AGL Requirements Specification V2.0

AGL All-Member Meeting @ DRESDEN
October 2018

Toshikazu Oiwa
toshikazu_ohiwa@mail.toyota.co.jp

TOYOTA MOTOR CORPORATION

Who is Oiwa?

- Software engineer, expert in in-vehicle infotainment.
- Have been developing software for in-vehicle infotainment system such as apps, services since 1994.
- In charge of HMI-Framework development.

Table of Contents

Background

Background

➢ Merits of “Code First“ approach

- Unification by source code

  Elimination of producing functional bugs caused by different understanding of requirements on documents.

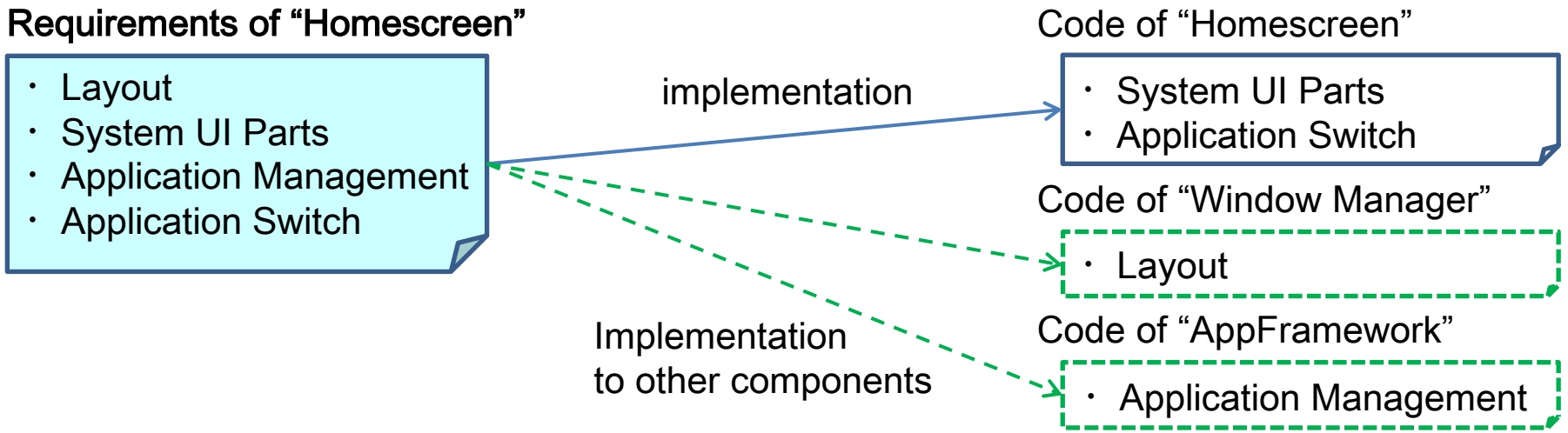
- Rapid development of function

- Collateral of behavior identity

Issues

Issue #1

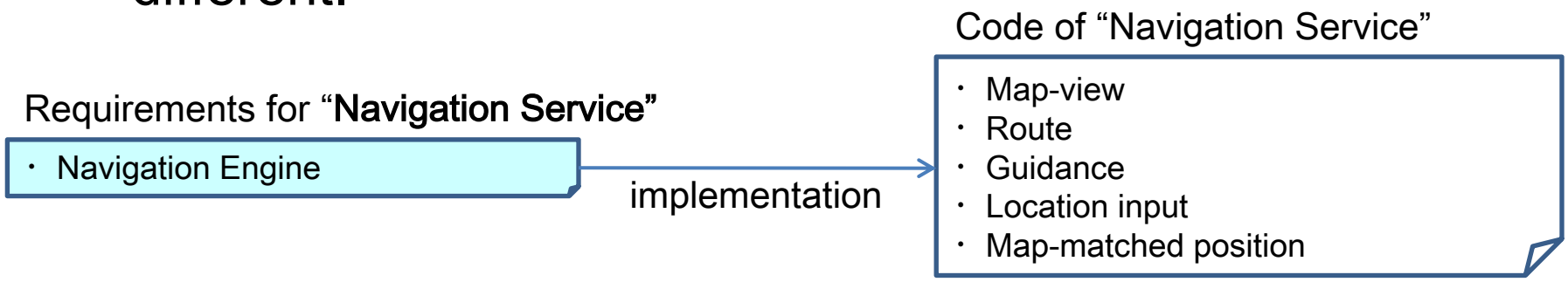
- Increase of the gap with the source code
  - Daily evolution of source code
  - More than 3 years since it was released
    - Derived implementation are different



