Automotive embedded system redesign

Jean-Baptiste Laurent

August 28, 2020

Presentation plan

- Introduction and abstract

- Research content and impact

- Administrative details, and research timeline

Section 1

Introduction and abstract

Presenting myself first

Jean-Baptiste Laurent

- 1 year teaching
- 3 years in research, static analysis
- 5 years in cyber-security
- 10 years in software development

Current status

- LR-Technology consultant, working for Faurecia
  - First, 6 months on the RAPIDE platform [a]
  - Now, exclusively working on the following research topic

[a] Generic software platform for car embedded systems

Involved Parties

RICHEFEU Julien

- Faurecia Clarion Electonics, Platform Software Manager
- PhD Director

MENSUEZ Bruno

- ENSTA Paris-Tech, Teacher
- PhD Co-Director

Objective of the presentation

- **Presenting**
  - The **new project** content
  - Who is involved
  - The work I will brought in
- **Precising**
  - How this project could **impact** you in the long run
- Get the time to **exchange** on this

Definitions and wording

EN -> FR

- **ECU**: Unité de controle
- **Socket suppliers**: Fournisseur de micro processeur
- **OEMs**: Les assembleurs
- **Car maker**: Le fabricant
- **The model**: L'ensemble de l'architecture, du microprocesseur à l'application haut niveau.

