"Really" useful solutions in embedded Linux/OSS-based IVI system development

NTT DATA MSE CORPORATION

July 13th, 2016

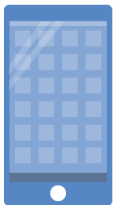
NTT DATA

Who am I ?

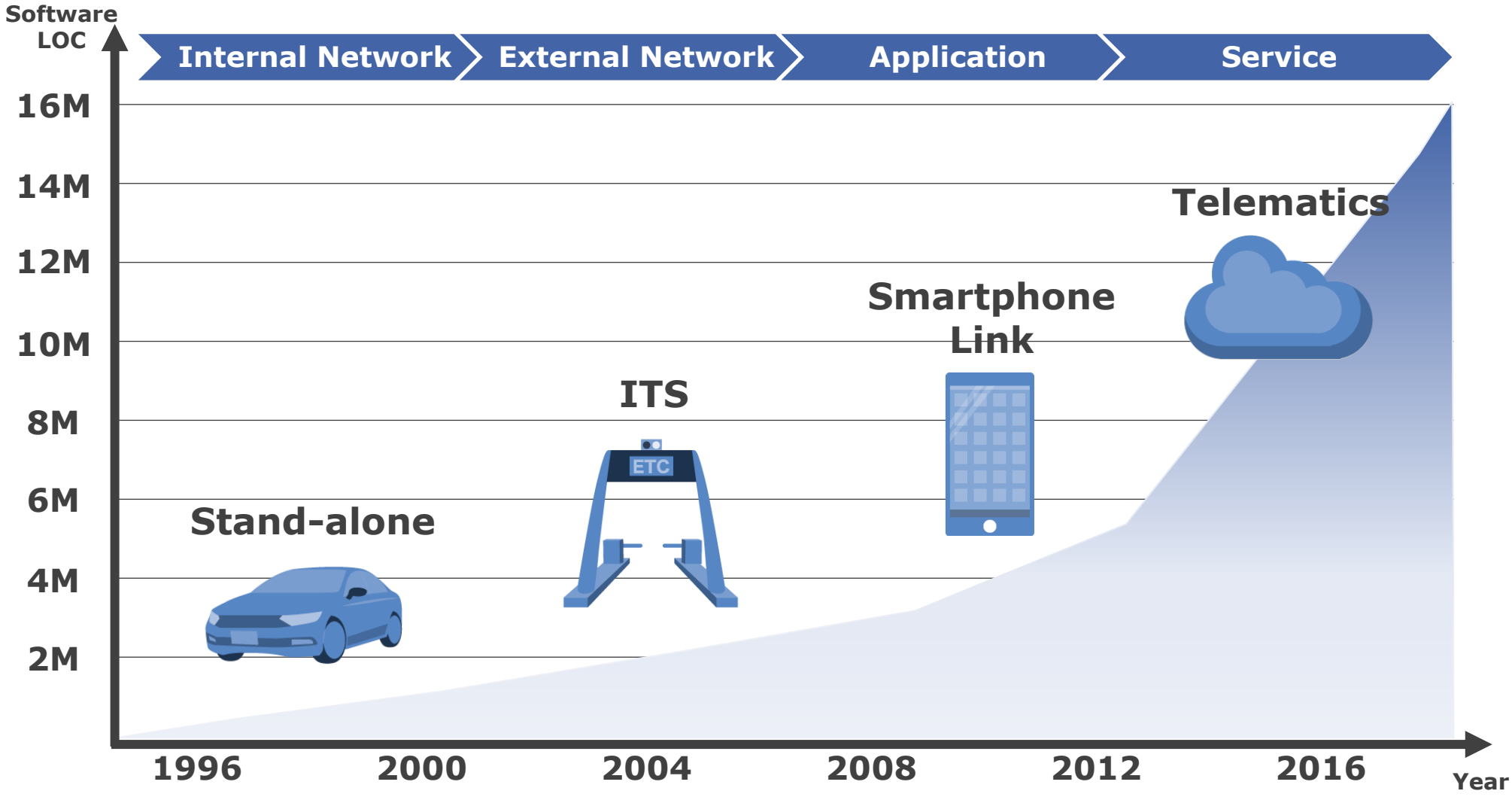
NTT DATA

- **Name**
  Hiroto Imamura

- **Occupation**
  NTT DATA MSE CORPORATION
  - **Platform Strategy Office**
- **Career**
  LINUX system development
    - **Architecture design**
    - **Performance optimization**
    - **Security**
    - **System debugging**