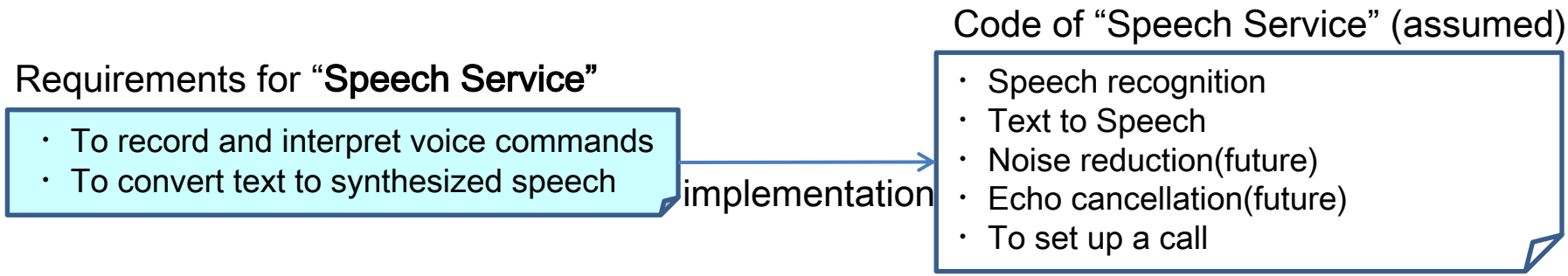
- It is hard to judge what is actually implemented, because requirements do not cover components architecture.

Issue #1 - Other examples

- The granularity of the requirements described in the specification and the granularity of implementations are different.



- Undefined requirements are implemented



Issue #2

- There is no implementation criteria.
  - Unclear scope or importance of implementation
  - Currently coded by developer's own understanding & intention
  - Implementation may not enough for production

Requirements for "Sound Manager"

- Routing Sound Streams
- Mixing Level Control
- Sound Effect
- Reduced Dependency of Hardware

The lack of implementation for "Mixing Level Control"

Sounds are overlapped and outputted

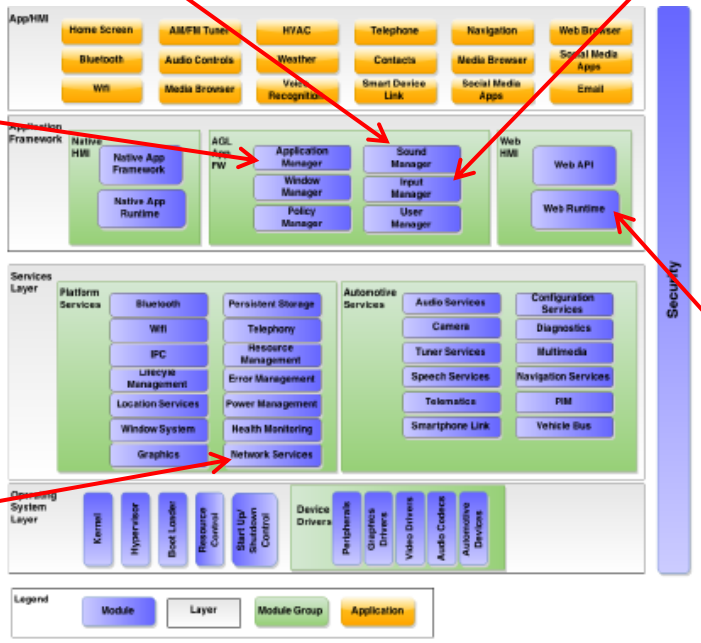
- For application development of real products, AGL UCB implementation is not yet enough.

