Account SDK Interface

Prepared by

Ford Motor Company

10-Sept-2021

Version 1.7

Contributor

Zhu Jialin、Zhang Jia

|  |  |  |  |
| --- | --- | --- | --- |
| **REVISION HISTORY** | | | |
| Version | Date | Name | Comment |
| 0.1 | 18-Aug-2020 | Zhu Jialin | Initial version |
| 0.2 | 23-Sep-2020 | Zhu Jialin | Modify interface callback  Add get token interface |
| 0.3 | 24-Sep-2020 | Zhu Jialin | Add “UserProfile”  Modify interface callback |
| 0.4 | 19-Oct-2020 | Zhu Jialin | Add param to logout interface  Add new interface to handle error |
| 0.5 | 26-Oct-2020 | Zhu Jialin | Change class name add openId in UserProfile |
| 0.6 | 10-Nov-2020 | Zhu Jialin | Add param to getUserProfile() Add param to getAccountToken() |
| 0.7 | 11-Nov-2020 | Zhu Jialin | Update BaiduAccount |
| 0.8 | 11-Nov-2020 | Zhu Jialin | Add param to logout interface |
| 0.9 | 11-Nov-2020 | Zhu Jialin | Add new interface to release connection |
| 1.0 | 1-Dec-2020 | Zhu Jialin | Fix jar crash bugs  Add interface for ev sdk connection |
| 1.1 | 5-Dec-2020 | Zhu Jialin | Add param to isLogin interface |
| 1.2 | 10-Dec-2020 | Zhu Jialin | Add interface to go to AccoutList page Add environment switch related interface |
| 1.3 | 11-Jan-2021 | Zhu Jialin | Add interface to optimize ci token polling  Modify faceId interfaces |
| 1.4 | 29-Jan-2021 | Zhu Jialin | Add interface to fix bug |
| 1.5 | 02-Apr-2021 | Zhu Jialin | Add init failure callback  Modify UserProfile to add userImage |
| 1.6 | 25-Aug-2021 | Zhang Jia | 1. Add communication interface with Account HMI apk 2. Add FaceID registration interface   3. Modify the login success interface return type  4. Add an interface to get the information of the currently logged-in user  5. Add interface to get enhancedMemory registration status |
| **1.6.1** | 10-Sept-2021 | Zhang Jia | Add Ford tripartite app to obtain login user information and monitor login and logout messages.  Add user info explain |
| **1.7** | 24-Sept-2021 | Zhang Jia | Add an interface to obtain license plate information for launcher and Personal Center. |

1. Introduction

This document describes the interfaces exposed by FordAccount management for baidu.