Changes in Automotive field



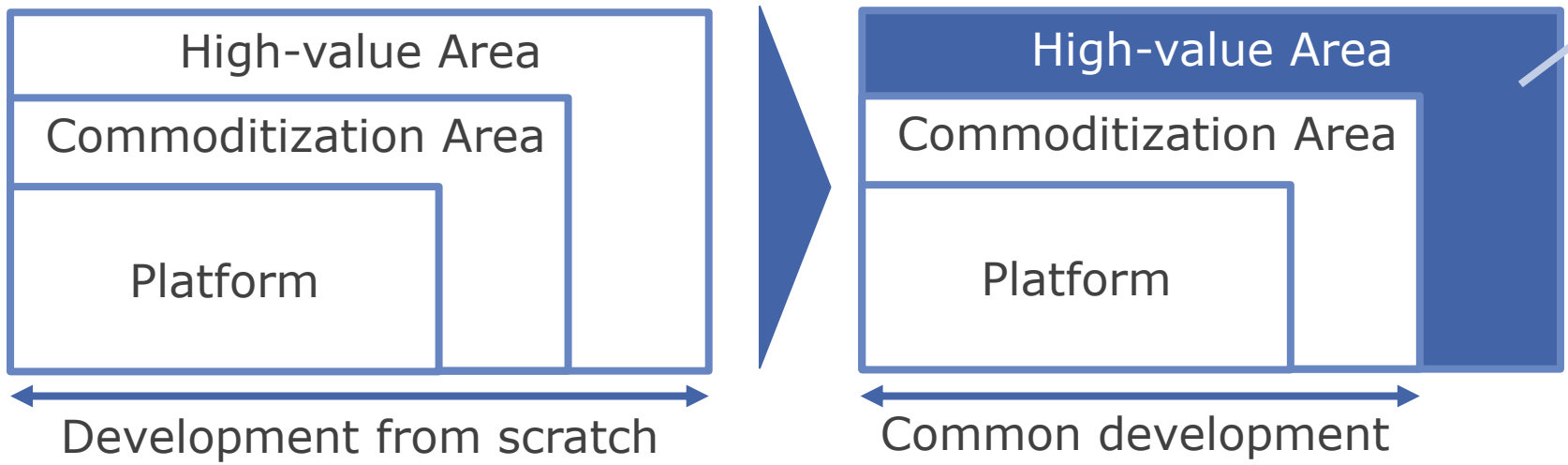
Vehicles become part of IoT devices
Software volume grows explosively to realize services

Necessity of using Linux in automotive field

NTT DATA

Shift to High-value added development

Resources to High-value development



Less cost of commoditization area development

+

Base platform for High-value area (Networks, Graphics,…)

Make good use of Linux

NTT DATA

Tools and Framework

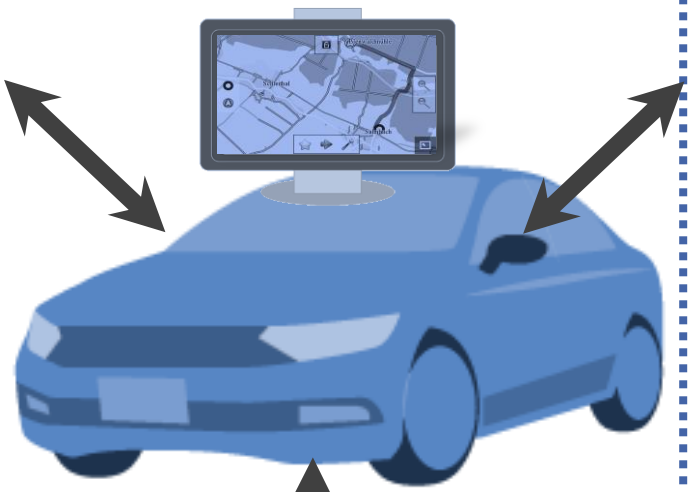
Speed of evolution

Connected vehicle systems

NTT DATA

Telematics Service

- Big data utilization
- Traffic information
- Intelligent Transport System



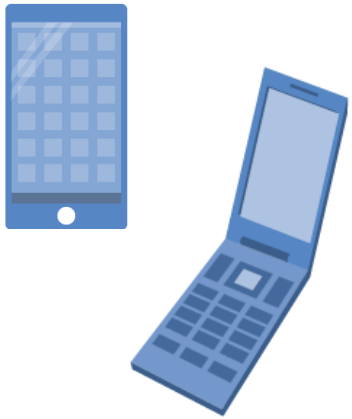
Smartphone Link

- Hands Free
- Navigation
- Multimedia Service

ECU Information

- Engine/Gear/Brake
- Steering wheel/Mirror/Battery

By the way…



Solutions

From RTOS to Linux

Mobile field

Solutions

From RTOS to Linux

Automotive field

Solutions

for Mobile

- Solution A
- Solution B
- Solution C
- Solution D

Solutions

for Automotive

- Solution A Usable
- Solution B Usable
- Solution C’ Not C but C’ Customization required
- Solution D Not usable
- Solution E Automotive-specific

Suggestion to automotive field

Theme	Description
Architectural design	Approach from evaluation viewpoint to solve memory related issues, which are not be solved from design viewpoint.
Efficient debug method	Approach to taking effective logs for analysis when an issue occurs
Boot time optimization	The case example for boot time analysis using “Bootchart”
Security	Method to realize secure system to protect resources using embedded Linux system, “Mandatory Access Control”
Extend the life of a flash memory	Design / Countermeasure to extend a flash memory’s life

Architectural design

NTT DATA

functional design

architectural design

functional evaluation

performance evaluation

memory evaluation

system evaluation

Priority

Priority Design

- Thread division and priority by response time of procedure
- Design evaluation using prototype

Memory

Memory Usage Design

- Memory layout
- Lifecycle of memory block

Dynamic Memory Solutions

- Memory Leak Detection
- Memory usage measurement tool

Quality of architectural design and evaluation leads to product satisfaction.

Memory evaluation

NTT DATA

The major issues of memory handling

#	Type	Problems that can occur	Approach
1	Memory Leak	- Performance decrement - Terminating a process by OOM-killer	- Measurement a memory usage - Tools e.g. Valgrind, memwatch
2	Memory corruption	- Program runaway - Terminating a process by Segmentation Fault	- Tools e.g. Valgrind, memwatch - Console log, debug message - Debugger
3	Invalid memory reference	- Terminating a process by Segmentation Fault	
4	Invalid memory free	- Unintended terminating a process	- Tools e.g. Valgrind, memwatch - Console log, debug message

#2-4 : Difficult to analyze problem without dedicated debug environment in most of the cases
#1 : Possible to analyze problem without dedicated debug environment

Proposal : "Method to measure a memory usage"