Section 2

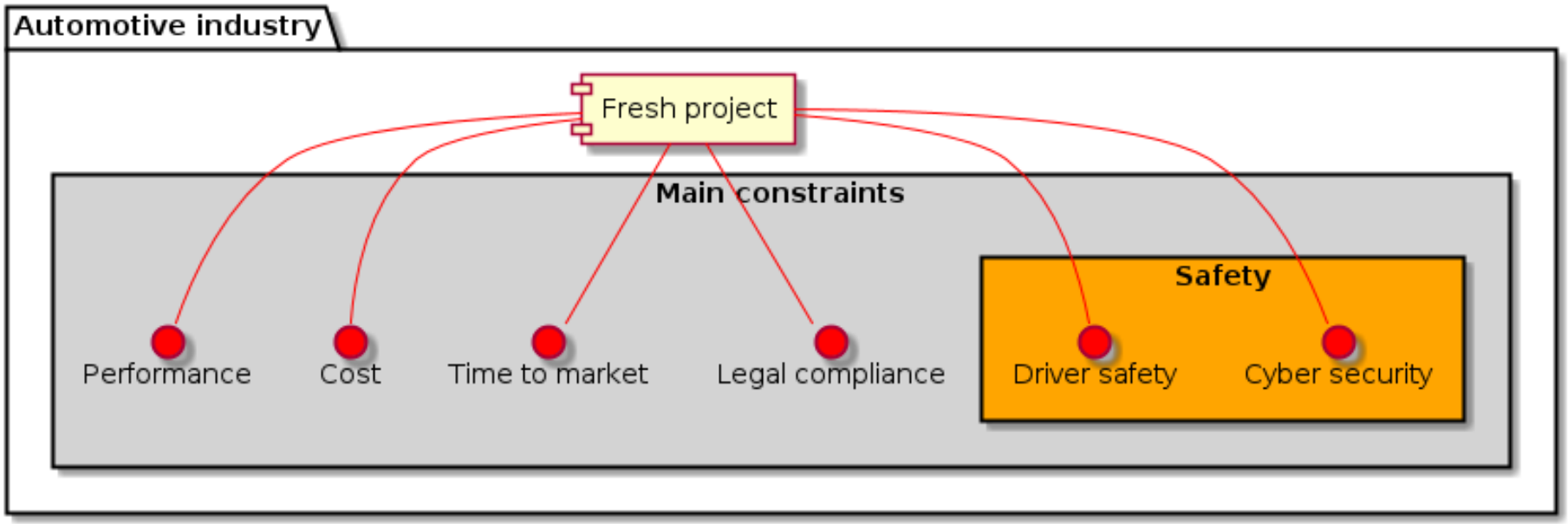
Research content and impact

Subsection 1

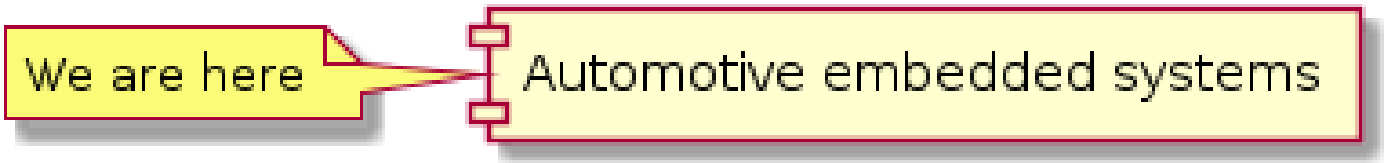
Presentation of the current model

Reminder of the main constraints

- We will get back to them next
  - Mostly when comparing **pros** and **cons**
  - And also why this project

