

## Data resumption cases

Key	Summary	Description
APP LINK -239 77	[Data Resumption]: Policies rules check must be performed before Resumption process	Resumption check must be performed after all Policies rules and DUPLICATE_NAME check performed. In case of invalid registration because of Policies rules resumption data must not be clean up. For more details about rules see RegisterAppInterface requirements.
APP LINK -159 87	[Data Resumption] Application data must not be resumed	Application data <b>must not</b> be resumed if the application unregistered itself gracefully (e.g. user turn off mobile app via UnregisterAppInterface) OR it's unregistered by SDL via OnAppUnregistered() notification by any reason.
APP LINK -159 58	[Data Resumption] hmi_appID must be the same for the application between ignition cycles	If the application is connected from the same device and with the same Policy AppID number, SDL must assign the same internal_hmi_appID for the application between ignition cycles.
APP LINK -156 90	[Data Resumption]: SDL data resumption process start not earlier than Duplicate name and policy table permissions check	The check of hashID value of RegisterAppInterface <b>must</b> be performed not earlier than DuplicateName and PolicyTable restrictions check during application registering.
APP LINK -156 57	[Data Resumption]: Data resumption on Unexpected Disconnect	SDL must initiate data resumption process after the app's RegisterAppInterface_request with <b>valid &lt;hashID&gt;</b> in case the application has been disconnected as a result of Unexpected transport disconnect not later than 3 ignition consecutive Ignition cycles before.  <i>Information:</i> if app didn't register during 3 ign consecutive cycles -> SDL must delete app's related data from resumption list
APP LINK -156 34	[Data Resumption]: Data resumption on IGNITION OFF	SDL must initiate data resumption process after the app's RegisterAppInterface_request with valid hashID in case the application has been disconnected on Ignition Off not later than 3 ignition consecutive cycles before.  <i>Information:</i> if app didn't register during 3 ign consecutive cycles -> SDL must delete app's related data from resumption list

6 issues

## Data persistence

Key	Summary	Description
APP LINK -159 92	[Data Resumption] Data persistence for an application failed to be re-registered	SDL <b>must not</b> clean up application-persistent data if the application failed to be registered by DUPLICATE_NAME reason after unexpected_disconnect/IGN_OFF.
APP LINK -159 91	[Data Resumption] Persistence Data clean up trigger	SDL <b>must</b> wait for the reconnection and store the app-related data for resumption during three consecutive ignition cycles only and clear all app-persisted data on the 4th ignition on.
APP LINK -157 06	[Data Resumption]: OnAwakeSDL in terms of resumption	In case SDL receives OnAwakeSDL notification from HMI after the preceding OnExitAllApplications(SUSPEND), SDL <b>must</b> : 1. Store an application resumption-related data it received from the application during SUSPEND state into DB 2. Send a single OnHashChanged notification with new hashID to mobile app  <b>Note:</b> data persistence on a file system happens according to AppSavePersistentDataTimeout
APP LINK -157 03	[Data Resumption]: OnExitAllApplications (IGNITION_OFF) in terms of resumption	In case SDL receives OnExitAllApplications(IGNITION_OFF), SDL must clean up any resumption-related data obtained after OnExitAllApplications( SUSPEND). SDL must stop all its processes, notify HMI via OnSDLClose and shut down.  <b>Note:</b> OnAppInterfaceUnregistered(Ignition_OFF) <b>must not</b> be sent to mobile applications on IGN_OFF

In case SDL receives OnExitAllApplications(SUSPEND) from HMI, SDL **must**

1. Store all applications resumption-related data on the file system
2. Store correct <isMedia> value for each app received in *RegisterAppInterface* request and allowed by Policies to resumption database
3. Send BC.OnPersistenceComplete to HMI on data persistence complete
4. Continue generating resumption related data (in case getting some from mobile application) without sending OnHashChanged notification to mobile app.

APP LINK -157 02 [Data Resumption]: OnExitAllApplications (SUSPEND) in terms of resumption

**Information:** Applink Service terminates (that is, closes) both BT and USB transports together with sending BC.OnExitAllApplications (SUSPEND). SDL is not in time to unregister all of the applications. SDL will resume them upon the next ign\_on. Such operation is ok with Ford (confirmed in e-mail attached).