Memory evaluation

NTT DATA

#	Method	Measurable memory			
		Kind P:Physical/V:Virtual	System total	By area	By process
1	free or vmstat	P	X approximation		
2	/proc/meminfo	V/P	X detail	X detail	
3	/proc/[pid]/status	V/P			X approximation
4	ps	V			X approximation
5	top	V	X detail		X approximation
6	pmap	P		X detail	X detail
7	/proc/[pid]/smaps	V/P		X detail	X detail
8	/proc/zoneinfo	P		X	
9	/proc/buddyinfo	P		X	

Possible to know rough memory usage using meminfo and pmap

Memory evaluation

NTT DATA

Memory leak analysis case using pmap

- Trace a memory usage of each area
- Identify specific point of increased memory
- Investigate the cause

```
----------
725: {no such process} malloc_test 64
Address   Kbytes     PSS   Dirty    Swap  Mode  Mapping
00008000       4       4       0       0  r-xp  /malloc_test
00010000       4       4       4       0  rw-p  /malloc_test
00011000     388      28      28       0  rw-p  [heap]
.....
--------  ------  ------  ------  ------
total        1972      93      76       0
```

[heap] area is increasing...

```
----------
725: {no such process} malloc_test 64
Address   Kbytes     PSS   Dirty    Swap  Mode  Mapping
00008000       4       4       0       0  r-xp  /malloc_test
00010000       4       4       4       0  rw-p  /malloc_test
00011000     580      32      32       0  rw-p  [heap]
.....
--------  ------  ------  ------  ------
total        2164      97      80       0
```

Doubt "malloc()" size is 128KB or under

```
----------
892: {no such process} malloc_test 256
Address   Kbytes     PSS   Dirty    Swap  Mode  Mapping
00008000       4       4       0       0  r-xp  /malloc_test
00010000       4       4       4       0  rw-p  /malloc_test
b6da9000     520       8       8       0  rw-p    [ anon ]
.....
--------  ------  ------  ------  ------
total        2104      73      56       0
```

[anon] area is increasing...

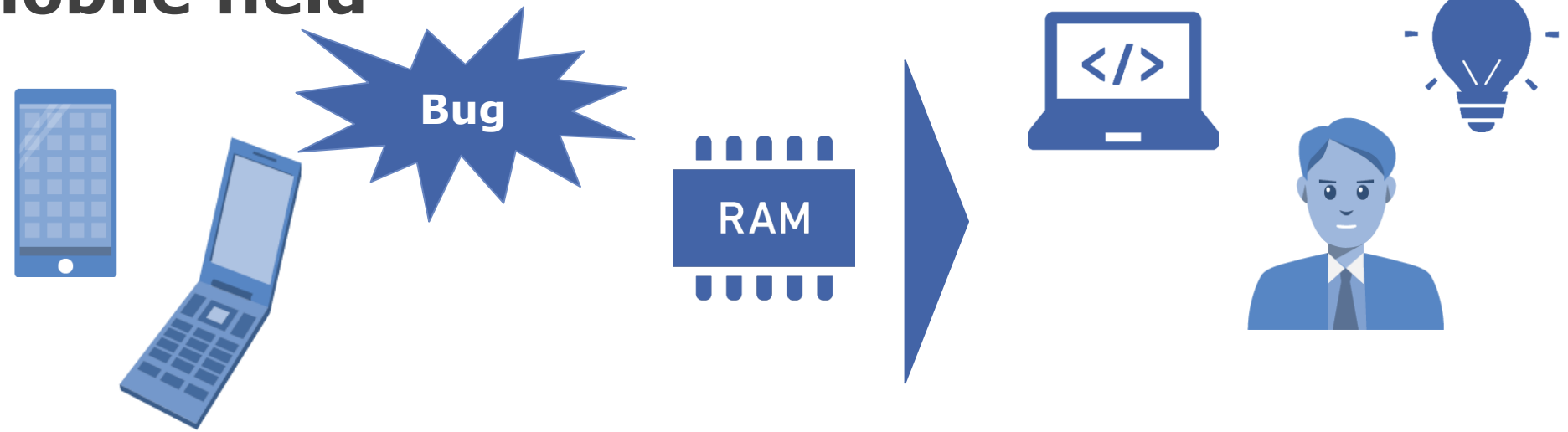
```
----------
725: {no such process} malloc_test 64
Address   Kbytes     PSS   Dirty    Swap  Mode  Mapping
00008000       4       4       0       0  r-xp  /malloc_test
00010000       4       4       4       0  rw-p  /malloc_test
b6d68000     780      12      12       0  rw-p    [ anon ]
.....
--------  ------  ------  ------  ------
total        2164      97      80       0
```

