# 用户协议流程分析

[用户协议流程分析 1](#_Toc526947865)

[初始化App#onCreate 3](#_Toc526947866)

[1.初始化initHVISDK() 3](#_Toc526947867)

[2.跟新配置项(多语言情况下) 3](#_Toc526947868)

[3.添加登录监听，添加检查权限监听 3](#_Toc526947869)

[4. 设置默认的协议配置ServiceTermModel.getInstance().init(); 3](#_Toc526947870)

[5. 初始化UDID 4](#_Toc526947871)

[6. 预置之前的协议版本号 4](#_Toc526947872)

[启动页流程 4](#_Toc526947873)

[国内版本 4](#_Toc526947874)

[1.Intent检测 4](#_Toc526947875)

[2.检查协议 5](#_Toc526947876)

[3.上层checkSignStatueTask启动，上层执行start 5](#_Toc526947877)

[4. HVI LocalTermsManager检查本地数据库 5](#_Toc526947878)

[5.查询返回： 5](#_Toc526947879)

[6.dealwithCheckSignStatus处理是否需要dialog 6](#_Toc526947880)

[国外版本 6](#_Toc526947881)

[Intent检测 6](#_Toc526947882)

[添加登录监听 6](#_Toc526947883)

[执行登录 6](#_Toc526947884)

[服务器请求状态 7](#_Toc526947885)

[成功： 7](#_Toc526947886)

[失败： 7](#_Toc526947887)

[处理服务器回调，检查签署 7](#_Toc526947888)

[处理签署状态 8](#_Toc526947889)

[提示弹框 8](#_Toc526947890)

[同意签署： 9](#_Toc526947891)

[隐私变更流程 9](#_Toc526947892)

[游客： 9](#_Toc526947893)

[用户： 10](#_Toc526947894)

[关闭服务流程 11](#_Toc526947895)

[海外版本本地视频打开在线服务 11](#_Toc526947896)

[常见问题列表 12](#_Toc526947897)

# 初始化App#onCreate

## 1.初始化initHVISDK()

1.1能力初始化：Context，网络，加密

1.2数据库初始化

1.3接口初始化（设置预置接口地址），设置国内/海外接口（HVIRequestConfig）

1.4逻辑层初始化：注册各个接口实现类（XXXXLogic.java 单例）

## 2.跟新配置项(多语言情况下)

2.1获取保存在Sp本地的旧的协议配置项，key= CUSTOM\_CONFIG

2.2LoginConfig#getCustomConfig同步保存旧的协议配置项

## 3.添加登录监听，添加检查权限监听

## 4. 设置默认的协议配置ServiceTermModel.getInstance().init();

4.1设置默认协议配置

4.2设置预置的最新协议配置

海外：

默认：\"argType\":\"124\",\"must\":\"1\",\"baseVersion\":\"20171225\"

\"argType\":\"10015\",\"must\":\"1\",\"baseVersion\":\"20171225\"

最新：无

国内：

默认：\"argType\":\"114\",\"must\":\"1\",\"baseVersion\":\"20180227\

\"argType\":\"10007\",\"must\":\"1\",\"baseVersion\":\"20180227\

最新：\"argType\":\"114\",\"latestVersion\":\"20180915\",\"needSign\":true

\"argType\":\"10007\",\"latestVersion\":\"20180915\",\"needSign\":true

## 5. 初始化UDID

查找Sp中ParameterTable.UDID字段，如果没有，那么从系统中读取UDID

## 6. 预置之前的协议版本号

ServiceTermUtils.initTermsOfServiceVersion();

5.1获取Sp中ParameterTable.IS\_AGREE是否同意签署

5.2获取Sp中ParameterTable.TERMS\_OF\_SERVICE\_VERSION用户服务条款版本号

比较四种版本

1：v6.0.0.311 之前

2：v6.0.0.311 之后

3：v6.0.0.313 之后

4：v8.0.6.1 之后

5.3判断方式如下：

5.3.1

默认当前为0

本地协议大于2(6.0.0.311)