1. Interface Matrix

Below table summarize all APIs provided by IVI Service Layer

|  |  |  |
| --- | --- | --- |
| class | interface | description |
| **IAccount**  **Manager** | **void** initConnection(IConnectionListener iConnectionListener) | Init connection for ev sdk or other apps |
| **void** connectFordAccountService(in IFordAccountListener iFordAccountListener) | register account management listener(Support for FaceOS calls) |
| **void** accountConnectAccountService(final IAccountViewListener iAccountViewListener) | register account management listener(Support for Account HMI calls) |
| **void** informLoginSuccess(String openId) | inform ford account that account with openId login success |
| **void** submitLoginInfo(String qrCode) | submit baidu qrcode info to ford |
| **List<BaiduAccount>** getAccountList() | get account list |
| void deleteAccount(String openId) | delete account |
| **void** logout(String openId, boolean isPassive, String reason) | inform ford account that account with openId logout, param isPassive identifies whether account logged out by user or baidu |
| **UserProfile** getUserProfile(String openId) | get user profile with openId |
| **void** startLoginView() | request login ui to start up |
| **boolean** isLogin(String openId) | to judge is ford account with openId login |
| **void** setAutomaticLogin(booleanisAutomatic) | to set automatic login property, for system setting |
| **boolean** isAutomaticLogin() | obtain automatic login property above |
| **void** faceIdLoginResult(String openId, booleanisLogin) | inform ford account that faceId with openId login info |
| **String** getAccountToken(Context context) | get account access token |
| **void** disconnectFordAccountService() | Disconnect if needed |
| **UserProfile** getCurrentUserInfo(boolean fromCache) | Get the information of the currently logged-in user.  fromCache is true, It should be called after receiving the login success broadcast.  fromCache is false, It should be called in other scenarios to get the latest user information |
| **void** accountViewFaceIdRegister(String openId) | When Account HMI apk need FaceId Register |
| **void** onFaceIdRegisterResult(String openId, boolean result) | When BaiDu FaceOS need notice Ford Account FaceId Register result |
| **void** accountViewLoginCancel() | When Account HMI apk need notice Ford AccountService Login cancel |
| **void** accountViewCheckAccountMaxLimit() | When Account HMI apk need notice Ford AccountService check Account max storage quantity limit |
| **boolean** onGetUserEmRegisterState(String openId) | When BaiDu FaceOS need get enhancedMemory registration status |
| **void** onEnvironmentSet(Boolean isSandbox) | To inform ford that baidu environment changes |
| **void** onUserScaned() | When user scaned qrcode, baidu should use this interface to inform ford |
| **Void** onBaiduLoadSuccess() | When login success baidu need time to load state from cloud, after that baidu call this interface to inform ford to exit qr page. |
| **void** accountViewGetBaiduQrCode() | When Account HMI apk need notice Ford AccountService request BaiDu QrCode info |
| **void** accountViewExchangeToken(String accessToken, String userCode, String traceId) | When Account HMI apk need notice Ford AccountService make exchange token operation |
| **void** accountViewFaceIdLogin() | When Account HMI apk need notice Ford AccountService make FaceId login operation |
| **void** accountViewUploadEvent(EventUploadList event) | When Account HMI apk need notice Ford AccountService make upload event log operation |
| **void** onRequestLicensePlateNumber() | When the current launcher needs to request license plate information |
| **void** onSendLicensePlateNumber(String lpnInfo) | When the current personal center needs to return license plate information |
| **boolean** onGetUserEmRegisterState(String openId) | When need EnhancedMemory and account binding relation |
| **void** registerLicensePlateChangeListener(int applicationContextHash, ILicensePlateChangeListener listener) | When the current launcher needs to get license plate information |
| **void** unRegisterLicensePlateChangeListener(int applicationContextHash, ILicensePlateChangeListener listener) | WWhen the launcher does not need license plate information |
| **IFordAccountListener** | **void** onDisconnected(); | when this method called the service is unavailable |
| **void** onGetBaiduQrcode(); | Request baidu qrcode info, baidu should cal  IAccountManager.submitLoginInfo() to pass baidu  Qrcode info to ford |
| **void** onFordAccountLoginSuccess(UserProfile userProfile); | inform baidu that account with openId login success |
| voidonAccountLoginSuccess(); | Baidu account login |
| voidonFordAccountLogoutSuccess(String openId); | inform baidu that account with openId logout success |
| void onAccountLogoutSuccess(); | Baidu account logout |
| void onFordAccountLoginError(String openId, String reason); | inform baidu that ford account login process failed with reason |
| void onAccountLoginError(String reason); | when global account login fails, inform baidu |
| void onFordAccountLogoutError(String reason); | when ford account logout fails, inform baidu |
| void onAccountLogoutError(String reason); | when global account logout fail, inform baidu |
| **boolean** isBaiduInSandbox() | Get baidu environment |
| **Void** onAccountListView(); | Go to user list page |
| **Void** onFaceIdLogin(); | Start faceId login process |
| **void** onCancelLoginView() | When Ford Account camcel login |
| **void** onFaceIdRegister(String openId) | When Ford Account need FaceID register |
| **Void** onRequestLicensePlateNumber() | AccountService requests license plate number info from Baidu. |
| **IConnectionListener** | **void** onConnected(); | while service connected this method will be called |
| **void** onDisconnected(); | when this method called the service is unavailable |
| **IAccountViewListener** | **void** onConnected() | When Ford Account HMI apk connect Ford AccountService success |
| **void** onDisconnected(); | When Ford Account HMI apk disconnect Ford AccountService |
| **void** accountViewGetBaiduQrCodeResult(String baiduQrCode); | When Ford AccountService notice Ford Account get BaiDu QrCode result |
| **void** accountViewLoginResult(boolean loginSuccessful, UserProfile userProfile) | When Ford AccountService notice Ford Account login result |
| **void** accountViewFaceIdRegisterResult(String openId, boolean result) | When Ford AccountService notice Ford Account FaceID register result |
| **void** accountViewCheckAccountMaxLimit() | When Account HMI apk need notice Ford AccountService check whether the maximum number of stored account numbers is exceeded |

**UserProfile：**

|  |  |  |
| --- | --- | --- |
| **Param** | **Type** | **Describe** |
| mUserId; | **String** | 用户唯一识别码 |
| mUserName; | **String** | 用户手机号 |
| mFirstName; | **String** | 用户姓 |
| mLastName; | **String** | 用户名 |
| mTitle; | **String** | 标题 |
| mVehicleNickname; | **String** | 车辆昵称 |
| mUserProfileImage; | **String** | 用户头像（图片字节码，需使用Bitmap解码后使用） |
| mPartnerRelationships; | **List<PartnerRelationshipsBean>** | 登录协作方信息（例：百度账号信息） |
| mOpenId; | **String** | 百度账号唯一识别码 |
| mFirstLogin | **boolean** | 是否首次登录（删除账号登录记录会重置） |