Doubt "malloc()" size is over 128KB

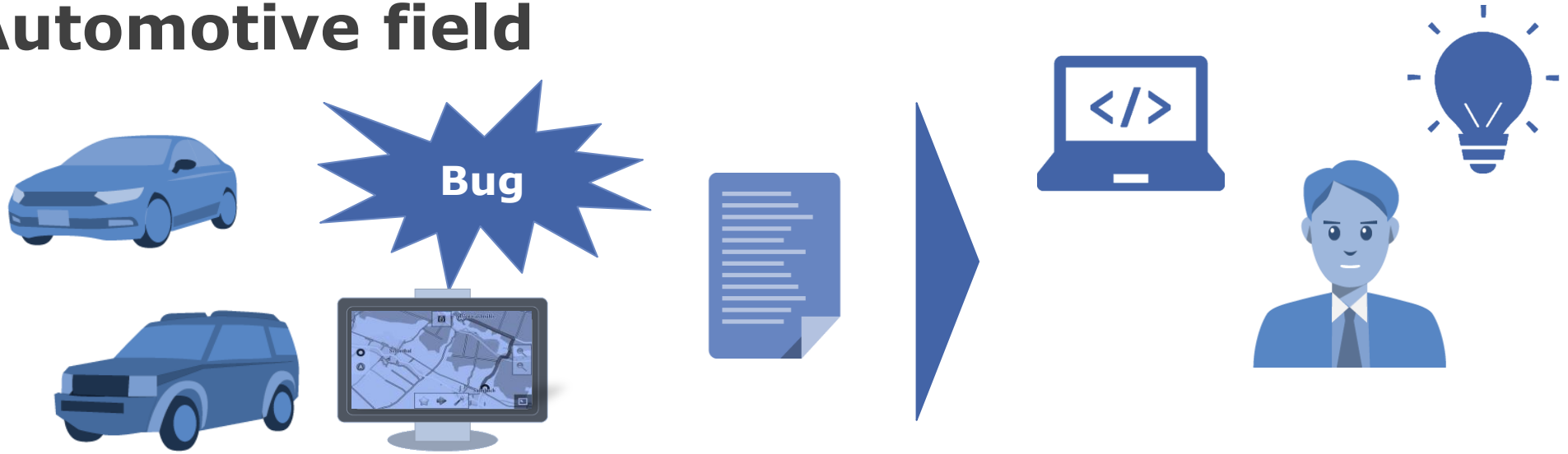
Efficient debugging method

NTT DATA

Mobile field



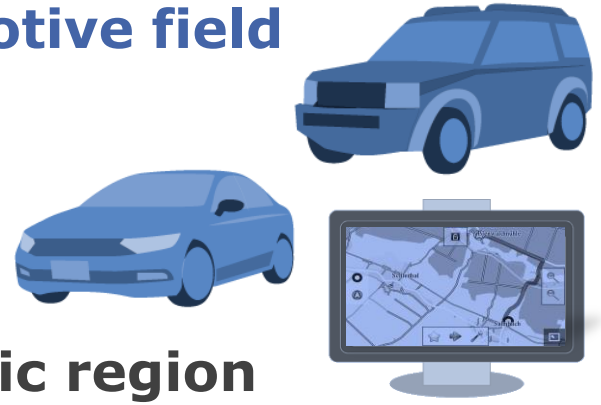
Automotive field



Logging design

NTT DATA

The reason of hard to reproduce bugs in automotive field



- Bug depends on **a timing**
- Bug depends on **a place**
- Bug depends on **an environment**

  e.g.
  - A bug of a music player occurs only in **a specific region**
  - A bug of a navigation occurs only in **a real vehicle environment**

It is important to design a logging system

Point

- Taking logs should be easy
- Generated logs should be accurate
- Generated logs should have enough information

  e.g.
  - Generation environment : peripheral device, external signal
  - Procedure for reproducing : user operation

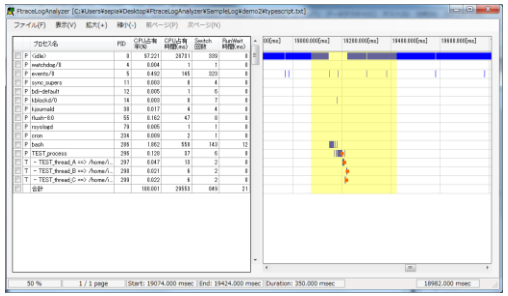
Record enough information when bug occurs at the first time