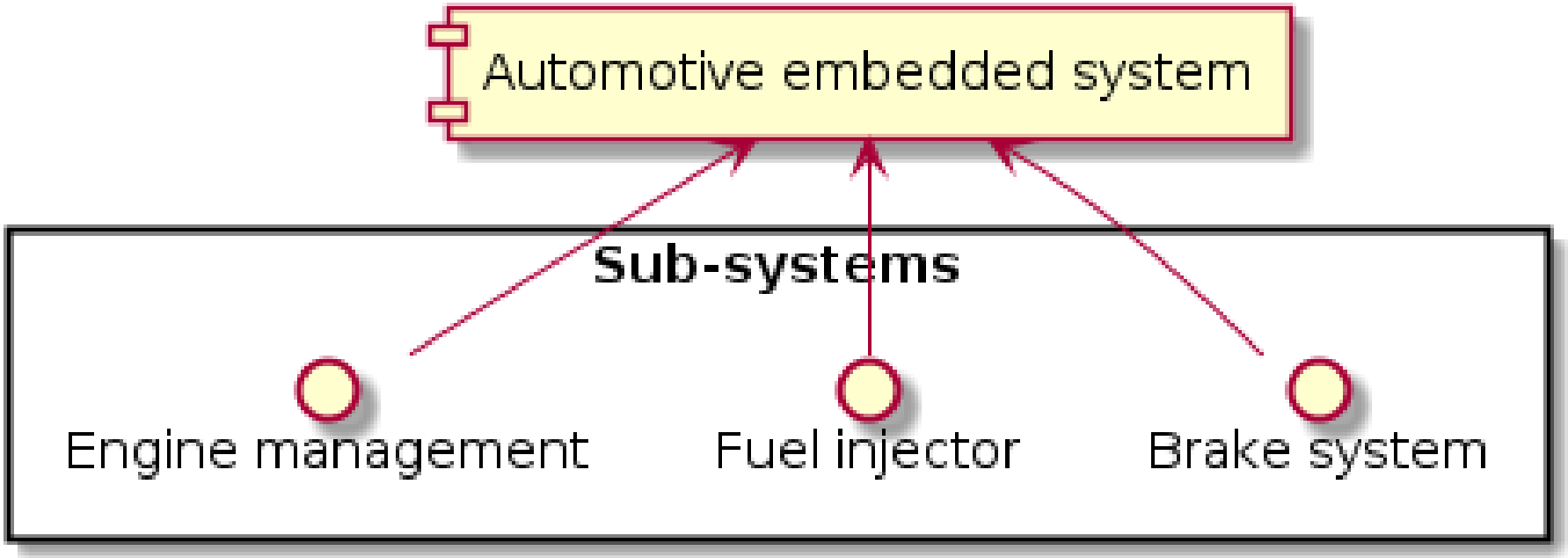


Model explanation, let's start small

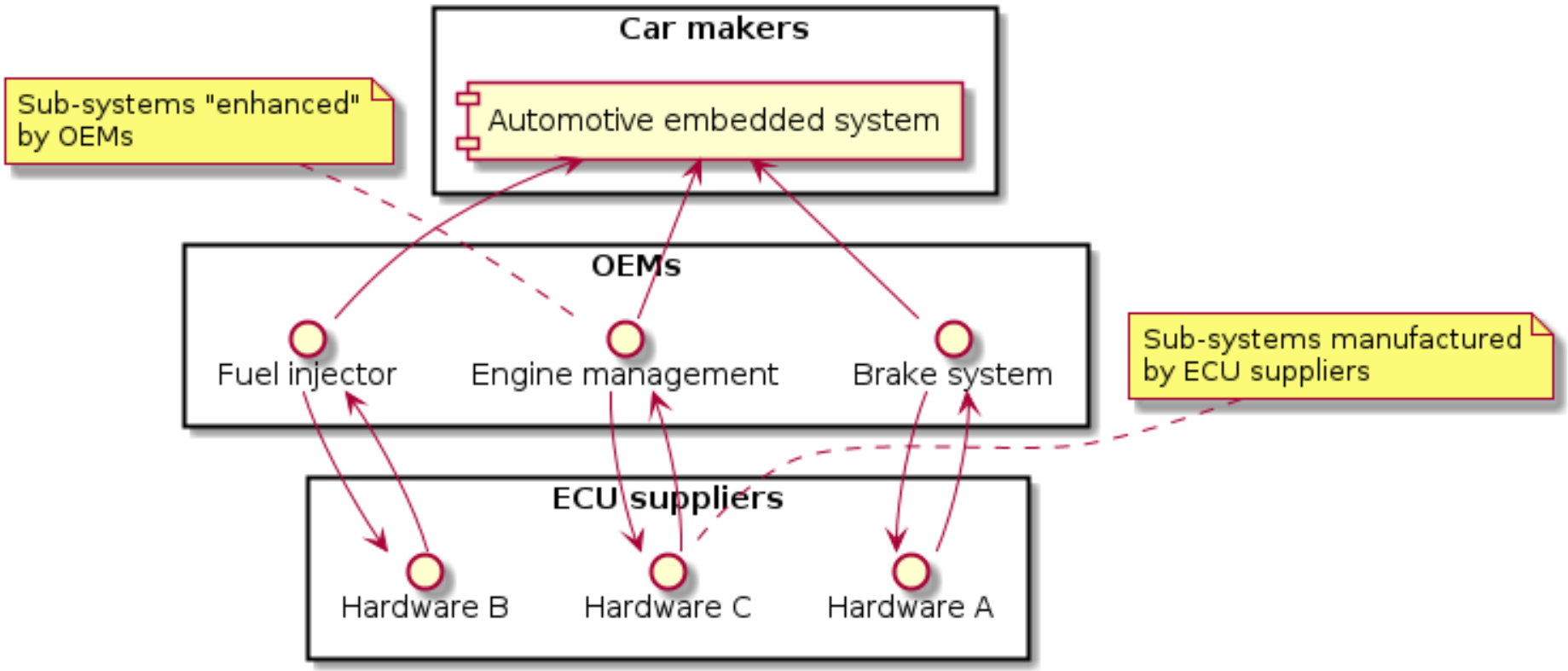


A brief, brief model representation

- **The model** is composed of
  - An **orchestrating component**
  - Individual and isolated **sub-systems**