Issue #2 - Detailed examples

“Sound Manager“ (In the above example)
Lack of implementation of
“Mixing Level Control” feature
⇒ Sound outputs are overlapped

“Input Manager”
Not implemented yet
⇒ Judgment from the operation devices becoming complicated

“Application Manager”
Lack of implementation of
“Get the state of an app”
feature
⇒ difficult to change apps in timely manner



“Network Service”
Need to be conscious of each communication device
⇒ difficult to send and receive messages at the requested timing

“Web Runtime”
Not separated from Applications
⇒ Consume a lot of resource in each application

Proposal for activity

Countermeasure

① Rewrite requirements specification
② Define criteria in the specification

Why countermeasure① ? (For Iuuse#1)

- It is hard to judge what is implemented.

⇒ It is necessary to match what is described in the specification and what is implemented in the code.

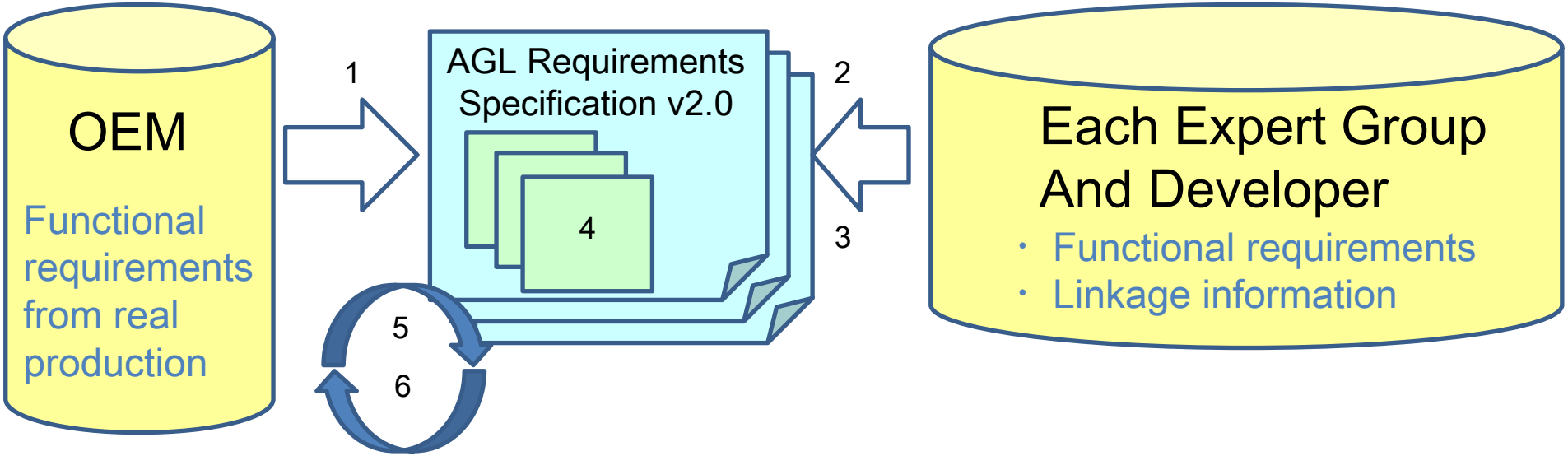
Why countermeasure② ? (For Iuuse#2)

- For application development of real products, AGL UCB implementation is not yet enough.

⇒ It is necessary to have a clear criteria of implementation which development of application can rely on.

Proposal #1 Rewrite specification

Content quality improvement	1 . Collection of functional requirements from OEM 2 . Feedback from developers 3 . Linkage with the source code and the test code
Readability	4 . Unification of the wording & expressions
Maintainability	5 . Maintenance periodically 6 . Release periodically