默认当前协议不变，311之前如果同意，统一预置为1（311之前）

否则判断是否存在升级/覆盖安装查找对应文件存在是否，如果存在设置默认协议1

5.3.2

新更改的版本号保存到Sp中

# 启动页流程

## 国内版本

## 1.Intent检测

来源：开放能力（直接查询签署，不会显示UI页面），Video，正常启动，判断是否进入本地视频

1.1是非华为设备，EMUI3.0一下也只显示本地视频

1.2是移动定制机ROM版本显示本地视频

## 2.检查协议

调用showPromptDialogIfNeed

## 3.上层checkSignStatueTask启动，上层执行start

3.1判断是否第一次使用/同意/拒绝过协议，初始化版本号是否在3.0.0.311之后，值>=1，判断App初始化设置的当前版本号是否>=1（这是在App#onCreate下面跟新lastVersion来判断是否存在值），是否之前使用过，如果没有使用过，调用dealWithNoSignRecord

3.2否则执行HVI TermsManager#checkSignStatue查询

## 4. HVI LocalTermsManager检查本地数据库

4.1getAllRecord查询所有最新签署记录（不区分用户和协议类型），将获取到的LocalSignRecord类型列表转化为上层可以使用的SignRecord类型列表，由此可以看出保存在DB中的字段有以下：

用户Id,版本号，协议类型，签署时间，是否同意

## 5.查询返回：

如果返回查询为null集合，返回无记录NO\_SIGN，否则拿签署记录集合之后按照签署时间进行降序排序，获取第一条的签署记录，并检查，返回无需签署，已经签署

5.接收返回提示

5.1 CheckSignStatusTask回调

5.2检查本地签署无记录，使用dealWithNoSignRecord，查找RECORD\_AGREE是否为之前同意过，并且版本号>=1，返回已同意，如果是版本4（8.0.6.1之后）使用agreeTerms（5.2.1）保存在本地（传入List签署记录是空的），否则表示首次无缓存，返回需要签署

5.2.1 首先创建保存在本地的签署记录AgreeTermsTask#initSignRecord

TermsConfig#getTermLatestConfig获取最新协议配置项（返回为保存本地跳过，本地不保存任何信息）

如果没有相应的协议信息，则根据视频云配置的信息则生成

如果没有视频云配置项，则根据默认的协议信息生成

## 6.dealwithCheckSignStatus处理是否需要dialog

6.1如果已经同意签署，设置ParameterTable.RECORD\_AGREE中Sp字段为true

6.2否则显示弹框（国内版本提示弹框），如果同意，ParameterTable.RECORD\_AGREE设置为true，并且将ParameterTable.TERMS\_OF\_SERVICE\_VERSION默认协议跳转设置为4版本（8.0.6.1之后）

6.2.1HVI AgreeTermsTask保存签署记录到本地数据库（查找Sp里面基础协议和最新协议，如果没有使用默认协议），返回成功，没有失败状态

6.2.2 ConfigKey.USER\_ID获取用户ID，判断如果不是游客，需要上传服务器（将SignRecord转化为SignInfo）（查找Sp里面基础协议和最新协议，如果没有使用默认协议），上传服务器不管成功失败，回调无操作

接收dealwithAddSignRecord

6.3走requestPermissionIfNeed申请权限

## 国外版本

## Intent检测

来源：开放能力，Video，正常启动

2.调用showPromptDialogIfNeed检查协议

## 添加登录监听

上层添加检查签署状态监听（执行登录BeInfoLoader触发），initAdvertViews添加GRS监听

4.执行doNext登录请求

5.HVI执行CBGLoginTask/GuestLoginTask

5.1LoginManager执行登录，AccountLogic.hasAccountLogin判断是否是有登陆过

## 执行登录

如果是游客，执行游客登录loginGuest

如果是用户，执行CBG登录

5.2执行GetBeInfoTask获取BeInfo配置

6.HVI执行GRSLoaderTask/BeinfoLoader，执行getBeInfo完成之后如果失败，将设置为LoaclVideo,否则BaseBeInfoLoader#saveBeInfo将请求结果GetBeInfoResp使用LoginConfig保存到Sp中

