

Product Ready Specification

目次 Contents

1. 目的/Purpose.....	4
2. 適用範囲/Scope of Application.....	6
3. 関連文書と用語集/Related Specifications Terms and Definitions.....	6
4. 全体アーキテクチャ/Entire Architecture	8
5. 要件	9
5.1. Health Monitoring	9
5.1.1. AGL spec and implimentation	9
5.1.2. System start/stop, and Anomaly detection and Abnormal processing ...	10
5.1.3. ResourceManagement	13
5.1.4. Logger Service	16
5.2. Automotive Communication Devices.....	17
5.2.1. AGL spec and implimentation	17
5.2.2. CAN	18
5.2.3. Positioning	20

変更履歴 Changes

Ver	Date	Change point	Author

1. 目的/Purpose

AGL仕様書 **v1** と AGL 実装 **v9.0.2** に大きなギャップがある

There is a large gap between AGL specifications v1 and AGL implementation v9.0.2

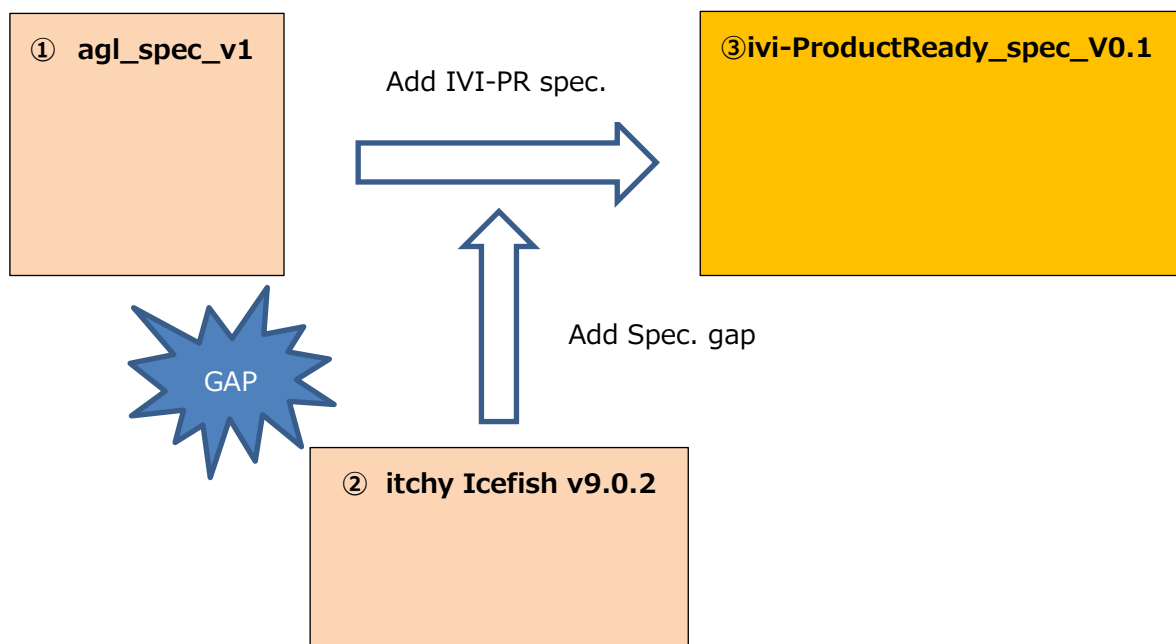
- ① **agl_spec_v1 (2015)**
- ② **itchy Icefish v9.0.2 (2020)**

本仕様書では、①、②の仕様差分と、ProductReady としての新規仕様を追加し

- ③ **ivi-ProductReady_spec_V0.1** を作成する

In this specification, the difference specification of ① and ② and the new specification as Product Ready are added.