**PartnerRelationshipsBean：**

|  |  |  |
| --- | --- | --- |
| **Param** | **Type** | **Describe** |
| mPartnerName; | **String** | 登录协作方名称（例：baidu） |
| mIdentifierInPartnerSystem; | **String** | 登录协作方唯一识别码（百度账号识别码openId） |

1. Interface description

com.ford.sync.accoutmanager.FordAccountManager.java

// for passive logout  
public static final String *REASON\_BAIDU* = "reason\_baidu";  
public static final String *REASON\_FORD* = "reason\_ford";

// for startLoginUI  
public static final int *FROM\_PASSIVE\_LOGOUT* = 0x11; //in passive logout dialog  
public static final int *FROM\_MANUAL\_LOGIN* = 0x12; //in launcher  
public static final int *FROM\_SWITCH\_ACCOUNT* = 0x13; //in personal center, switch account  
public static final int *FROM\_AUTO\_LOGIN* = 0x14; //when boots up, auto login  
public static final int *FROM\_FACEID\_LOGIN* = 0x15; //faceId login  
public static final int *FROM\_AUTH\_APP* = 0x16; //click other app which need login  
public static final int *FROM\_OTHER* = 0x17; // other cases

* 1. public void initConnection(IConnectionListener iConnectionListener);

Provides interface to connect account service to obtain access token for other apps, like ev sdk etc.

* 1. public void connectFordAccountService(IFordAccountListener iFordAccountListener);(Support for FaceOS calls)

Use this method to create connection to account service, when success “connected()” method in listener will be called.

* 1. public void accountConnectAccountService(final IAccountViewListener iAccountViewListener)(Support for Account HMI calls)

Use this method to create connection to account service, when success “connected()” method in listener will be called.

* 1. public void informLoginSuccess(String openId);

inform ford account that account with openId login success

* 1. public void submitLoginInfo(String qrCode);

submit baidu qrcode info to ford

* 1. public List<BaiduAccount> getAccountList();

BaiduAccount is contained in com.ford.sync.accountmanager, it has mOpenId and mFirstName, mLastName, mUserAvatar, mLastLoginTimestamp.

public class BaiduAccount implements Parcelable {  
  
 String mOpenId;  
 String mFirstName;  
 String mLastName;  
 String mUserAvatar;  
 long mLastLoginTimeStamp;

}

mUserAvatar is empty string currently, in this case default avatar should be used.

In user list, name should be displayed with mFirstName and mLastName combination.

* 1. public void deleteAccout(String openId);

delete account with specified openId

* 1. public void logout(String openId, Boolean isPassive, String reason);

log out the account with specified openId. When logout process succeed, onAccountLogoutSuccess() method in IFordAccountListener will be called. When parameter “isPassive” is true, this means account logged out due to baidu account relationship change or account delete, user logged out manually otherwise. Param reason identifies whether passive logout is trigged by fordpass remove car etc or baidu login state changes.

public static final String *REASON\_BAIDU* = "reason\_baidu";  
public static final String *REASON\_FORD* = "reason\_ford";

FordAccountManager.REASON\_BAIDU and FordAccountManager.REASON\_FORD.

* 1. public UserProfile getUserProfile(String openId, boolean fromCache);

get the profile of account with specified openId. User profile data is like below:

*/\*\*  
 \* userName : 13000000001  
 \* firstName : Test  
 \* lastName : 123  
 \* title :  
 \* vehicleNickname : MyCar  
 \** userProfileImage : userProfileImage

*\* partnerRelationships : [{"partnerName":"BAIDUID","identifierInPartnerSystem":"20062375000"}]  
 \*/*

When “fromCache” is true, the result is only requested from local db , otherwise account will do request to the real time UserProfile from ivisl apk incase of update.

* 1. public void startLoginView(int from);

request login ui to start up. Param from means where starts login page, options can be found in FordAccountManager.

* 1. public boolean isLogin(String openId)

get current account login status in ford site.

* 1. public void setAutomaticLogin(boolean isAutomatic);

to set automatic login property, for system setting

* 1. public boolean isAutomaticLogin();

obtain automatic login property above, when succeed the following two result are possible:

“true” and “false”

* 1. public void faceIdLoginResult(String openId, boolean isLogin);

inform ford account that faceId with openId login info

* 1. public String getAccoutToken(Context applicationContext);

get ford account access token which communicates with IVISL. It maybe a time-consuming interface because refresh token with IVISL must be done when access token expires. ApplicationContext will be used for package verification.