Every *AppSavePersistentDataTimeout* milliseconds defined in smartDeviceLink.ini file, SDL **must** persist in the database <applID>, <hmi\_appID>, <deviceID>, <isMedia> (value requested by app and allowed by Policies) and a successful application data updates by the following mobile API :

- AddCommand (Menu +VR)
- AddSubMenu
- CreateInteractionChoiceSet
- SetGlobalProperties
- SubscribeButton
- SubscribeVehicleData
- SubscribeWayPoints (added per APPLINK-21898)
- DeleteCommand,
- DeleteSubMenu,
- DeleteInteractionChoiceSet,
- ResetGlobalProperties,
- UnsubscribeButton,
- UnsubscribeVehicleData
- UnsubscribeWayPoints (added per APPLINK-21904)

APP LINK -156 89 [Data Resumption]: Data Persistence

**Note:** HMI mustn't store any HMI persistent data, except VR grammars. VR grammars should be stored on HMI side, any other persistent data should be stored on SDL's side.

6 issues

## HashID and resume concept

Key	Summary	Description
APP LINK -156 86	[Data Resumption]: SDL data resumption failure	<p>In case &lt;hashID&gt; of the application registered <b>doesn't equal</b> stored value for the application, SDL <b>must</b> interrupt data resumption process:</p> <ol style="list-style-type: none"> <li>a. return <i>RegisterAppInterface_response</i> (success: true, resultCode: RESUME_FAILED)</li> <li>b. notify HMI by <i>OnAppRegistered</i> (resumeVrGrammars:false) to restore VRgrammars persisted on HMI</li> <li>c. clean up all stored application related data: <ul style="list-style-type: none"> <li>• AddCommand (Menu +VR)</li> <li>• AddSubMenu</li> <li>• CreateInteractionChoiceSet</li> <li>• SetGlobalProperties</li> <li>• SubscribeButton</li> <li>• SubscribeVehicleData</li> <li>• SubscribeWayPoints</li> </ul> </li> <li>d. start data persistence process from the beginning for the current application (see also )</li> </ol> <p><b>Note:</b> It's application's responsibility to re-send all application data to SDL after RESUME_FAILED  <b>Note:</b> HMI cleans up persisted Vrgrammars data by itself after getting <i>resumeVrGrammars:false</i></p>
APP LINK -156 83	[Data Resumption]: SDL data resumption SUCCESS sequence	<p>In case hashID of the application registered equals stored value for the application, SDL <b>must</b> register the application according to standard process and:</p> <ol style="list-style-type: none"> <li>1. Return RegisterAppInterface response of (success: true, resultCode: SUCCESS, info "Resume succeeded") after successful registering</li> <li>2. Notify HMI by <i>OnAppRegistered</i> (resumeVrGrammars:true) to restore VRgrammars persisted on HMI</li> <li>3. Restore application related data: <ul style="list-style-type: none"> <li>• AddCommand (Menu +VR)</li> <li>• AddSubMenu</li> <li>• CreateInteractionChoiceSet</li> <li>• SetGlobalProperties</li> <li>• SubscribeButton</li> <li>• SubscribeVehicleData</li> <li>- SubscribeWayPoints</li> </ul> </li> <li>4. Send the following restored data to HMI right after OnAppRegistered notification sent to HMI: <ul style="list-style-type: none"> <li>- AddCommand (<b>Menu only</b>)</li> </ul> </li> </ol>

- AddSubMenu
- CreateInteractionChoiceSet (AddCommands of the ChoiceSets)
- SetGlobalProperties
- OnButtonSubscription (isSubscribed=true)
- SubscribeVehicleData
- SubscribeWayPoints

SDL **must** update hashID value and send OnHashChange() to mobile app after each of the following RPCs being executed with "success:true" and responded to mobile application:

- AddCommand (Menu or/and VR)
- AddSubMenu
- CreateInteractionChoiceSet
- SetGlobalProperties
- SubscribeButton
- SubscribeVehicleData
- SubscribeWayPoints (added by APPLINK-21644, APPLINK-21629)
- DeleteCommand,
- DeleteSubMenu,
- DeleteInteractionChoiceSet,
- ResetGlobalProperties,
- UnsubscribeButton,
- UnsubscribeVehicleData
- UnsubscribeWayPoints (added by APPLINK-22888, APPLINK-21644)