- ③ Create **ivi-ProductReady_spec_V0.1**



2. 適用範囲/Scope of Application

IVI-ProductReady-Profile

3. 関連文書と用語集 /Related Specifications Terms and Definitions

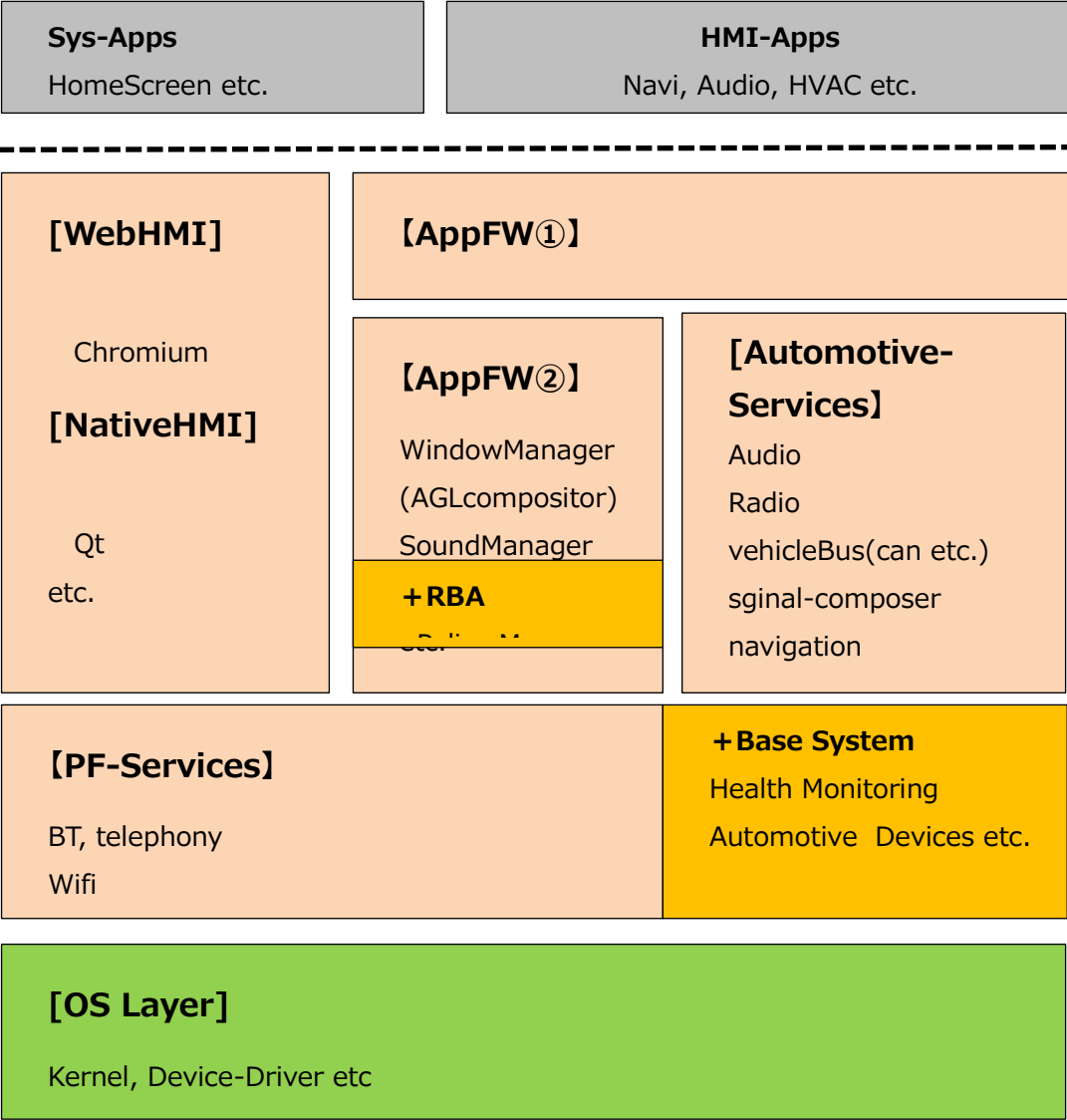
関連文書 Related Specifications

	文書名
	agl_spec_v1_280515.pdf

用語 Terms and Definitions

語句 Terms	説明 Explanation

4. 全体アーキテクチャ/Entire Architecture V0.1 (Trial)



現時点で V0.1 に追加予定のパッケージ
 Packages currently planned to be added to V0.1

5. 要件/Requirements

5.1. Health Monitoring

5.1.1. AGL spec and implimentation

[agl_spec_v1]

5.1.5 Health Monitoring

Platform monitorting service such as watchdog or active monitoring

Spec v1 では車両信頼性を確保するための「Health Monitoring」を要件としているが、具体的な要件がない。

Spec v1 requires "Health Monitoring" to ensure vehicle reliability, but there is no specific requirement.

[itchy Icefish v9.0.2]

車両信頼性に対応する機能として、“HertBeat”のみ実装されている

"HertBeat" is implemented as a function that supports vehicle reliability.