Boot time optimization

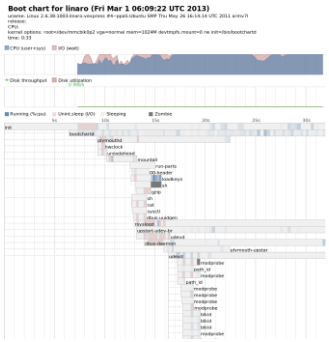
High load

Resource Conflict

Out of memory



Ftrace



Bootchart

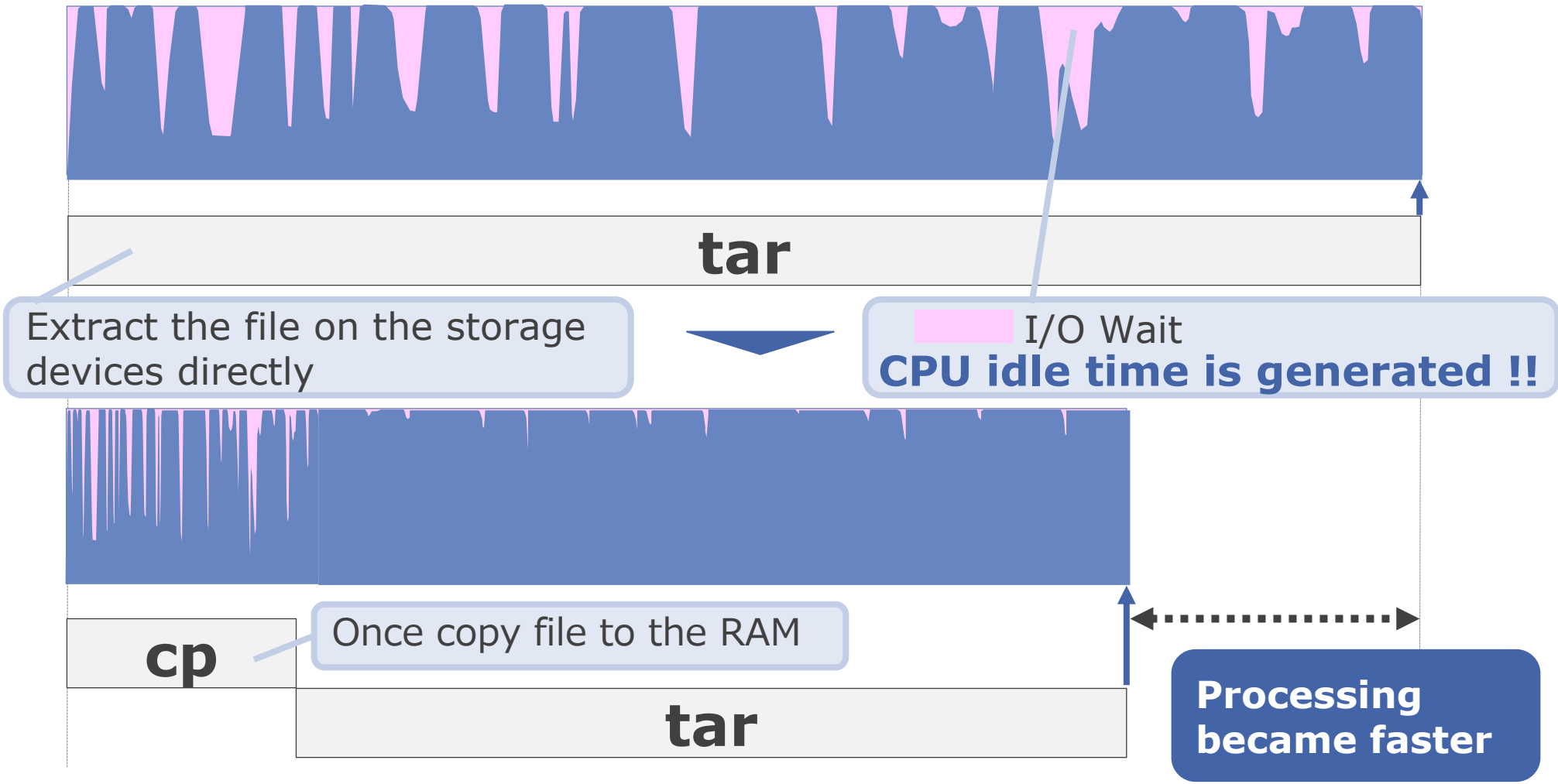
System profiling

System tuning

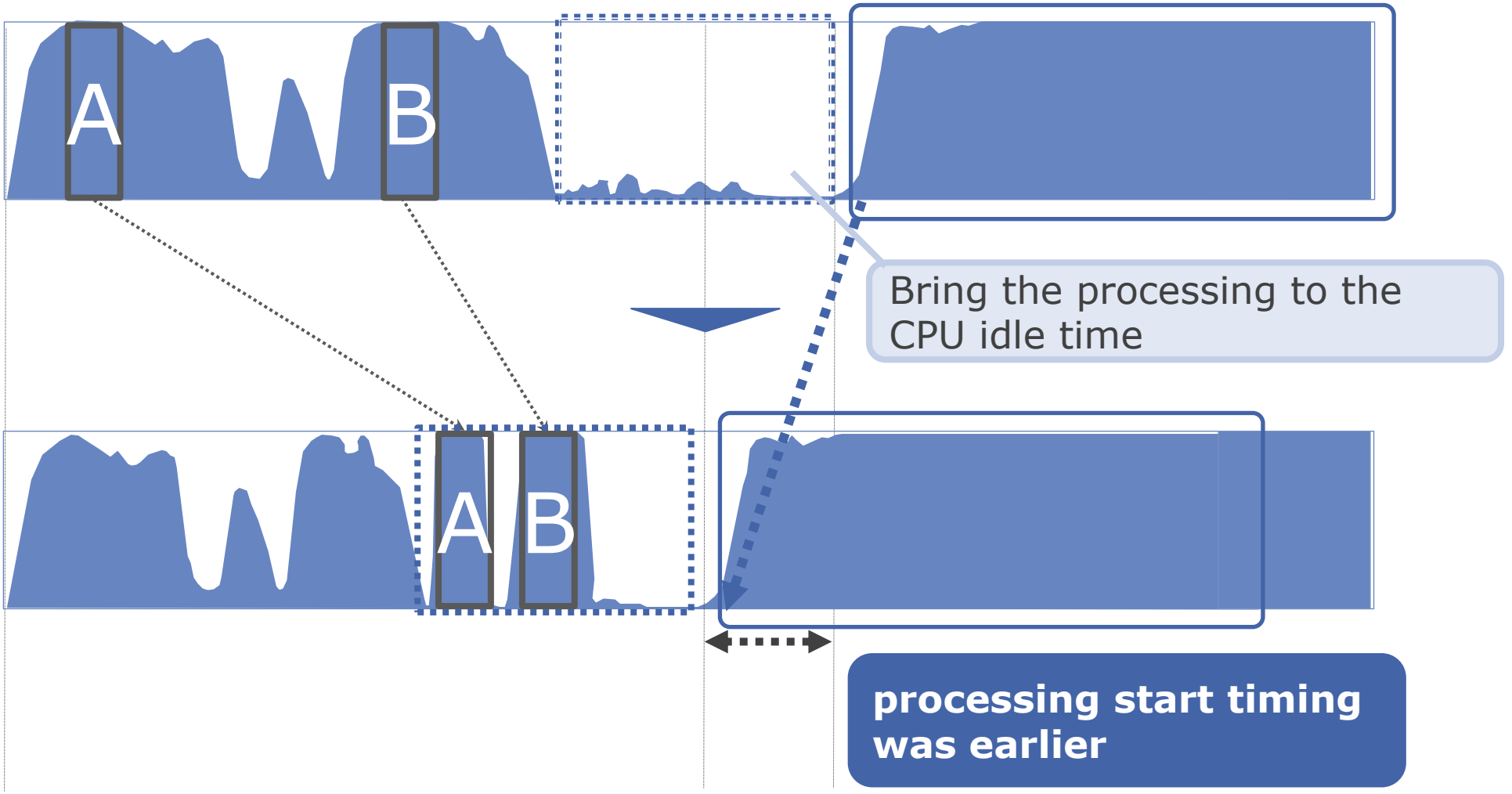
Evaluation

Method for specifying the bottle-neck using Bootchart

Case 1 : Reducing I/O Wait Times