APP LINK-156-82 [Data Resumption]: OnHashChange

**Information:**

1. "success:true":
  - > WARNINGS,
  - > SUCCESS
  - > UNSUPPORTED\_RESOURCE (*only in case <Interface> is supported by system -> HMI respond with <Interface>. IsReady (available:true) to SDL*)
  - > RETRY
  - > SAVED
  - > WRONG\_LANGUAGE

APP LINK-156-81 [Data Resumption]: RegisterAppInterface with no or wrong hashID

SDL **must not** resume application data if receives no hashID (by RegisterAppInterface request) or the value different from hashID stored on SDL for the appropriate application. In case SDL has previously stored data for the mentioned application, the app-persisted data **must** be cleaned up by SDL.

APP LINK-156-70 [Data Resumption]: RegisterAppInterface with hashID the same as stored before

SDL **must** resume persistent data only if receives the same value of hashID (by RegisterAppInterface request) as stored before for this application.

5 issues

**Not implemented yet**

Key	Summary	Description
APPL INK-28367	[AddCommand] [Data Resumption] SDL must restore AddCommands in the same order as they were created by mobile app	<p>In case SDL receives <i>AddCommand</i> from mobile app (<i>UI+VR, only UI, only VR</i>)</p> <p><b>SDL must:</b>  <u>generate</u> &lt;internal_consecutiveNumber&gt;  <u>assign</u> this &lt;internal_consecutiveNumber&gt; to <b>each</b> <i>AddCommand</i> requested by app  <u>restore</u> <i>AddCommand</i> by this &lt;internal_consecutiveNumber&gt; <u>during data resumption</u> (<i>meaning: SDL must send UI.AddCommands to HMI by &lt;internal_consecutiveNumber&gt; one by one from resumption list</i>)</p> <p><b>Information:</b></p> <ol style="list-style-type: none"> <li>1. AddCommands (F-S) requirements are still applicable</li> <li>2. Data Resumption(F-S) requirements are still applicable</li> <li>3. The resumption list should stay <u>without any changes</u> (<i>should NOT be updated with &lt;internal_consecutiveNumber&gt;</i>)</li> </ol>

**Non-functional requirements**

Key	Summary	Description
-----	---------	-------------

## Related HMI API

▼ [OnAppRegistered xml structure](#)

**Reference #:** APPLINK-16117

Key	Summary	Description
-----	---------	-------------

```

<function name="OnAppRegistered" messagetype="notification">
  <description>Issued by SDL to notify HMI about
new application registered.</description>
  <param name="application" type="Common.
HMIApplication" mandatory="true">
    <description>The information about application
registered. See HMIApplication. </description>
  </param>
  <param name="ttsName" type="Common.TTSCchunk"
minsize="1" maxsize="100" array="true" mandatory="false"
  >
    <description>
      TTS string for VR recognition of the mobile
application name, e.g. "Ford Drive Green".
      Meant to overcome any failing on speech
engine in properly pronouncing / understanding app name.
      May not be empty.
      May not start with a new line character.
      Not unique value
    </description>
  </param>
  <param name="vrSynonyms" type="String" maxlength="
40" minsize="1" maxsize="100" array="true" mandatory="fa
lse">
    <description>
      Defines an additional voice recognition
command.
      Must not interfere with any name of
previously registered applications(SDL makes check).
    </description>
  </param>
  <param name="resumeVrGrammars" type="Boolean"
mandatory="false">
    <description>The flag if VR grammars resume is
required</description>
  </param>
  <param name="priority" type="Common.AppPriority"
mandatory="false">
    <description>Send to HMI so that it can
coordinate order of requests/notifications
correspondingly.</description>
  </param>
</function>

```

#### HMIApplication

is changed by APPLINK-30845 -> see req\_1 for openSDL ONLY

```

<struct name="HMIApplication">
  <description>Data type containing information about