Content quality improvement

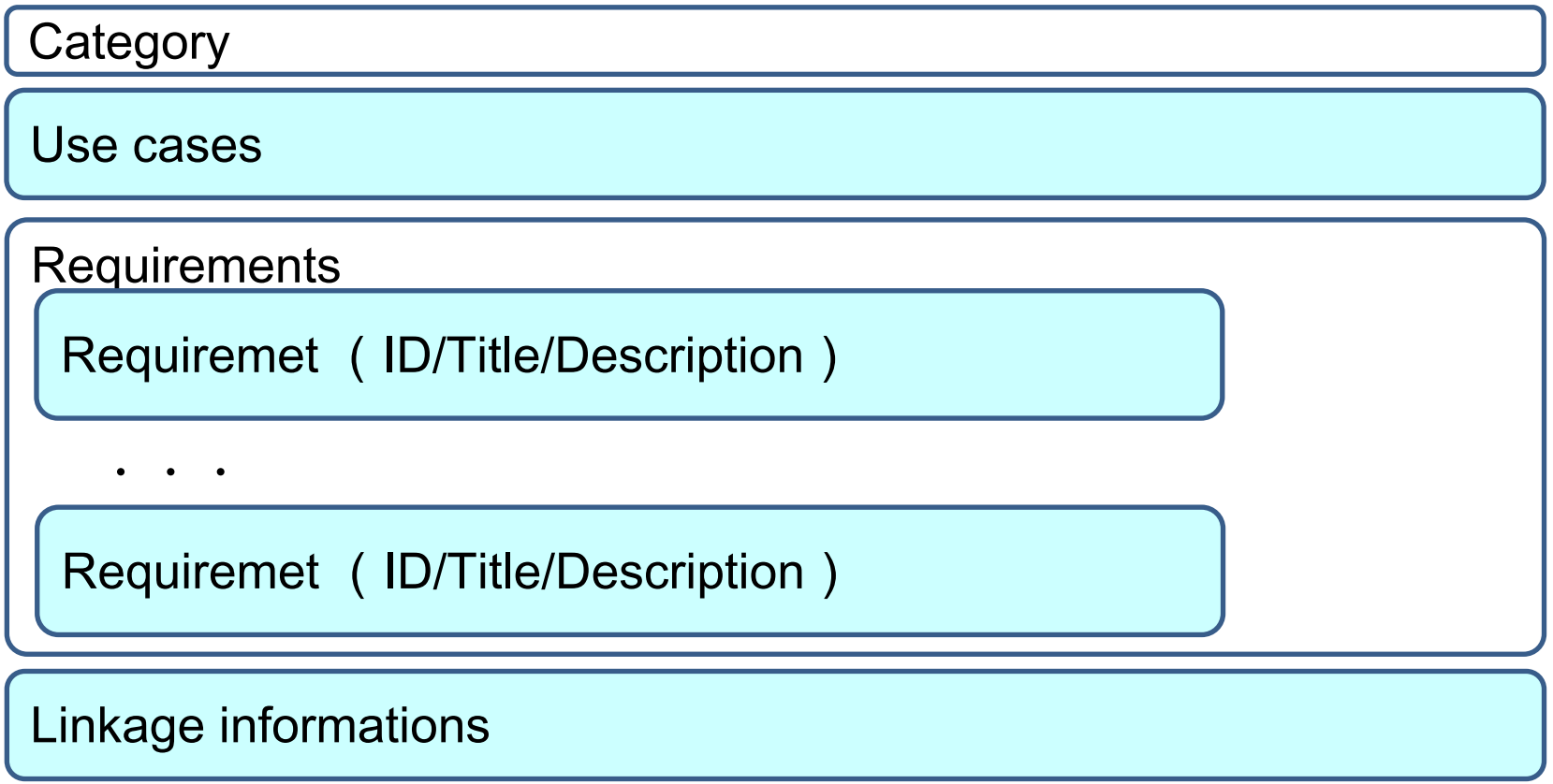
- Improvement point of description based on the collected

Use cases
・ Understandability
Visualize for easy understanding
Requirements
・ Functionally
Review functional requirements again by OEMs and developers
・ Efficiency
Performance requirements in terms of time constraints and resource consumptions
・ Adaptability
Requirements for hardware dependencies
Linkage information
・ Operability
Information that makes it easy to grasp the implementation and test

Readability

- Improvement for unification of wording & expression

Example



Maintainability

- Improvement for the maintenance and release periodically

EG Name

"Requirements Specification Expert Group"(RS-EG)

Charter(proposal)

- The Expert Group is responsible for defining and maintaining requirements as specifications targeted for implementation in the AGL UCB.