External parties

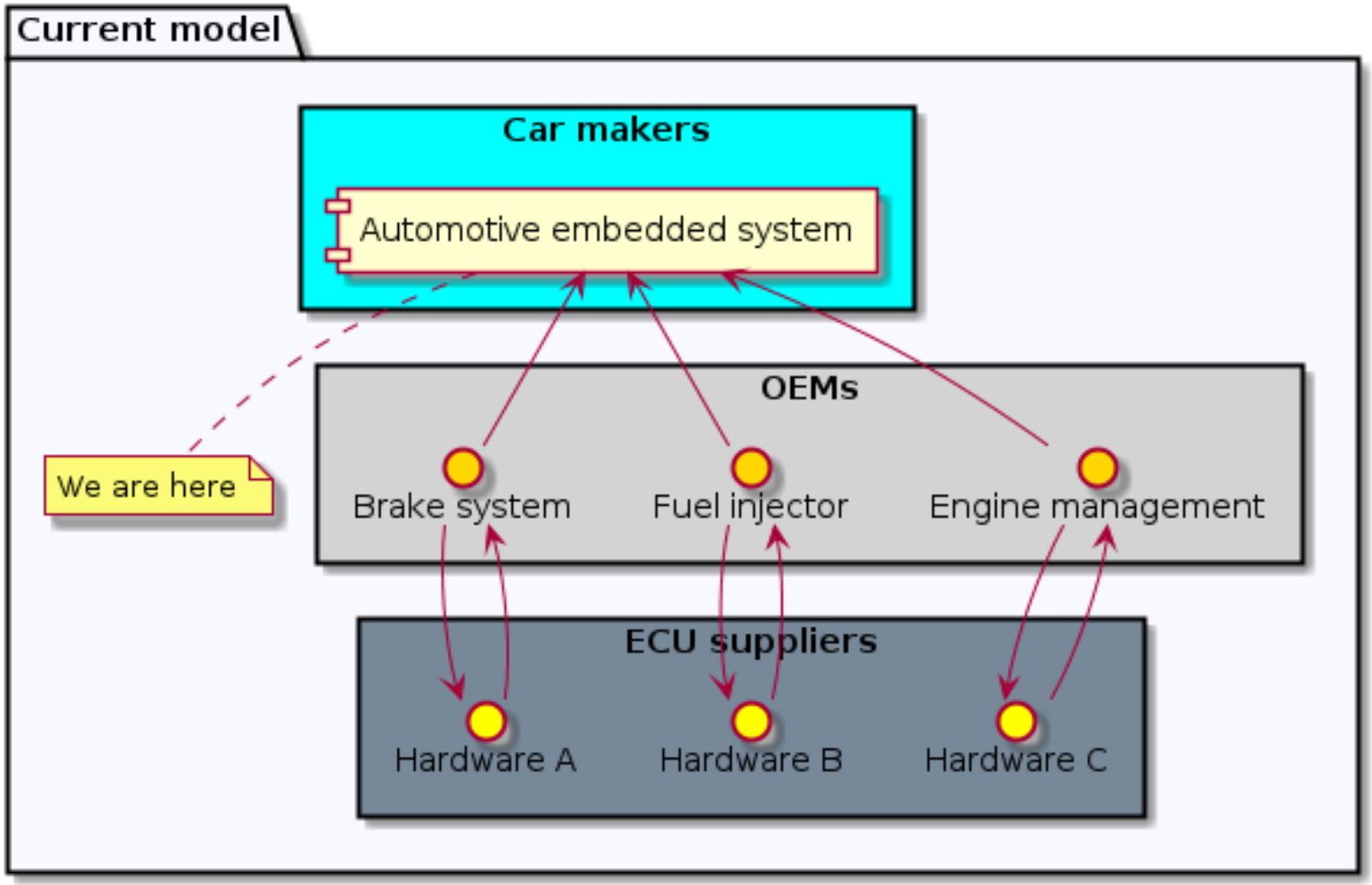


Subsection 2

Current model growing issues

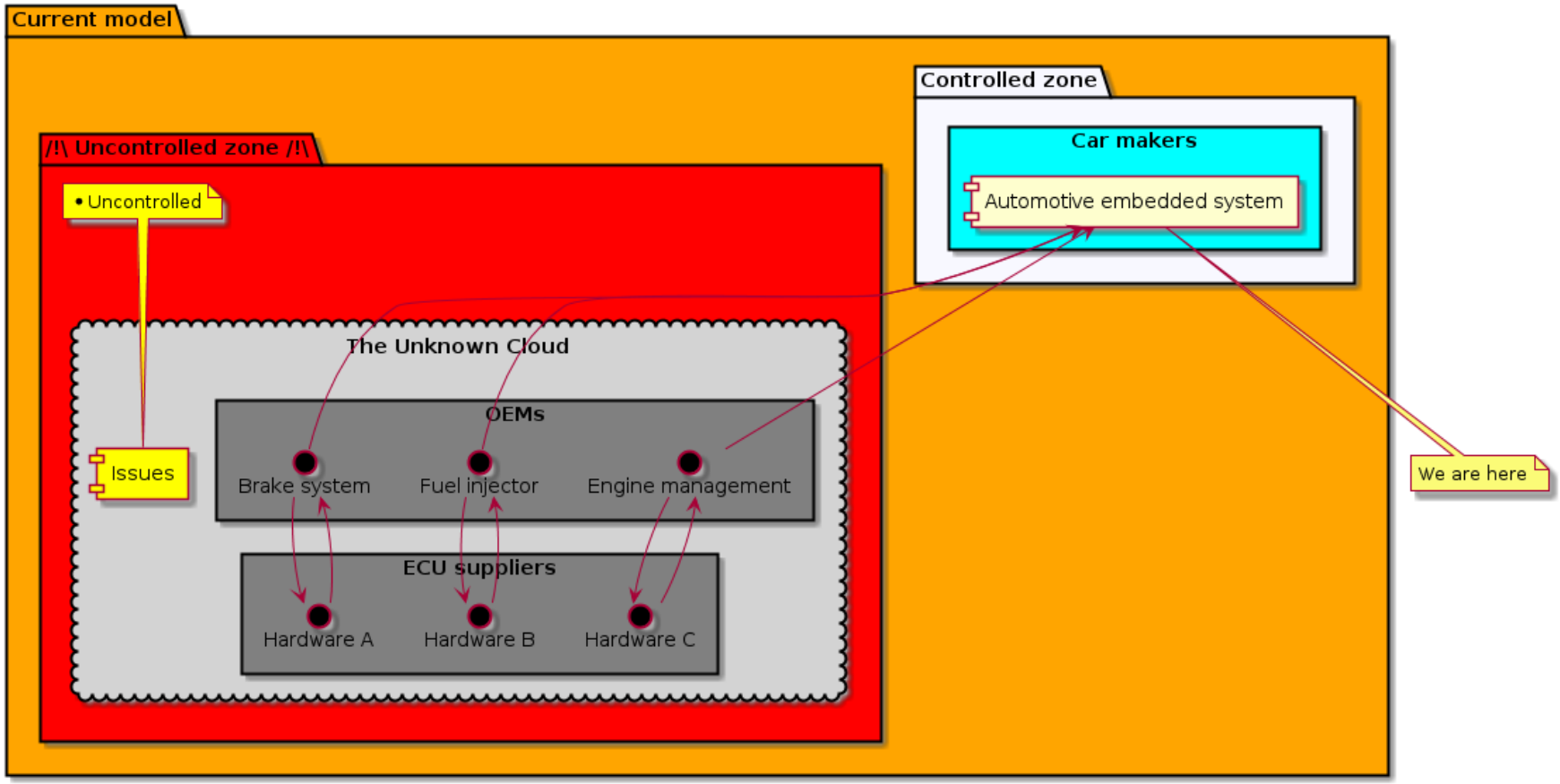
Afterthought

- Do we have any **issues** with that model ?



First issues

- **Third parties** involved ?
  - **Proprietary** code