[ivi-ProductReady_spec_V0.1]

IVI-PR では信頼性要件を詳細化し、IVI に追加する

IVI-PR refines reliability requirements and adds to IVI

5.1.2. System start/stop, and Anomaly detection and Abnormal processing

【Requirements List】

#	Term	Description
1	システム起動 [System start]	Configuration ファイルに設定された順序に従って他の常駐型サービスを起動 Start other resident services according to the order set in the configuration file
2	システム終了 [System shutdown]	Configuration ファイルに設定された順序に従って他のサービスを終了 Terminate other services according to the order set in the configuration file
3	異常検出と異常処理 [Anomaly detection and Abnormal processing]	<p>異常検知 [Anomaly detection]</p> <ul style="list-style-type: none"> ・サービスの生存確認による異常検出 [Abnormality detection by confirmation of survival] ・サービス異常終了 [Services abnormally shutdown] ・システムメモリ枯渇の検知[System memory shortage] <p>異常処理 [Abnormal processing]</p> <ul style="list-style-type: none"> ・システムリセット [System-Reset] ・プロセス再起動 [Process-Restart] ・異常ログ出力 [Abnormal log output] ・Rob ログ保存 [Store Rob(*1) log] <p>その他 [Others]</p> <ul style="list-style-type: none"> ・モデル切り替え [Change model] ・電源状態遷移 [Power State Transition](*2)

(*1) Rob (Record of Behavior)

ドライバーに違和感を覚えさせる制御を行った状況などの記録

Own system sound state, record status such as controlled let the driver feel uncomfortable.

(*2) 独自の電源マイコンとの連携が必要なため、スタブで対応します。

Since it is necessary to link with a unique power supply microcomputer, stubs

are used.

- ・停止 Stop
- ・乗車中 Pre-Boot
- ・駐乗車起動 Background-Boot
- ・通常起動 Normal-Boot

【Implementation Example】

(1) システム起動 [System start]

system_manager は起動すると、Configuration ファイルに設定された順序に従って他の常駐型サービスを起動する。サービスはグループ化することができ、system_manager はこのグループ単位でサービスを起動する。

When system_manager is started, it starts the other resident services in line with the order in Configuration file. These services can be grouped and system_manager starts the services per the group.

ただし、AGL では Linux 標準の Sysyemd を採用しているため、本機能は BaseSystem 内のサービス起動にのみ使用する。

However, since AGL uses the Linux standard Sysyemd, this function is used only for starting services within the BaseSystem.

(2) システム終了 [System shutdown]

system_manager は、Configuration ファイルに設定された順序に従って他のサービスを終了する。システム終了は、Power Supply Sub-System（ハードウェアユニット）からの要求、サービスからの要求、及び異常検出等に基づいて行う。

system_manager terminates the other services in line with the order in Configuration file. The system shutdown is processed based on Power Supply Sub-System requests, the service requests and malfunction detections etc.

(3) 異常検出 [Malfunction detections]

system_manager は各種異常を検出すると、その異常を通知し、種々の異常処理を行う。異常処理の内容は、予め静的に Configuration ファイルとして規定する。規定する種類としては、システムリセット、再起動（事象発生のプロセス）、発生事象の履歴保存等である。

When system_manager detects every kinds of the malfunctions, it executes the various failure processing. The contents of the failure processing are statically

prescribed in Configuration file in advance. The prescribed ones are the system reset, the restart (event occurrence processes) and the occurred event log storing.

A) Heart Beat

管理対象サービスとの間で定期的に通信（Heart Beat リクエスト）することによって生存確認を行う。サービスの応答がないと異常と判断し、Configuration ファイルの設定に従い、サービスの再起動、終了、あるいはシステムリセットを行う。

system_manager confirms the behaviors by the regular communication (Heart Beat request) to the controlled services. When it has no responses from the services, system_manager recognizes the malfunctions and restarts and terminates the services or resets the systems in line with the set-up in Configuration file.

B) サービス異常終了 [Services abnormally shutdown]

起動したプロセスの状態を監視する。異常終了を検知すると、Configuration ファイルの設定に従い、サービスの再起動、終了、あるいはシステムリセットを行う。

system_manager monitors the process status after start. When it detects the abnormal shutdowns, it restarts and terminates the services or resets the systems in line with the set-up in Configuration file.

C) システムメモリ枯渇 [System memory shortage]

resource_manager と連携してシステムメモリを監視する。一定時間規定の閾値を下回るとシステムメモリ不足と判定し、システムリセットを行う。

system_manager monitors the system memory in cooperation with resource_manager. When the value gets lower than the certain level for the specified time, it recognizes the system memory shortage and resets the system.

(4) 異常ログ出力 [Abnormal log output]

system_manager は、サービスに対して異常ログ出力機能を提供する。アプリケーションからの要求に応じて、保存対象のログを USB ストレージデバイス、SD カード等に保存する。

system_manager provides the function to output the abnormal log for the services. It stores the targeted log in USB storage device and SD card etc due to the requests from the applications.