6.0

服务器请求失败或者GetBeInfoResp#getBeinfo为空：

判断是否无服务，LoginConfig从Sp里面获取使用缓存信息

无对应服务和缓存

失败情况下优先使用缓存信息，有缓存也表示请求成功

## 服务器请求状态

### 成功：

将返回信息GetBeInfoResp使用LoginConfig加密存在Sp里面

BaseBeInfoLoader#saveBeInfo保存运营渠道信息到本地

判断是否开通在线功能，如果没有，返回失败，进入本地视频，否则执行检查签署状态6.1

### 失败：

回调失败设置非本地视频，成功设置为本地视频LoginConfig.getInstance().setLocalVideo(true);

发送成功/失败的EventBus给上层GRSloaderListener接收

上层GRSLoader只处理GetBeInfo成功（包括判断版本号、协议）的结果，成功后表示可以进入在线

跳转主页面，设置Record\_Agree为true，保存设置版本号为4

## 处理服务器回调，检查签署

6.1添加签署状态监听，执行检查签署状态CheckSignStatusTask，外海版本检查签署状态

6.1.1 TermsConfig.getInstance().getTermsConfig()获取需要检查的基础协议配置项（查询不到则返回默认），如果getTermsConfig默认为空，直接返回成功，参数为null，使用dealWithNoLocalRecords，6.1.3传入空列表

6.1.2 LocalTermsManager检查本地签署记录（UID查询）

0.如果本地有记录，使用6.1.3判断条件，传入数据是本地数据而不是服务器数据

1.如果本地没有记录，使用dealWithNoLocalRecords，6.1.3传入空列表

判断如果是游客，返回无记录或者判断是否需要跳过签署，如果是用户，那么ServerTermsManager检查TermsConfig 协议的TMS签署记录（先获取AccessToken）

服务器出错则返回无签署记录NO\_SIGN

如果返回空记录，则需要签署

上层监听回调处理Need\_Sign和No\_Sign两种协议状态

下层登录处理拒绝和同意两种协议状态

如果有记录则将TMS返回记录保存在本地，使用6.1.3判断，传入数据是服务器数据

6.1.3 返回CheckSignStatusTask#hasAgreedTerms判断是否需要签署还是跳过

1. 空数据表示需要签署/跳过签署

2. 获取视频云协议配置项（LoginConfig#CustomConfig#configStatmentArgType），3. 将服务器的数据转为List<SignRecord>

4. 如果2为空，从3中获取默认预置签署记录第一条类型的签署记录，与其比较默认签署记录版本号，判断是否高，返回是否签署

5. 遍历2每种协议类型，查询2中必须签署协议类型，得到服务器返回的对应协议类型每条SignRecord判断是否需要签署（如果不存在表示需要签署，表示服务器不存在该本地基础协议），以及与CustomConfig#configStatmentArgType每条协议基础初始版本号比较返回，小于则表示需要签署

## 处理签署状态

7. 如果返回NO\_SIGN即服务器出错情况下，判断ParameterTable.RECORD\_AGREE中值，是否之前签署过，返回同意或者需要签署（去除此类情况，因为不是使用的上层CheckSignStatueTask，上层Launcher才使用该判断）

8.回调上层的签署记录，VideoFolderActivity.KEY\_JUST\_TRY\_GRS用于区分是否需要跳过下面操作，因为登录的回调一直存在于单例中，可能与VideoFolderActivity中打开在线视频有冲突，接收VideoFolderActivity中登录的通知

8.1处理Need\_Sign与No\_Sign状态，CheckSignStatusTask在收到No\_Sign状态，判断SIGN\_AGREE sp是否为true，如果true那么走已经同意，否则需要签署Need\_Sign

8.1.1 如果页面被销毁了，重启页面

8.1.2 创建提示弹框，用户签署

如果同意签署，（如果来自开放能力，重置登录，回到开放能力）正常启动，将同意签署返回给HVI，用于HVI接收发送通知

如果不同意签署，（如果来自开放能力，关闭页面）正常启动，将拒绝签署返回给HVI，用于HVI接收发送通知

8.2其余8.1状态比如SKIP\_SIGN检查用户渠道是否发生变化，来判断是否需要重新开屏

## 提示弹框

9.签署弹框提示，点击确定/取消（除了开放能力之外，开放能力将检查权限，或者是否执行doNext需要回到开放能力,并重置登录状态）都会通知HVI层的checkSignStatueListener，(即BeInfoLoader#ICheckSignStatusCallback)。

