流程只适合后端逻辑不是很复杂的流程，其实这种观念是错误的。Diy流程和lf流程是后端控制能力上无本质差别。Lf本身是靠diy流程来实现的。

### 流程节点类型

目前antflow有六种节点类型，做为快速上手，这里不一一介绍。主要介绍审批人节点、抄送节点、条件节点和并行审批节点。

#### 审批人节点

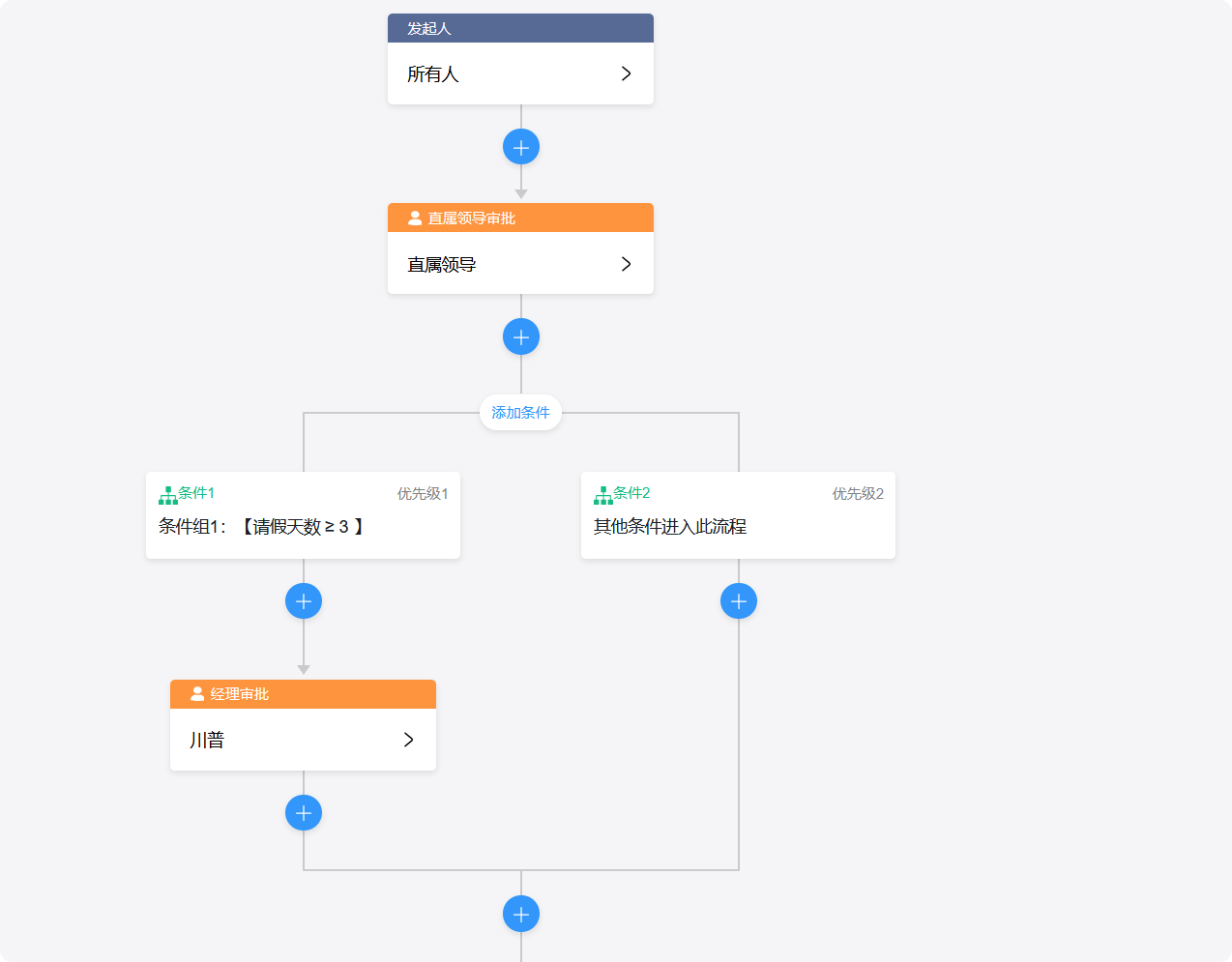
审批人节点如其名，即可以设置审批人的节点。Antflow提供的丰富的审批人选择规则。例如指定人员，指定角色，直属领导，部门领导等。用户还可以结合自身的业务系统对其进行扩展。