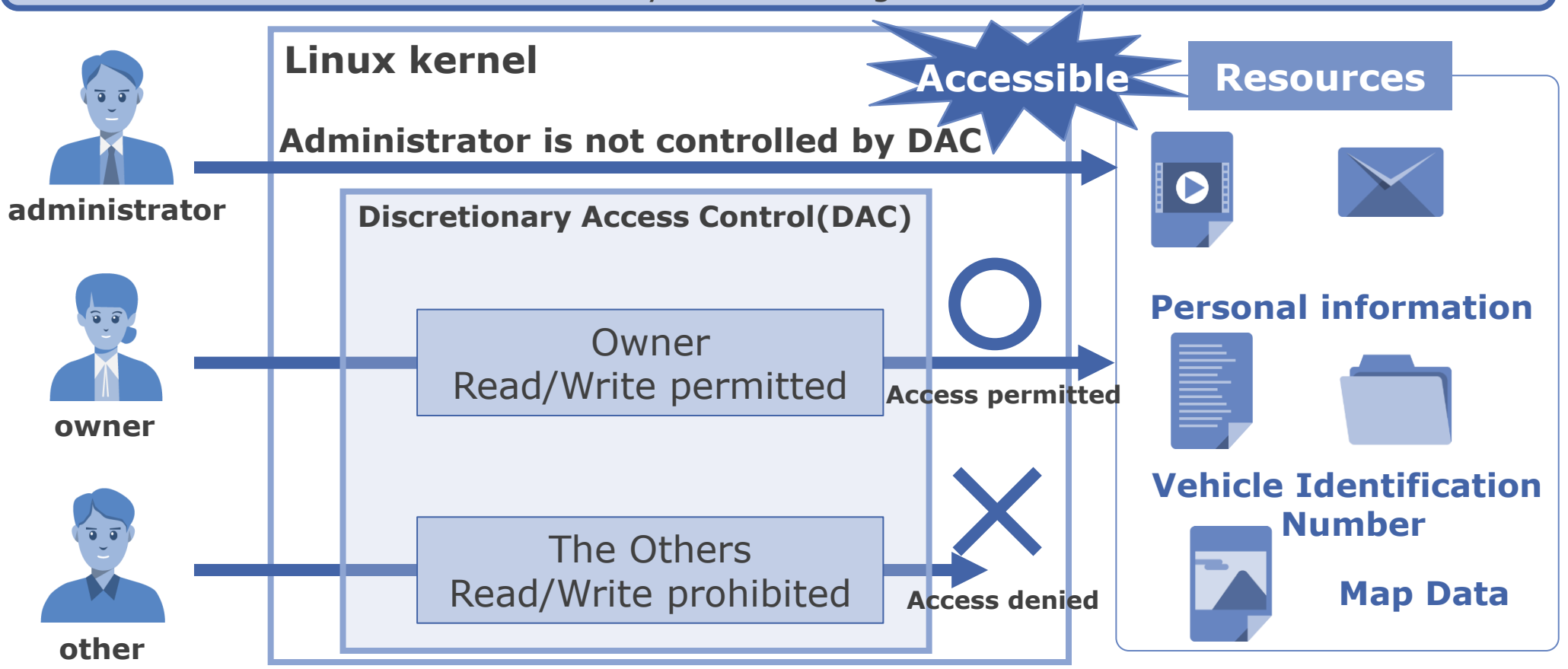
Case 2 : changing the processing order



Security

Security control by normal access control of Linux (Discretionary Access Control)

- Owner set Read/Write/Execute permission to files
- Owners' access control is not applied to administrator
- Administrator can access to any resource to guard