9.1 notifySignResult如果同意那么执行agreeTerms方法（具体实现见下），将签署记录保存在本地

下发通知给登录模块BeinfoLoader里面添加的检查签署状态监听

9.2 ICheckSignStatusCallback接收签署状态回调

9.2.2 如果拒绝签署SKIP\_SIGN||SIGNED\_REJECT，那么设置LoginConfig.getInstance().setLocalVideo(true)，标志为本地视频，否则设置为非本地视频，发送GetBeInfoFinish的EventBus消息

10. 上层GRSLoader上层收到该消息，

10.1只有同意才表示getBeInfo成功，ParameterTable.RECORD\_AGREE设置为true并且跟新版本为4，功跳转主页面JumpToMain，否则不做任何操作，等待登录完成判断是否进入本地视频（VideoFolderActivity）

11.登录完成GRSLoader接收EventBus消息，并且判断是否进入本地视频

12.此时国外版本流程结束，没有请求权限

同意签署流程详细分析：（TermsManager#agreeTerms）

# 同意签署：

LocalTermsManager添加到本地数据库，完成之后判断是否是用户，则需要上传服务器，忽略服务器返回状态

上传服务器准备工作见下面问题6

1. 遍历传入的签署记录，创建并生成新的签署记录包装类，设置签署时间，设置同意签署

2. 如果传入的签署记录是空的，默认优先使用基础协议类型和最新协议，如果最新协议全部包含基础协议才使用最新协议，如果都不存在，使用默认协议配置

3. 创建2中协议包装类，用于上传

4. TMS上传服务器需要获取AccessToken，如果没有则先获取，获取成功执行上传操作，否则返回失败，无其他操作响应

# 隐私变更流程

检查用户协议CheckTermsStatusTask（包含提示弹框）

HVI层检查用户协议跟新

1. TermsConfig#getTermsConfig获取签署协议类型，如果是空返回默认基础协议类型，如果是空，那么跳过协议检查，主页面不做任何提示

2.检查本地1中对应的协议类型并获取DB中特定协议记录

3.判断是游客还是用户

LocalTermsManager检查用户签署状态信息DB文件，海外版本使用UID查询，国内版本查询全部，然后判断是游客还是用户

注意：同意或者不同意不会设置RECORD\_AGREE与版本号4，该部分只会出现在检查签署状态同意里面

## 游客：

1.将本地签署记录和LoginConfig中数据进行比较，

2.获取CustomConfig中最新协议配置，或者基础协议配置，如果基础协力里面不含最新协议，返回基础协议配置项，并且设置need\_sign = ture，这里获取优先返回基础协议，只有最新协议全部包含基础协议情况下才返回最新协议

CustomConfig().getConfUrlStatementArgType()：基础协议类型

CustomConfig().getConfUrlStatementArgVer()：最新协议

3.比较方法：GuestCheckTermsStatusTask#compareSignRecords：

3.1 检索本地SignRecord和2返回的SignRecord，取2里面对应协议类型的协议，比较版本号，如果2中比较新（获取取本地不存在的新协议）加入noticeList提示列表，本地如果存在新的或者2中不存在的协议那么跳过该协议的比较，不加入提示列表

4.返回记录查询NoticeList中是否存在需要签署来判断显示的提示框类型

5.如果同意执行agreeTerms，不同意关闭服务并退出应用（回调MainActivity#exit）

## 用户：

ServerTermsManager获取服务器2中对应协议的签署记录

1. 如果获取失败，使用CheckLocalRecordTask比较

1.1.如果签署协议列表是空，返回showNotice null不做任何事（几乎不出现，因为会使用默认的）

1.2.如果本地签署记录是空，TermsConfig查找Sp里面每个协议对应记录，默认为0，获取最新版本号和基础版本，并加入noticeList列表，返回确认类型B弹框

1.3.如果本地签署记录不为空，读取并设置LocalSignRecord的TermsConfig查找Sp里面每个协议的最新版本号，默认为0，判断是否同意，不同意则加入提示列表，不是最新版本（本地版本号小于Sp查询的版本号）也加入提示列表，存在不同意优先返回提示B弹框，其次提示A弹框，其次通知类弹框

