

JOEL FISCHER - MARCH 6, 2019

---

# SDL-0216 WIDGET SUPPORT WORKSHOP

## ISSUE #1 – POLICIES, SDL SERVER, & DEVELOPER PORTAL

- ▶ `CreateScreen` and others should be allowed in `HMI\_NONE`, requiring policy changes
- ▶ Open SDL policy server and developer portal need updates to support this
  - ▶ Policy server needs a new functional group for widgets RPCs
  - ▶ Developer portal needs a new request for widgets permissions
- ▶ Proposal ought to describe these changes and UX ramifications of not following these new defaults (using “Should”, not “Must” language)
- ▶ This issue also affects App Services

## ISSUE #2 - EXPLICIT WIDGET / APP LIFECYCLE CHART

- ▶ As agreed upon, there should be a chart of widget and app lifecycle.
  - ▶ For example, what happens when the widget appears, but app is in `HMI\_NONE`?
  - ▶ Can app be in `LIMITED`? In what situations?

## ISSUE #3 – RPC REJECTION FOR “DUPLICATING” WIDGETS

- ▶ Should Core or HMI handle rejecting RPCs for widgets that are duplicating main screen content?

## ISSUE #4 - DISPLAY CAPABILITIES FOR ALTERNATE SCREENS

- ▶ Should this be placed in `RegisterAppInterfaceResponse` like the main app?
- ▶ Should this use `GetSystemCapability` and `OnSystemCapability`
  - ▶ Can widget dimensions and capabilities change at runtime (e.g. smaller / larger widget placement on HMI)

## ISSUE #5 – HOW ARE MAIN SCREEN / WIDGET RPC PERMISSIONS MANAGED

- ▶ Many RPCs are only available for main screen apps, not widgets
- ▶ Is this limitation hardcoded into Core?
  - ▶ Or is this a policy change?

## ISSUE #6 - "DUPLICATE" VS. "MIRROR"

- ▶ Is "Duplicate" (current) or "Mirror" (maintainer-proposed) better language?
  - ▶ ``duplicateScreenId`` vs. ``mirrorScreenId``
- ▶ Functionality is for the HMI to automatically show in-perpetuity main screen content (within display capability limits) on the widget

## ISSUE #7 - WIDGET NAMES

- ▶ When an app has multiple widgets, they need to be differentiated in some way
  - ▶ e.g. Weather app with "24 hour" widget and "7 day" widget
- ▶ Maintainer proposal: Add `name` String parameter to `CreateScreen`.



## ISSUE #8 - RESUMING WITH WIDGETS

- ▶ If an app that has created screens is resumed, what happens with widget content?
  - ▶ Resuming is primarily for ChoiceSets & Menu Commands
  - ▶ Author Note: "They are included; with all the content and screen IDs."
  - ▶ Proposal should be updated to reflect this

## ISSUE #9 - CLOUD APP ISSUES

- ▶ Cloud apps are incapable of fixing the issues listed in issue #1 in their current implementation
  - ▶ Cloud apps only connect to their server after entering HMI\_FULL, preventing widgets and app services from registering in HMI\_NONE

## ISSUE #10 - CONNECTION WITH APP SERVICES

- ▶ Maintainers believe that widgets should have a tie to app services
  - ▶ e.g. "When my app is the primary service, this is the widget corresponding to my service"
  - ▶ This enables powerful features like always showing widgets connected to your current media, navigation, messaging, and weather service
  - ▶ Could be a boolean in `CreateScreen`, e.g. `isPrimaryWidget`, the last widget created with this boolean as `true` is the primary widget for this use case