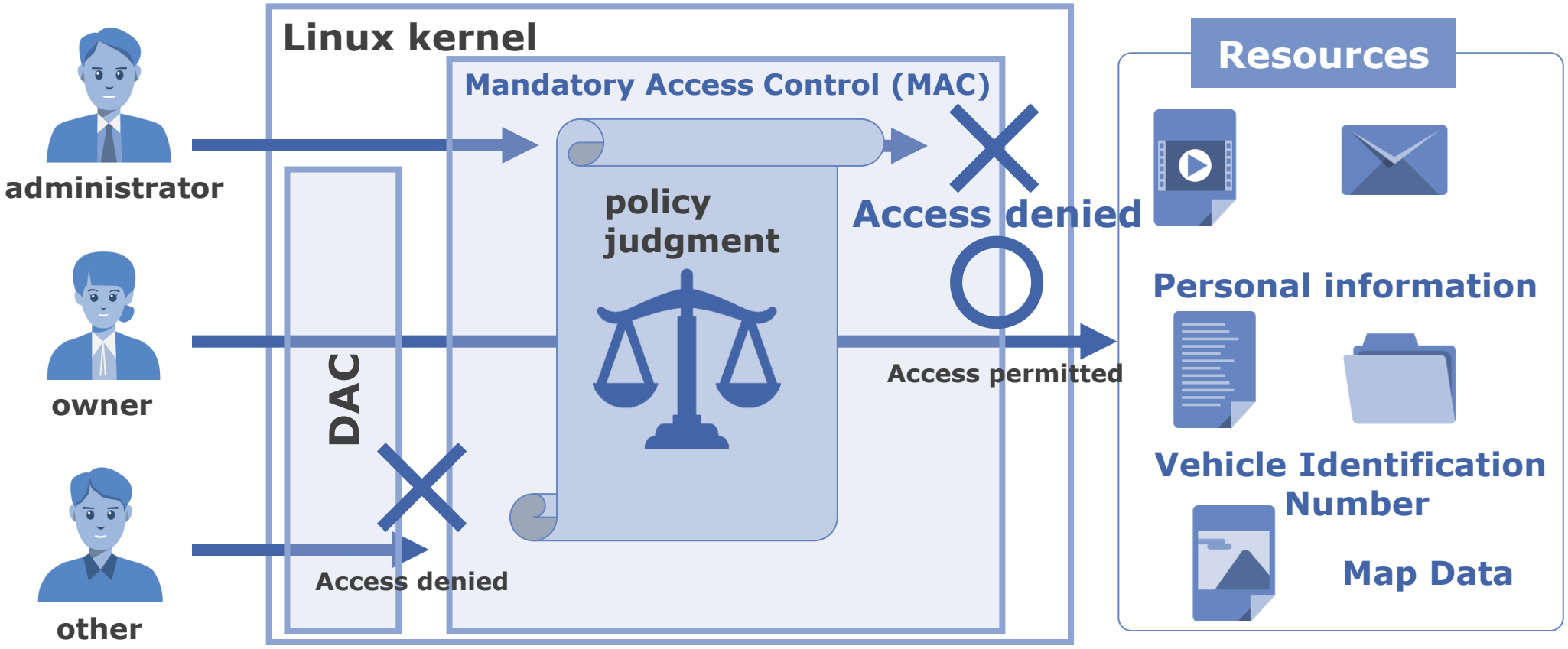


Importance of security is growing as automotive is connected to open network

Security

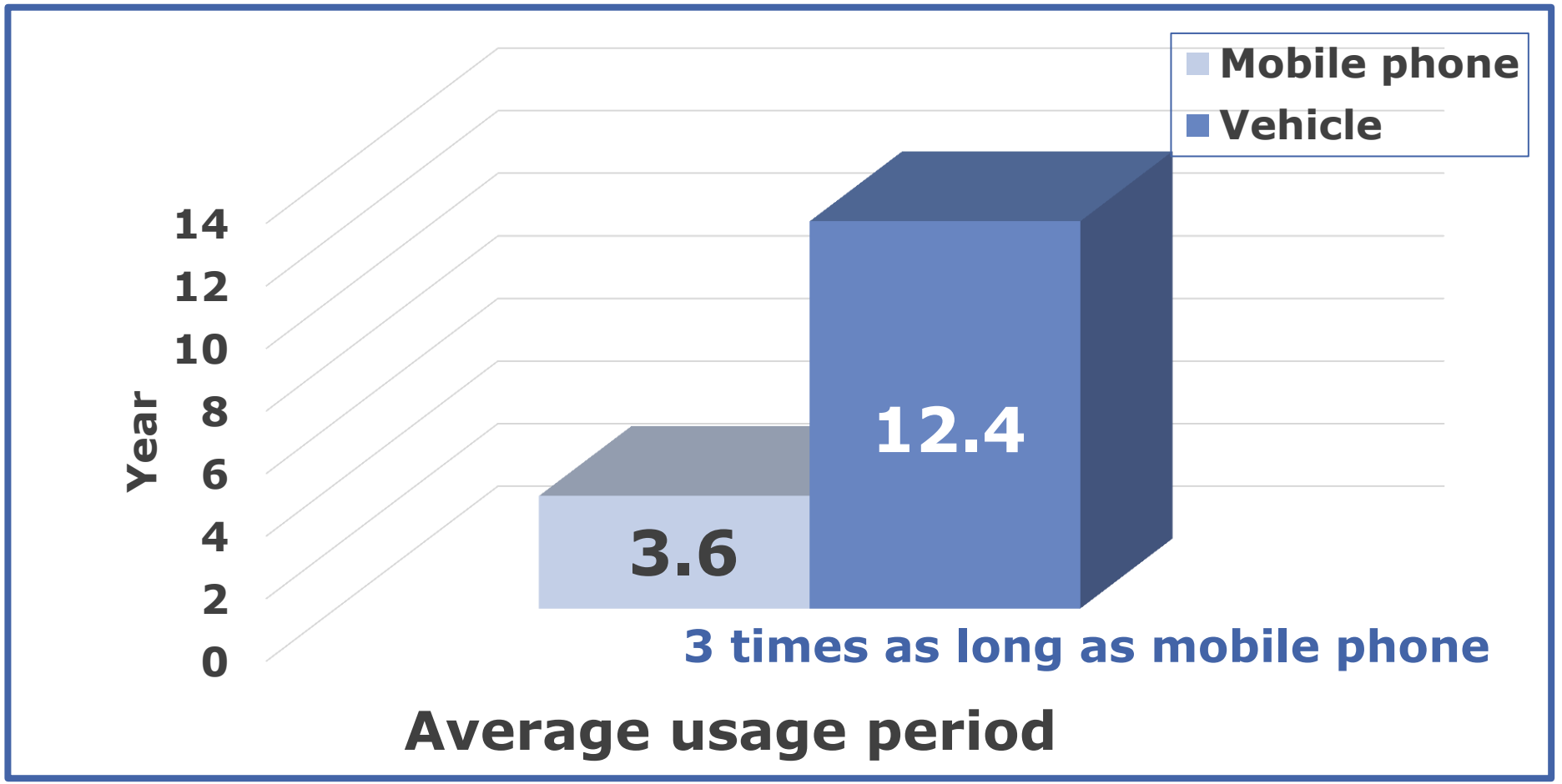
Security control by Mandatory Access Control