2. 如果获取成功，使用CompareSignRecordsTask比较服务器记录和本地

1.如果协议列表为null，那么将本地上传服务器，服务器记录写入到本地，返回null不做提示（几乎不出现）

2.遍历协议列表，得到本地记录与服务器记录，读取Sp里面对应最新版本号并设置，执行compareSignRecord比较

3.本地签署记录大于服务器签署记录（基础版本号或者签署时间比较新）

本地签署同意，本地是最新的，需要上传服务器，不会加入提示列表，否则如果不同意或者不是最新版本号（与2中最新协议版本比较）则会加入提示

4.否则服务器记录新，加入提示列表

如果服务器签署同意，并且服务器版本大于等于2中查找的最新协议版本，那么插入记录到本地DB，否则如果不同意或者不是最新版本号（与2中最新协议版本比较）则会加入提示

返回NoticeList列表和最新记录列表已经需要显示的弹框类型

5.如果本地或者服务器不存在某个协议记录，当作最老版本协议处理，时间和签署时间都是默认0，这样就可以将不存在的同步最新到本地或者服务器

返回提示弹框根据List判断存在不同意优先返回提示B弹框，其次提示A弹框，其次通知类弹框

上层得到最新记录和noticeList，用来提示用户是否同意或则是通知弹框

如果不同意，MainActivity关闭推送服务并且退出

如果同意（不管是通知弹框还是确认弹框），HVI执行AgreeTask写入本地，如果是用户还会上传Server

同意签署流程详细分析：（TermsManager#agreeTerms）

同意签署：

1. 遍历传入的签署记录，创建并生成新的签署记录包装类，设置签署时间，设置同意签署

2. 如果传入的签署记录是空的，默认优先使用基础协议类型和最新协议，如果最新协议全部包含基础协议才使用最新协议，如果都不存在，使用默认协议配置

3. 创建2中协议包装类，用于上传

LocalTermsManager添加到本地数据库，完成之后上传服务器，忽略服务器返回状态

上传服务器准备工作见下面问题6

# 关闭服务流程

1.判断是否允许关闭服务

1.1请求失败提示重试

2.如果不允许弹框提示

3.如果请求成功允许关闭，那么需要验证用户身份

3.1如果是游客，清楚本地缓存和DB

3.2 如果是用户，验证Account密码或者指纹，验证失败不做任何事，验证成功执行取消签署，上传服务器本地签署记录

3.2.1：创建保存在本地的签署记录，获取最新协议配置项（优先选择基础版本协议，如果都不存在，选择默认协议配置）

（如果设置了相应的协议信息，则生成对应的协议签署记录

如果没有相应的协议信息，则根据视频云配置的信息则生成

如果没有视频云配置项，则根据默认的协议信息生成）

如果成功则使用LocalTermsManager删除本地DB和LocalTermsManager中缓存记录，海外版本删除对应UID记录（不是全部删除）

失败则提示重试

# 海外版本本地视频打开在线服务

VideoFolderActivity#OverSeaOnlineController

添加GRSLoaderListener，添加签署状态监听，启动自动登录流程（见启动页流程），返回是否弹框，确认签署状态，确认或者取消设置Record\_Agree值，反馈HVI签署状态，登录接收处理同意或者拒绝状态发送EVENTBUS给上层GRSLoader的GRSLoaderListsner接收提示升级/进入主页面

LoginConfig:

基类使用Map或者Sp存储，将敏感信息存储在Map内存里面，将非敏感信息存在Sp里面

父类存储的时候会使用加密解密算法

主要信息：

包含Beinfo视频用户信息

GRSLoader：GRS调度信息

CustomConfig：自定义用户信息

# 常见问题列表

1.上报TMS之前所需要处理哪些？

1.将SignRecord转为SignInfo，多处国家和语言

2.TMS执行HuaweiClient获取AccessToken,返回成功执行doRequest，否则回调onFailed

3.设置accessToken和SignInfo，执行Http上传，返回onComplete或者异常onFailed

4.上传服务器不管返回状态

2.保存本地签署记录，协议版本号的获取？

首先会创建保存在本地的签署记录，协议版本号存在与CustomConfig#getUriStatementArgVer最新协议列表和CustomConfig#getUriStatementArgType协议列表里面，如果都不存在那么取默认预置的协议（初始化在App#onCreate里面预置）