(5) モデル切り替え [Change model]

System_manager はモデル仕様依存処理を config の inithook に実装し、libsmconf.so を置き換えることによりモデル切り替えを実現

system_manager implements model specifications-dependent processing in inithook of config and realizes a model change by rearranging libsmconf.so.

(6) 電源状態遷移 [Power State Transition]

電源状態に応じてシステムの動作を制御するため、power_service から電源状態遷移要求を受信し、管理下にある常駐型サービスに通知する。

In order to control the system according to the power state, notify the managed resident service of a power state transition received from power_service.

(7) Rob ログ保存 [Store Rob log]

システム再起動時に RoB ログを保存する機能を呼び出してリセット履歴を保存する。

Call the function to save RoB log at system restart and save reset history.

5.1.3. ResourceManagement

【Requirements List】

#	Term	Description
1	CPU 負荷の監視 [CPU load monitoring]	高負荷状態が一定時間異常継続している状態に至ると、CPU を占有する上位プロセスの LOGGING を行う。 execute LOGGING of the upper level process occupying CPU when the high load status is abnormally continued during the specified time.
2	システム情報の取得 [System information provision]	残メモリ情報、NAND フラッシュへの書き込み可否状態、ネットワークデバイスの通信量 Obtain the residual memory information, write access status to NAND flash and communication volume in the network device
3	システムメモリの監視 [System memory monitoring]	残メモリ量が閾値を下まわると、EVENT を発行 Issue an event when the residual memory falls below the certain level.

4	ウォッチドックタイマ(WDT)の更新 [Watch Doc Timer (WDT) Update]	マイコン内蔵の WDT により、ソフトウェアの暴走検出を行う。 Monitor the error detection for the software by WDT including microcomputer.
5	デバッグ表示情報の提供 [Debug display information provision]	デバッグ表示するための残メモリ情報、CPU 負荷情報、CMA の残メモリ情報を提 Provide the residual memory information, CPU load information and residual memory information for CMA
6	ACC-OFF 時の最小メモリ残量情報のログ出力 [Log output of the minimum residual memory information at ACC-OFF]	起動以降の最小メモリ残量情報を保持し、ACC-OFF を検出したときログ出力する。 Maintain the minimum residual memory information after the start and output the log at ACC-OFF detection.

【Implementation Example】

(1) CPU 負荷の監視 [CPU load monitoring]

CPU 負荷を監視し、高負荷状態が一定時間異常継続している状態に至ると、CPU を占有する上位プロセスの LOGGING を行う。

Monitor CPU load and execute LOGGING of the upper level process occupying CPU when the high load status is abnormally continued during the specified time.

(2) システム情報の取得 [System information provision]

残メモリ情報、NAND フラッシュへの書き込み可否状態、電源を入れてからのネットワークデバイスの通信量を取得する。

Obtain the residual memory information, write access status to NAND flash and communication volume in the network device after the power on.

(3) システムメモリの監視 [System memory monitoring]

システム全体のメモリを監視する。残メモリ量が閾値を下まわると、EVENT を発行する。

Monitor the memory in the whole system. Issue an event when the residual memory falls below the certain level.

(4) ウォッチドックタイマ(WDT)の更新 [Watch Doc Timer (WDT) Update]

MM マイコン内蔵の WDT により、ソフトウェアの暴走検出を行う。

FIFO および RR プロセスが、長時間 CPU を占有する状況が継続すると、異常と判断し WDT を発火させる。

Monitor the error detection for the software by WDT including MM microcomputer.

Recognizes the malfunction and triggers WDT when FIFO process and RR process keep occupying CPU for the long time.

(5) デバッグ表示情報の提供 [Debug display information provision]

デバッグ表示するための残メモリ情報、CPU 負荷情報、CMA の残メモリ情報を提供する。

これらは、ダイアグ画面からログ出力を Release から Debug に切り替えることにより LOG に出力される。

Provide the residual memory information, CPU load information and residual memory information for CMA for the debug display.

These are outputted to LOG by switching the log output to Debug from Release at the Diagnosis screen.

(6) ACC-OFF 時の最小メモリ残量情報のログ出力 [Log output of the minimum residual memory information at ACC-OFF]

起動以降の最小メモリ残量情報を保持し、ACC-OFF を検出したときログ出力する。

Maintain the minimum residual memory information after the start and output the log at ACC-OFF detection.

5.1.4. Logger Service

【Requirements List】

#	Term	Description
1	ログ保存要求機能 [Function to request the log storing]	不揮発メモリに保存する Save to non-volatile area (eMMC/USB/SD etc.)
2	周期読み込み機能 [Function to periodically read]	設定に従ってログの周期読み込みを行う Periodically read the log according to the settings

【Implementation Example】

(1) ログ保存要求機能 [Function to request the log storing]

他モジュールで作成したログを、保存要求受信時に読み込んで保存する。

At the receipt of the storing request read the log generated by other modules and save it.