- Define access policy
- Access policy is judged at each access request
- Administrator's access is under control of MAC



MAC is effective for security control of resources

Extension the life of a flash memory



Design and measures considering long-life of a flash memory

Extension the life of a flash memory

NTT DATA

Investigation use cases

Investigation Characteristics
Flash memory, File system

Simulating the writing quantity of data each use cases

Flash memory selection

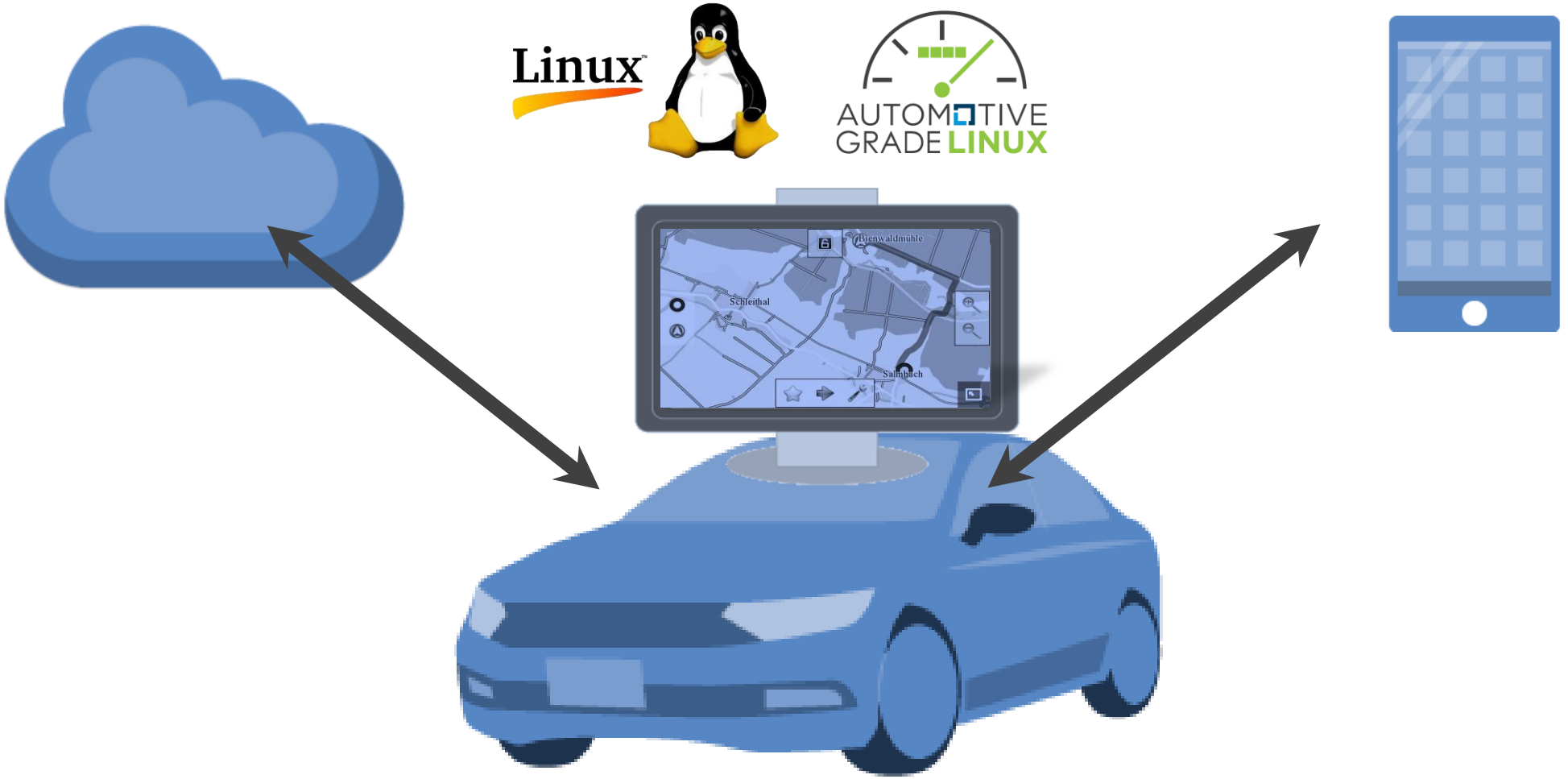
Reducing the writing quantity of data
- **Optimization of the writing**
- **Omitting useless writing**

Flash memory determination

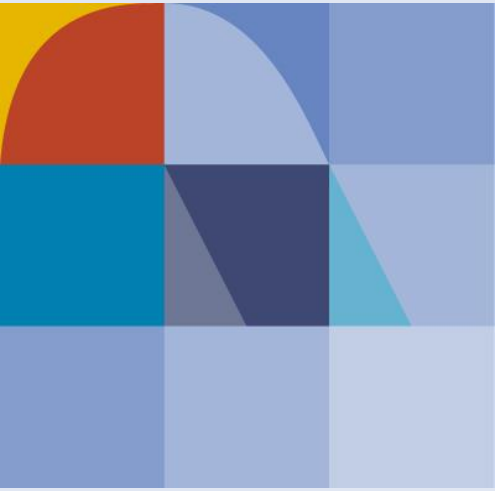
Evaluation the durability of a flash memory
–Writing test –Heat test –Stress test

Accurate simulation of use cases and Optimization of the number of writing times

Conclusion



Linux is suitable platform for creating service



NTT DATA

Global IT Innovator