3.协议有变更，游客的处理逻辑是？

TermsConfig.getInstance().getTermsConfig()获取协议类型

获取最新协议配置项，如果最新argVer中包含所有argType协议，那么取argVer最新协议配置项，否则取argtype协议配置项，没有返回default预置协议配置项

获取本地协议SignRecord

GuestCheckTermsStatusTask比较签署状态

取协议类型对应的协议配置项，如果本地存在协议但是最新协议配置项不存在则跳过，如果本地不存在那么加入提示列表，两者都存在就比较版本号大小，最新的大则加入提示列表

4.对接协议服务器的地址，如何获取

HVIRequestConfig配置服务器地址

TermsDefaultConstants默认服务器地址

GetBeinfo完成使用CustomConfig写入Sp中的时候会跟新协议地址

隐私变更弹框点击跳转地址优先获取getBeinfo通过CustomConfig写入Sp里面的地址，如果获取不到使用DefaultTermsConst获取协议地址，协议分4种见下面

5.服务器签署拒绝状态

上传签署记录不管是否成功，不做任何操作

协议变更上传服务器也不做任何操作

6.日志TAG

Constants.LOG\_TAG = "TERM\_TAG\_";

7.当前协议相关配置项有哪些？

getBeInfo解析请求结果GetBeInfoResp并将结果存放在Sp/内存中，敏感的有Map过滤存在内存Map中，其他存放在Sp中

CustomConfig#getConfUrlStatementArgType

得到List<AgrTypeConfig>列表。

包含参数：协议类型argType，是否需要强制签署must "0/1"，初始化版本号baseVersion

CustomConfig#getConfUrlStatementArgVer

得到List<AgrVerConfig>

包含参数：协议类型argType，是否需要强制签署needSign true/false，最新版本号latestVersion

8.协议类型

服务条款版本号格式

114 华为视频用户协议默认版本号

115 华为视频会员协议默认版本号

116 华为视频连续包月默认版本号

10007 华为视频隐私申明默认版本号

用户服务条款

6.0.0.311

6.0.0.313

8.0.6.1

协议版本号

baseVersion:20180617

9.服务器地址默认配置与更改：

App#oncreate

HVIConfig.config(new HVIRequestConfig());配置服务器默认地址，HVIRequestConfig分海外和国内

显示弹框可点击文本获取地址定义默认TermsDefaultConstants地址，优先取getBeinfo返回写入的地址

BaseBeInfoLoader#saveBeInfo保存运营渠道信息到本地

11.传参为null，getBeinfo未配置时，如何查询签署记录

服务器请求失败或者GetBeInfoResp#getBeinfo为空：

判断是否无服务，LoginConfig从Sp里面获取使用缓存信息

无对应服务和缓存

失败情况下优先使用缓存信息

回调失败设置非本地视频，成功设置为本地视频LoginConfig.getInstance().setLocalVideo(true);

发送成功/失败的EventBus给上层GRSloaderListener接收

上层GRSLoader只处理GetBeInfo成功（包括判断版本号、协议）的结果，成功后表示可以进入在线

否则在登录成功GRSLoader处理判断是否进入本地视频

12.A+B两种类型，A变更 B未变更，会如何

隐私变更流程，取最新上传服务器或者写入本地

将变更的协议类型加入提示列表

游客：

1.将本地签署记录和LoginConfig中数据进行比较，

2.获取CustomConfig中最新协议配置，或者基础协议配置，如果基础协力里面不含最新协议，返回基础协议配置项，并且设置need\_sign = ture，这里获取优先返回基础协议，只有最新协议全部包含基础协议情况下才返回最新协议

CustomConfig().getConfUrlStatementArgType()：基础协议类型

CustomConfig().getConfUrlStatementArgVer()：最新协议

3.比较方法：GuestCheckTermsStatusTask#compareSignRecords：

3.1 检索本地SignRecord和2返回的SignRecord，取2里面对应协议类型的协议，比较版本号，如果2中比较新（获取取本地不存在的新协议）加入noticeList提示列表，本地如果存在新的或者2中不存在的协议那么跳过该协议的比较，不加入提示列表