- The contents of the requirements specifications should include:

1. Guidelines for deciding design policy for developers
2. Priorities and criteria of developing AGL UCB which OEMs can judge whether their in-vehicle software meet the specifications or not.

Expected Members

It is expected that the team consists of AGL OEM members at least.

Proposal #2 Definition of Criteria

- The order of the implementation for each requirement

A view of the order of the implementation (Example)

- for components
  - Benefits
  - System assumed (e.g. single ECU, multiple ECUs)
  - Target board (e.g. reference hardware model)
  - Target grade of vehicle
  - Type of system in a vehicle
    (e.g. Infotainment, Instrument Cluster, Telematics/Connectivity, etc.)
- for requirements within a component
  - Importance for in-vehicle
  - Difficulty
  - Frequency of usage

  etc.

Proposal #2 Definition of Criteria

- the priority of implementation of components
  - the case of application development (Example)

Component

Priority 1 : Components related to HMI

Native App Framework	Application Manager	Sound Manager	Policy Manager	Web API
Native App Runtime	Window Manager	Input Manager	Speech Services	Web Runtime
Window System	Graphics	Audio Services	Vehicle Bus	

Priority 2 : Basic service for IVI

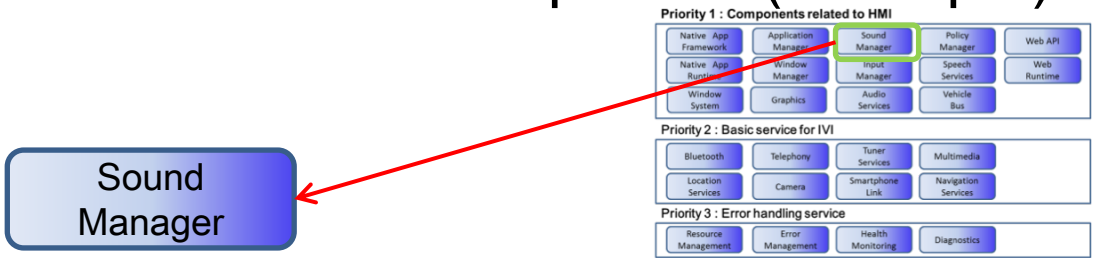
Bluetooth	Telephony	Tuner Services	Multimedia
Location Services	Camera	Smartphone Link	Navigation Services

Priority 3 : Error handling service

Resource Management	Error Management	Health Monitoring	Diagnostics

Proposal #2 Definition of Criteria

- the priority of implementation of features
  - the case within each component (Example)