```

APPL [HMI API]  
INK- OnAppRe  
16117 gistered

```
application needed by HMI.</description>
  <param name="appName" type="String" maxlength="100"
mandatory="true">
  <description>The mobile application name, e.g. "Fo
rd Drive Green".</description>
</param>
  <param name="ngnMediaScreenAppName" type="String"
maxlength="100" mandatory="false">
  <description>Provides an abbreviated version of
the app name (if needed), that may be displayed on the
NGN media screen.</description>
  <description>If not provided, the appName should
be used instead (and may be truncated if too long)<
/description>
</param>
  <param name="icon" type="String" mandatory="false">
  <description>Path to application icon stored on
HU.</description>
</param>
  <param name="deviceInfo" type="Common.DeviceInfo"
mandatory="true">
  <description>The ID, serial number, transport
type the named-app's-device is connected over to HU.<
/description>
</param>
  <param name="policyAppID" type="String" maxlength="5
0" minlength="1" mandatory="true">
  <description>Policy ID(=the appID the application
registers with) of registered application.</description>
</param>
  <param name="ttsName" type="Common.TTSChunk"
minsize="1" maxsize="100" array="true" mandatory="false"
>
  <description>
    TTS string for VR recognition of the mobile
application name, e.g. "Ford Drive Green".
    Meant to overcome any failing on speech engine
in properly pronouncing / understanding app name.
    May not be empty.
    May not start with a new line character.
    Not unique value
  </description>
</param>
  <param name="vrSynonyms" type="String" maxlength="40"
minsize="1" maxsize="100" array="true" mandatory="false"
>
  <description>
    Defines an additional voice recognition command.
    Must not interfere with any name of previously
registered applications(SDL makes check).
  </description>
</param>
  <param name="appID" type="Integer" mandatory="true">
```

```

        <description>Unique (during ignition cycle) id of
the application. To be used in all RPCs sent by both HU
system and SDL</description>
    </param>
    <param name="hmiDisplayLanguageDesired" type="Common
.Language" mandatory="false">
        <description>The language the application intends
to use on HU </description>
    </param>
    <param name="isMediaApplication" type="Boolean"
mandatory="false">
        <description>Indicates whether it is a media or a
non-media application.</description>
    </param>
    <param name="appType" type="Common.AppHMIType"
minsize="1" maxsize="100" array="true" mandatory="false">
        <description>List of all applicable app types
stating which classifications to be given to the app.<
/description>
        <description>e.g. for platforms like GEN2, this
determines which "corner(s)" the app can populate.<
/description>
    </param>
    <param name="greyOut" type="Boolean" mandatory="fals
e">
        <description>Indicates whether application should
be dimmed on the screen.</description>
        <description>Applicable only for apps received
through QueryApps and still not registered.<
/description>
    </param>
    <param name="requestType" type="Common.RequestType"
minsize="0" maxsize="100" array="true" mandatory="false">
        <description>The list of SystemRequest's
RequestTypes allowed by policies for the named
application</description>
        <description>(the app's SystemRequest sent with
RequestType out of this list will get 'disallowed'
response from SDL).</description>
        <description>If SDL sends an empty array - any
RequestType is allowed for this app.</description>
        <description>If SDL omits this parameter - none
RequestType is allowed for this app</description>
        <description>(either this is a pre-registered app
or such is dictated by policies).</description>
    </param>
</struct>

```

DeviceInfo

```

<struct name="DeviceInfo">
  <param name="name" type="String" mandatory="true">
    <description>The name of the device connected.<
  /description>
  </param>
  <param name="id" type="String" mandatory="true">
    <description>The ID of the device connected:
either hash of device's USB serial number(in case of
USB connection) or has of device's MAC address(in case
of BlueTooth or WIFI connection</description>
  </param>
  <param name="transportType" type="Common.
TransportType" mandatory="false">
    <description>The transport type the named-app's-
device is connected over HU(BlueTooth, USB or WiFi). It
must be provided in OnAppRegistered and in
UpdateDeviceList</description>
  </param>
  <param name="isSDLAllowed" type="Boolean" mandatory="f
alse">
    <description>Sent by SDL in UpdateDeviceList. '
true' - if device is allowed for PolicyTable Exchange;
'false' - if device is NOT allowed for PolicyTable
Exchange </description>
  </param>
</struct>

```

#### TransportType

```

<enum name="TransportType">
  <description>Lists of the transport types used for
device connection to HU.</description>
  <element name="BLUETOOTH"/>
  <element name="USB_IOS"/>
  <element name="USB_AOA"/>
  <element name="WIFI"/>
</enum>

```

## Diagrams

▼ AddCommand ordering for data resumption