When return null or empty, it means no account login currently.

Account app will do access token validity check and do access token refresh when it expires to make sure access token is available.

* 1. public void disconnectFordAccountService()

Disconnect service if needed

* 1. public UserProfile getCurrentUserInfo(boolean fromCache)

Get the information of the currently logged-in user.

fromCache is true, It should be called after receiving the login success broadcast.

fromCache is false, It should be called in other scenarios to get the latest user information

* 1. public void accountViewFaceIdRegister(String openId)

When Account HMI apk need FaceId Register

* 1. public void onFaceIdRegisterResult(String openId, boolean result)

When BaiDu FaceOS need notice Ford Account FaceId Register result

* 1. public void accountViewLoginCancel()

When Account HMI apk need notice Ford AccountService Login cancel

* 1. public void accountViewCheckAccountMaxLimit()

When Account HMI apk need notice Ford AccountService check Account max storage quantity limit

* 1. public boolean onGetUserEmRegisterState(String openId)

When Account HMI apk need notice Ford AccountService check Account max storage quantity limit

* 1. onEnvironmentSet(Boolean isSandbox);

when baidu changes the environment, should use this interface to inform ford apk to keep same environment.

* 1. Void onUserScaned();

To reduce IBM interface polling times, when user scaned qrcode and clicked login, baidu should use this interface inform ford to start polling ci token.

* 1. Void onBaiduLoadSuccess();

When login success baidu need time to load state from cloud, after that baidu call this interface to inform ford to exit qr page.

* 1. public void accountViewGetBaiduQrCode()

When Account HMI apk need notice Ford AccountService request BaiDu QrCode info

* 1. public void accountViewExchangeToken(String accessToken, String userCode, String traceId)Void onBaiduLoadSuccess();

When Account HMI apk need notice Ford AccountService make exchange token operation

* 1. public void accountViewFaceIdLogin()

When Account HMI apk need notice Ford AccountService make FaceId login operation

* 1. public void accountViewUploadEvent(EventUploadList event)

When Account HMI apk need notice Ford AccountService make upload event log operation

1. Permission

If you want to use getAccountToken interface above and make communications successful, please provide your package name, which will be added to the white list in Account project to make communications successful.

1. Obtain login and logout state

阐述福特系app获取账号登录状态、登出状态方式。

1. 在Android Manifest中声明使用Account登录广播权限。

<uses-permission android:name="com.ford.sync.account.service.permission.BroadcastReceiver" />

1. 注册动态广播接收登录、退出登出广播

注册广播接收action ：com.ford.sync.account.service.login\_state广播

该条广播包含两个参数：

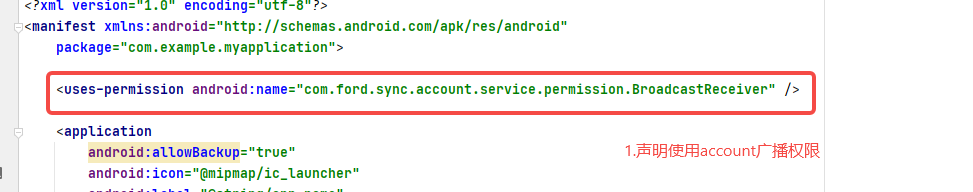
login\_state： int message\_login = 1 || int message\_logout = -1

login\_type：String first\_login = "first\_login" || String again\_login = "again\_login"

1. 收到登录广播时，调用FordAccountManager.getInstance(mContext).getCurrentUserInfo(false)获取当前登录的账号用户信息
2. 当收到登出广播时，设置自己应用登出状态。
3. 若在登录后启动的应用想要获取当前登录的用户信息请使用：FordAccountManager.getInstance(mContext).getCurrentUserInfo(true)

Demo:

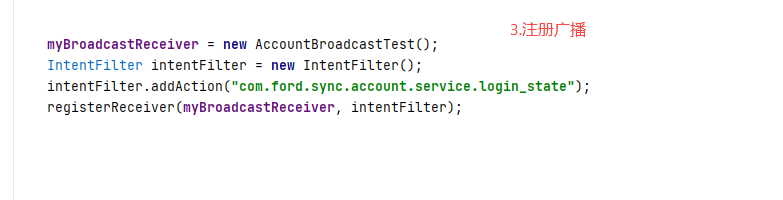
1.声明使用account广播权限。



2.自定义广播，接收account广播。



1. 注册广播。



1. 收到广播后调用获取用户信息方法。