#### 抄送节点

抄送节点也像它的名字一样，即设置流程抄送人的节点。有时候流程并不需要指定的人来审批，但是需要知会他，这时候抄送人节点就派上用场了。抄送人节点数据通过任务中心-抄送我的来查看。

#### 条件节点

有些流程需要根据条件来决定选择不同的审批人节点，这时候条件节点就派上用场了。比如一个请假流程如果请假天数大于3天就需要加上经理审批节点。可以使用增加条件节点来实现。Demo如下：



#### 并行审批

并行审批一般用于传统行业需要多人/多部门同时审批的场景。并行审批相较以上节点，并不是必须的，使用场景也不是特别多。如果不是特别复杂可以通过节点会签来达到类似的功能。并行分支节点过多分导致APP上呈现效果较差，建议使用的时候慎重考虑。

### 流程审批类型

有些审批人节点包括一个以上的审批人， antflow支持会签、顺序会签和或签三种审批类型。

#### 会签

会签需要当前节点所有人都审批通过了流程才会继续向下走，只要其中任意一个人拒绝流程即刻结束。

#### 顺序会签

和会签不同的是，会签是所有人都可以看到审批任务。顺序会签会按照用户指定的审批顺序依次签署。当前审批人同意以后下一个审批人才能看到审批待办任务。

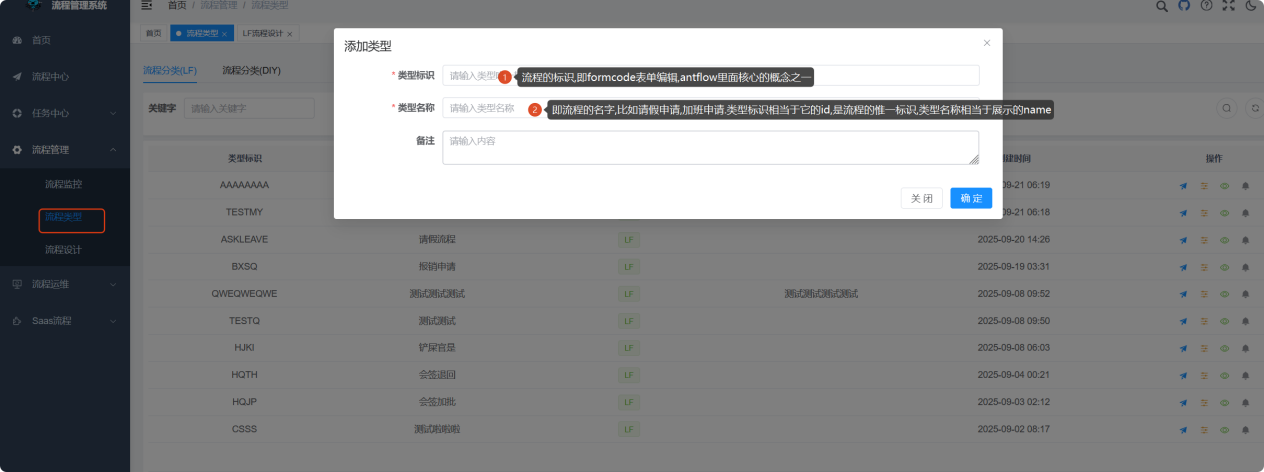
需要说明的是，目前仅指定人员支持顺序会签。签署顺序和用户选择人的顺序保持一致。其它像角色，部门负责人这样根据规则找出来的审批人不知道顺序是什么，因此不支持顺序会签。