4.返回记录查询NoticeList中是否存在需要签署来判断显示的提示框类型

用户：

ServerTermsManager获取服务器2中对应协议的签署记录

1. 如果获取失败，使用CheckLocalRecordTask比较

1.1.如果签署协议列表是空，返回showNotice null不做任何事（几乎不出现，因为会使用默认的）

1.2.如果本地签署记录是空，TermsConfig查找Sp里面每个协议对应记录，默认为0，获取最新版本号和基础版本，并加入noticeList列表，返回确认类型B弹框

1.3.如果本地签署记录不为空，读取并设置LocalSignRecord的TermsConfig查找Sp里面每个协议的最新版本号，默认为0，判断是否同意，不同意则加入提示列表，不是最新版本（本地版本号小于Sp查询的版本号）也加入提示列表，存在不同意优先返回提示B弹框，其次提示A弹框，其次通知类弹框

2. 如果获取成功，使用CompareSignRecordsTask比较服务器记录和本地

1.如果协议列表为null，那么将本地上传服务器，服务器记录写入到本地，返回null不做提示（几乎不出现）

2.遍历协议列表，得到本地记录与服务器记录，读取Sp里面对应最新版本号并设置，执行compareSignRecord比较

3.本地签署记录大于服务器签署记录（基础版本号或者签署时间比较新）

本地签署同意，本地是最新的，需要上传服务器，不会加入提示列表，否则如果不同意或者不是最新版本号（与2中最新协议版本比较）则会加入提示

4.否则服务器记录新，加入提示列表

如果服务器签署同意，并且服务器版本大于等于2中查找的最新协议版本，那么插入记录到本地DB，否则如果不同意或者不是最新版本号（与2中最新协议版本比较）则会加入提示

返回NoticeList列表和最新记录列表已经需要显示的弹框类型

5.如果本地或者服务器不存在某个协议记录，当作最老版本协议处理，时间和签署时间都是默认0，这样就可以将不存在的同步最新到本地或者服务器

返回提示弹框根据List判断存在不同意优先返回提示B弹框，其次提示A弹框，其次通知类弹框

13.协议变更弹框如何创建，如何获取协议变更内容

创建协议变更弹框如下：

TermUtils#mergeDialogType ：用户获取签署弹框类型

优先返回不同意确认类型B（DialogType.confirmB）弹框，如果新的协议或者需要签署，返回（DialogType.confirm）确认型弹框，否则返回通知类弹框（已同意）

TermUtils#getDialogTypeForGuest ：游客获取签署弹框类型

优先返回需要签署DialogType.confirm确认型弹框A，否则返回DialogType.notice通知类弹框（已同意）

获取协议变更内容如下：

确认类型B：PrivacyNoticeManager.getInstance().createNoticeDialog，启动PrivacyNoticeActivity页面，传入AGREEMENT\_LIST协议类型，和DIALOG\_TYPE = DialogType.RECONSIDER\_UPDATE\_AGREEMENTS

使用PrivacyNoticeUtils创建高亮可点击文字，根据协议类型，比如（114，115，116，10007，只有这四种情况，没有将不添加高亮），然后用CustomConfig获取Sp存储的对应协议Uri地址，如果不存在那么返回，默认协议地址TermsDefaultConstants中定义，然后设置点击跳转WebActivity页面（子页面ProtocolWebActivity）

确认类型A：PrivacyNoticeManager.getInstance().createNoticeDialog，启动PrivacyNoticeActivity页面，传入AGREEMENT\_LIST协议类型，和DIALOG\_TYPE = DialogType.NEED\_RESIGN

无高亮可点击文字，有确定和取消选项，如果点击取消，那么显示通知类型B弹框

通知类弹框：PrivacyNoticeManager.getInstance().createNoticeDialog，启动PrivacyNoticeActivity页面，传入AGREEMENT\_LIST协议类型，和DIALOG\_TYPE = DialogType.NOT\_NEED\_RESIGN

无高亮可点击文字，只有确定选项，表示同意

注意：3中弹框点击确认都会执行agreeTerms保存本地并选择上传TMS