

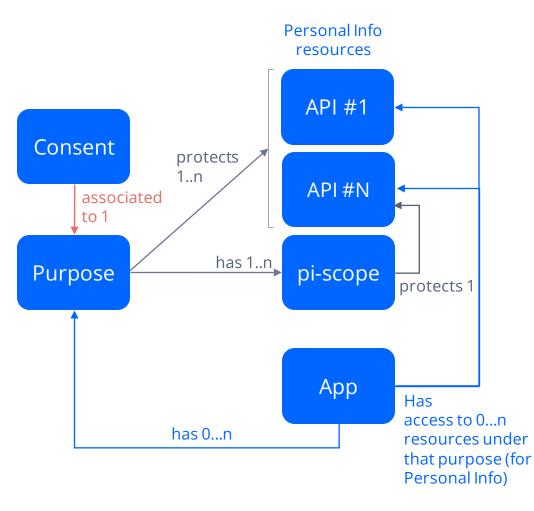
CAMARA APIs User Consent Management

Why User Consent Management?

- User Consent is required by law to process Personal Information.
 - **User** is the human participant which is identified in Telco Operator side by a unique user identifier.
 - Consent is given by user to process Personal Information under a specific purpose
 - **Purpose** declares the reason for which processing that Personal Information is required by the application.
- Many APIs deal with personal information (e.g., location, IP address, MSISDN, etc), so it is necessary a technical solution to manage the user consent.
- And to do so...
 - We need to know who the user is.
 - We need to register the user consent under a certain purpose.
- How is it proposed to do it?
 - Proposed solution follows **OpenID Connect** (OIDC) standard for user authentication → <u>CIBA</u> (Client-Initiated Backchannel Authentication) Flow.
 - A Consent API is proposed for third-parties to register the consent collected from the user.



High-level view of how concepts are related



- Scopes tell what resources (e.g. APIs) are accessible under those scopes.
- A purpose has associated a set of personal information scopes (pi-scopes).
 - These are scopes which refer to resources providing access to Personal Information and thus requiring special protection.
 - Getting access to pi-scopes MUST always be done by explicitly declaring a purpose.
- Scopes are assigned to an application, granting access to APIs and other resources.
- Purposes are assigned to an application.

Therefore, an application will have access to:

- APIs or resources covered by the assigned scopes
- APIs or resources providing personal information under the assigned purpose (because of the pi-scopes included in that purpose):
 - Only if user has granted her consent for that purpose
 - Which may happen automatically (automatic purposes) or may need an explicit user action to provide consent. In the second case, the consent is created and stored and will be validated during authorization.



High level journey example - terms & conditions

Decoupled from the access to the PI Pl resource resources consumed on behalf of Marketplace/App User logs in and the userthat BE registers user App collects **Consume PI** granted consent user consent to consent resource **Provision Phase** access Pl under Consent API e.g. CAMARA WiFi (done once, App, purposes... QoS API certain purpose CIBA flow (OIDC) CIBA flow (OIDC) App launched Marketplace/App back-Marketplace/App backend requests access to PI end requests access for resource under required user consent purpose management User consent Out-of-band collection is backchannel user delegated to third authentication (no Non-authorized requests (no consent party application registered) will be directly rejected at the user interaction from (e.g. T&C)

consumption device)

Authorization entry point (Auth Server)



CIBA flow (OIDC)

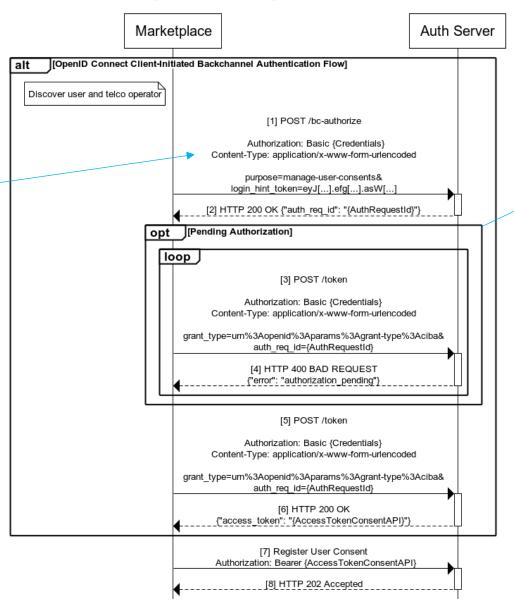
Consent is registered once.

Additional purpose query parameter to request a list of purposes in the authorization request (e.g. manage-user-consents).

login_hint_token: A signed JWT token containing information to identify the end-user.

```
{
  "aud": "https://auth.example.baikalplatform.com/",
  "iss": "https://marketplace-issuer",
  "exp": 1504807731,
  "iat": 1504804131,
  "identifier_type": "ip",
  "identifier": "66.77.88.99"
}
```

Obtaining Tokens to Manage Consents with CIBA



The Auth Server may identify the user and choose a channel to authenticate the user and authorize the request...

If user has not been authenticated or has not authorized the request yet, an authorization pending error is returned.



CIBA flow (OIDC)

The call must be done on behalf of the user that granted consent.

Additional purpose query parameter to request a list of purposes in the authorization request (e.g. improve-network-performance).

acr_values the acr values that the IdP is going to use for processing the authentication request.

login_hint_token: A signed JWT token containing information to identify the end-user.

```
"aud": "https://auth.example.baikalplatform.com/",
"iss": "https://marketplace-issuer",
"exp": 1504807731,
"iat": 1504804131,
"identifier_type": "ip",
"identifier": "66.77.88.99"
```

* The values for the strings in the identifier_type claim could be: ip, phone_number and sub.

Consume a Network API