(2) 周期読み込み機能 [Function to periodically read]

他モジュールで書き込まれたログを周期的に読み込む。

Periodically read the log written by other modules.

5.2. Automotive Communication Devices

5.2.1. AGL spec and implimentation

[agl_spec_v1]

7.5.5 Automotive Devices

Spec v1 では、車両で利用するデバイスを要件としているが、具体的な要件はない。

Spec v1 requires devices to be used in vehicles, but there are no specific requirements

[itchy Icefish v9.0.2]

AGL では通信デバイス（CAN、Positioning）に対して汎用的な実装を追加している。

AGL adds a general implementation for communication devices (CAN, Positioning).

(e.g.) AGL-Service CAN、Signal-Composer etc.

[ivi-ProductReady_spec_V0.1]

IVI-PR では、通信デバイスの要件に、車両で使用する独自の処理を追加する。

IVI-PR adds to the requirements of the communication device the unique processing used in the vehicle.

5.2.2. CAN

【Requirements List】

#	Term	Description
1	CAN を利用した車両データの送受信 Transmission and reception of vehicle data using CAN	CAN デバイスとのデータ送受信 Data transmission/reception with CAN device
2	通信途絶からの復帰等の信頼性機能 Reliability functions such as recovery from communication interruption	CAN データ通信途絶・復帰を監視する Check the can data communication suspension or resume

【Implementation Example】

(1)can_hal へ CAN データを送信する。

Send CAN data to can_hal.

(2)can_hal から CAN データを受信する。

Receive CAN data from can_hal.

(3)CAN データ通信途絶・復帰を監視する。

Check the can data communication suspension or resume.

(4)CAN コマンド制御処理。

Transmission control of CAN command.

(5)CAN サービスの Availability を通知する。

Notifies the CAN service availability.

(6)送信した CAN データを配送する(エコーバック)

Distribute CAN data transmitted by AVN.

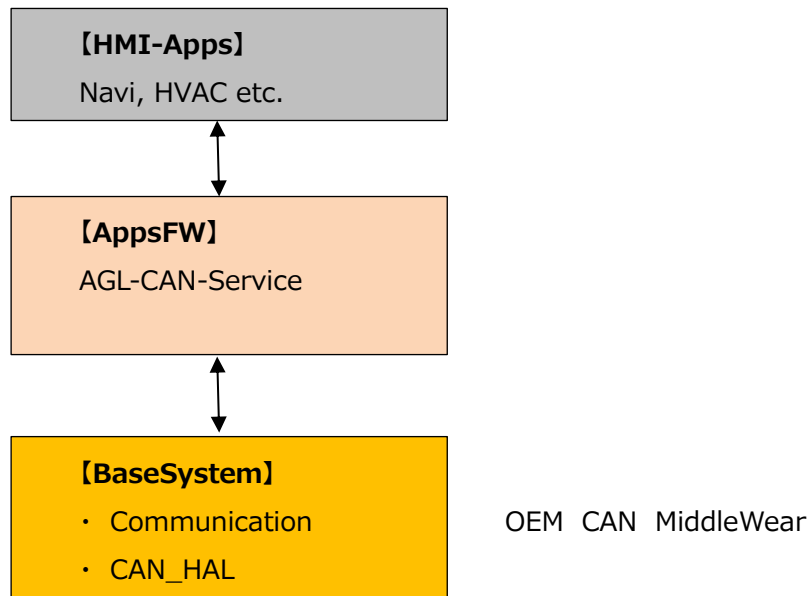


Fig. CAN implementation example

5.2.3. Positioning

【Requirements List】

#	Term	Description
1	GPS Device managemen	緯度経度、高度、速度(*1)、方位情報を提供する Provide longitude and latitude, altitude, speed, heading data
2	SensorDevice(*2) management	ジャイロスコープ、加速度計、車速パルス、車両リバースを提供する Provide Gyroscope, Accelerometer, Speed pulse, Vehicle Reverse

(*1)(*2)

AGL 標準 HW にデバイスがない場合、スタブで対応します。

If there is no device in the AGL standard HW, it will be supported by a stub.

【Implementation Example】

(1) GPS

- GPS 情報を取得する
Get GPS information
- GPS のリセットを要求する
Request GPS reset
- GPS 時刻を提供する
Provide GPS time

(2) Sensor

- Sensor データを提供する
Provide Sensor data