#### 或签

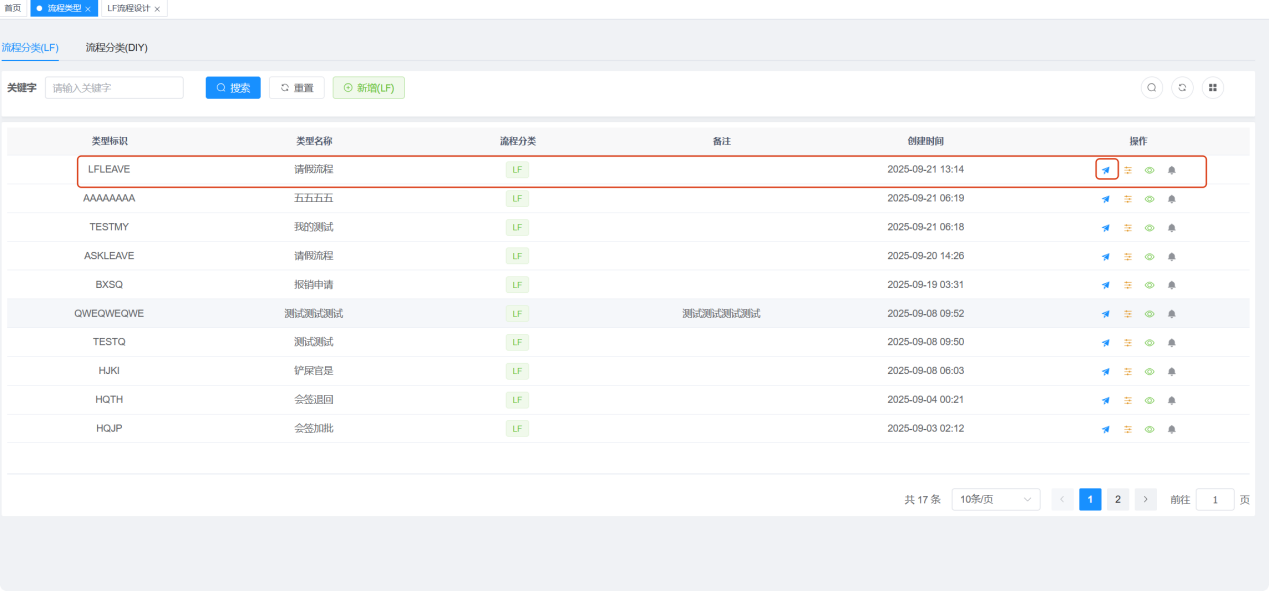
会签和顺序会签需要所有审批人都审核同意以后流程才继续向下流转。或签则只需要当前节点任意一审批人同意流程即继续向下走。

## 快速开始一个LF(低代码流程）

要发起一个流程，首先要定义这个流程。即流程模板（也可以叫作流程图）。进入系统，打开流程管理-流程分类，里面有两个tab，默认的是低代码流程。点击【新增LF】按钮，填写完基本信息



以上操作完成以后，下面列表里就会多出一条记录。下面点击记录后面的纸飞机图标，即可设计流程



关于流程设计这里不再赘述，用户可以自行玩一玩，尝试点一点。完了之后点击右上角【发布】按钮即完成了流程设计。发布这后就会跳到列表页面。



这里并没有完成，还需要点击【启动】才可以使当前设计的流程生效。启动以后流程设计就完成了。这时候回到流程中心，就可以看到可以起的流程多了一个你刚刚设计的，需要注意的是。有时候按以上操作以后可发起的流程并没有你刚刚设计的。这时候刷新页面或者退出重新登陆（一般是不需要退出登陆的）

填写表单，即可以发起流程了。然后退出登陆，使用流程审当前节点审批人账号登陆，当前节点审批人便能从【任务中心】-【待办任务】看到需要审批的流程了。

## 快速开始一个DIY流程

Diy流程需要设计配置页面。关于如何将自己的页面和流程绑定。详情查看官网首页右侧【操作手册-自定义表单】

这里着重强调一下后端内容，后端java和.net有不小的区别。

用户创建一个类，任意命名。但是需要实现IFormOperationAdaptor。如果用户不不知道具体方法是做什么的，可以先不管，先把类创建出来，其中IFormOperationAdaptor为泛型接口，泛型参数为BusinessDataVo，或者其子类。这个类包含了流程的表单信息和一些其它的用户提交的信息和一些流程的运行时信息。一般情况下用户只需要关心定义表单字段，用于接入前端传过来的表单数据。

实现类除了集成以上接口，还需要加上DIYFormServiceAnno属性，其中SvcName即为formcode，代表这个业务的惟一标识，desc为流程的描述，实际上是流程的名字。和上面低代码表单流程创建是对应着的。完了以后，要将以上自定义类加到IOC容器中，进入到ServiceRegistration类仿照已有三方账号申请demo进行注册

services.AddSingleton<IFormOperationAdaptor<ThirdPartyAccountApplyVo>, ThirdPartyAccountApplyService>();

执行完以上操作后，重新启动应用，然后再打开前端页面，就可以发起一个DIY流程了。

关于IFormOperationAdaptor 接口里的方法都是干什么的，可以查看每个方法上的注释。为了方便可以快速上手，用户可以模仿三方账号申请demo流程ThirdPartyAccountApplyService，看里面的方法，刚开始可以将内容原样复制过去，把表名替换掉，字段保持一致。然后不断调整内容看看有什么变化。理解以后后面就可以写出更复杂的逻辑控制了。