

App Framework in IVI EG

Automotive Grade Linux / Collabora

Daniel Stone <daniels@collabora.com>



COLLABORA

Open First



What is the App Framework?



~~What~~ is the App Framework? Why is the App Framework?

Current App Framework in IVI

- App Framework developed by IoT.bzh
- Developed for AGL IVI applications
- Based on JSON + WebSockets
- Communication separated between services in different processes
- Clients and services can be written in any language
- Heavily integrated with SMACK



App framework: the good

- Communication separated between services in different processes
- Allows common authentication between security domains
- Clients and services can be written in any language
- Portable to multi-ECU solutions



App framework: the bad

- Communication separated between services in different processes
 - ... but is JSON + WebSockets the best transport mechanism?
 - IoT.bzh proposing to replace JSON with binary serialisation due to performance overhead
 - Not always the best signalling for every usecase



App framework: the bad

- Allows common authentication between security domains
 - ... but that authentication is heavily based on SMACK
 - Reliant on UNIX process model and privilege inheritance
 - Complex, difficult to get right
 - (to the point it's a FAQ)
 - Still no support for WAM-like usecases



App framework: the bad

- Clients and services can be written in any language
 - Few helpers and bindings for many languages
 - Lacks rich features compared to other IPC and RPC systems: deeper API integration (FFI, callbacks), service enumeration and discovery



App framework: the bad

- Portable to multi-ECU solutions
 - ... but, SMACK
 - Enumeration and discovery also undefined



App framework: the ugly

- Requires bespoke effort and binding for every language and app
- No community support buy-in outside AGL-IVI & IoT.bzh
- AGL app framework is not production ready (lacks features, performance, etc)
- Toyota proposing to replace app framework (?) as part of PR effort
- IC EG proposing to avoid IVI app framework



App framework: the ugly

- Requires bespoke effort and binding for every language and app
- No community support buy-in outside AGL-IVI & IoT.bzh
- AGL app framework is not production ready (lacks features, performance, etc)
- Toyota proposing to replace app framework (?) as part of PR effort
- IC EG proposing to avoid IVI app framework
- ... a lot of effort for little help



App framework: the ... good, again?

- AGL production-readiness model emphasises tier-1 needs



App framework: the ... good, again?

- AGL production-readiness model emphasises tier-1 needs
- AGL IC effort has own clearly defined architecture



App framework: the ... good, again?

- AGL production-readiness model emphasises tier-1 needs
- AGL IC effort has own clearly defined architecture
- AGL IVI supposed to be 'innovation' area



App framework: the ... good, again?

- AGL production-readiness model emphasises tier-1 needs
- AGL IC effort has own clearly defined architecture
- AGL IVI supposed to be ‘innovation’ area
 - New technology development
 - Emphasis on collaboration with upstream open community
 - Success stories: PipeWire, Wayland, etc



App framework: getting to good?

- Restart from fundamentals, focus on base requirements
- Activity, lifecycle, lifetime management of services
- Authentication domains
- Inter-service discovery, enumeration, connection (like Android intents and Binder, D-Bus, cloud services)



App framework: getting to good?

- Restart from upstream OSS projects
- Consider systemd sessions and scopes for lifecycle
 - Works with standard UNIX services
 - Works with modern container workloads e.g. CRI
 - Uses cgroups for isolation and separation
 - Monitoring process lifecycle, bidirectional notification



App framework: getting to good?

- Restart from upstream OSS projects
- Reconsider authorization strategy
 - Investigate alternate authorization mechanisms not based on single LSM
 - Use of privileged sockets (as in Wayland privilege model), network namespaces, to differentiate different services
 - Alternate authorization mechanisms such as OAuth2/JWT or ephemeral certificates (as in Kubernetes) for remote services



App framework: getting to good?

- Restart from upstream OSS projects
- Strongly reconsider IPC mechanisms
 - Most other IPC services (Binder, D-Bus, gRPC, others) already handle common problems
 - Authorization, performance, tracing: key considerations
 - Service enumeration and discovery: helpful addition on top of existing app FW
 - Investigate domain-specific solutions: like WirePlumber for audio, Wayland for window management, etc



App framework: getting to good?

- Restart from upstream OSS projects
- Accept limitations of current resourcing and funding
- We do not have enough engineering time to produce a complete app framework from scratch
- AGL should be close to open upstream communities
- Focus on the glue: reuse, improve, integrate, iterate!

App framework: when will we be good?

- Detailed time estimates for new app framework:



App framework: when will we be good?

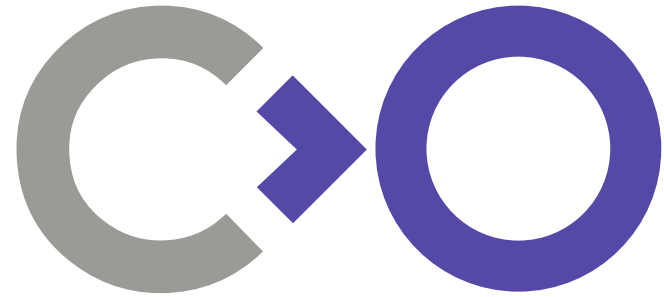
- Detailed time estimates for new app framework:
- ...



App framework: when will we be good?

- Detailed time estimates for new app framework:
- ...
- Please discuss! :)





Thank you!



COLLABORA

Open First