Priority 1 : High usage frequency

Requirement

- Routing Sound Streams
- Mixing Level Control

Priority 2 : Mid usage frequency

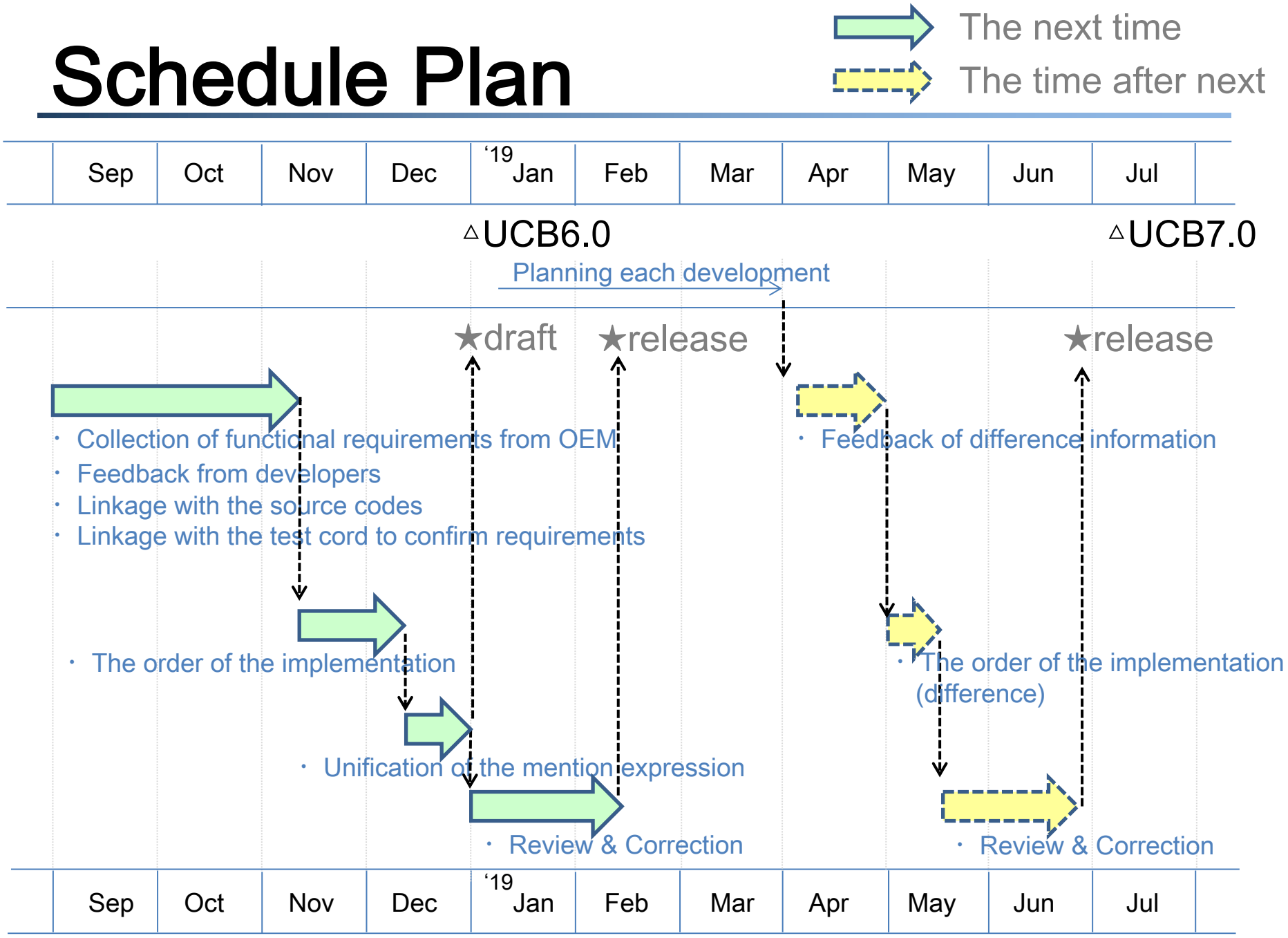
- Sound Effect

Priority 3 : Mid usage frequency

- Reduced Dependency of Hardware

Schedule Plan

Schedule Plan



Conclusion

Conclusion

- ① Rewrite requirements specification

  Evolve and maintain the document as open activity

- ② Define criteria of implementation

  The index of the development promotion by the addition of the priority information

By our proposal, we hope that AGL will become more active and OEM will promote product development using AGL.

Introducing GPL related activity

Introducing GPL related activity

GPLv2's termination clause

- No opportunity to cure
- Automatic termination

→ Only a single mistake makes us never to use the OSS

GPLCC (GPL Cooperation Commitment)

- Introduces a cure opportunity for GPLv2 and LGPLv2
- **On July, TOYOTA joined the GPLCC**
- https://www.toyota.co.jp/jpn/sustainability/governance/compliance/Toyota_GPL_Commitment.pdf

Introducing GPL related activity

Adopt the GPL Cooperation Commitment

1. **No Agreement.** There are no agreements or other documents to sign.

2. **No Fees.** There are no fees or payments required to participate.

3. **Post the Commitment.** Post the GPL Cooperation Commitment on your company's web page.

4. **More Information.** https://gplcc.github.io/gplcc/

Thank you once again for taking the time to join today’s presentation.

toshikazu_ohiwa@mail.toyota.co.jp