  - Uncontrolled **TTM** and **quality** standard



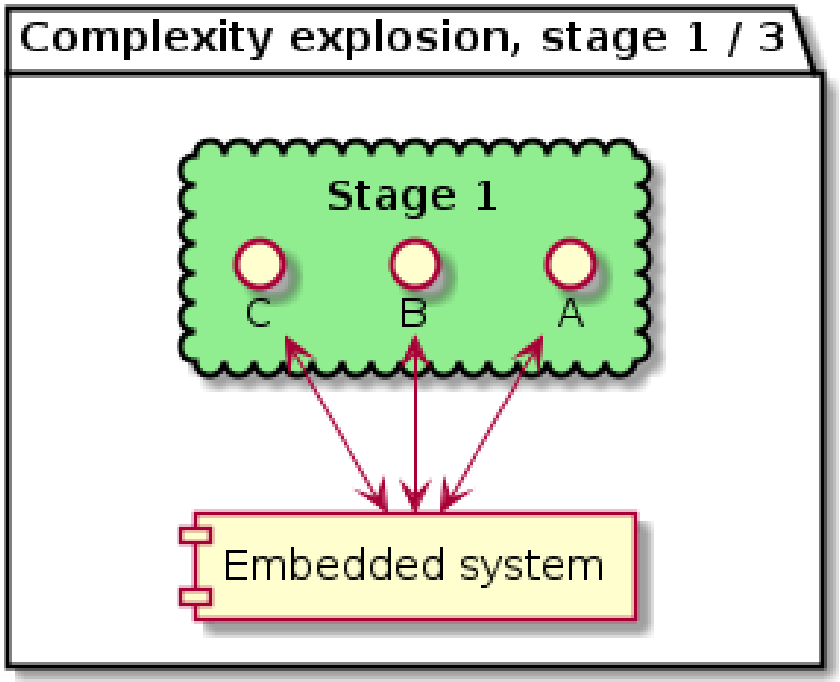
Cars become more and more complex

Let's focus on that for a minute, and see why it's really a problem within the current model

Current model, a complexity issue

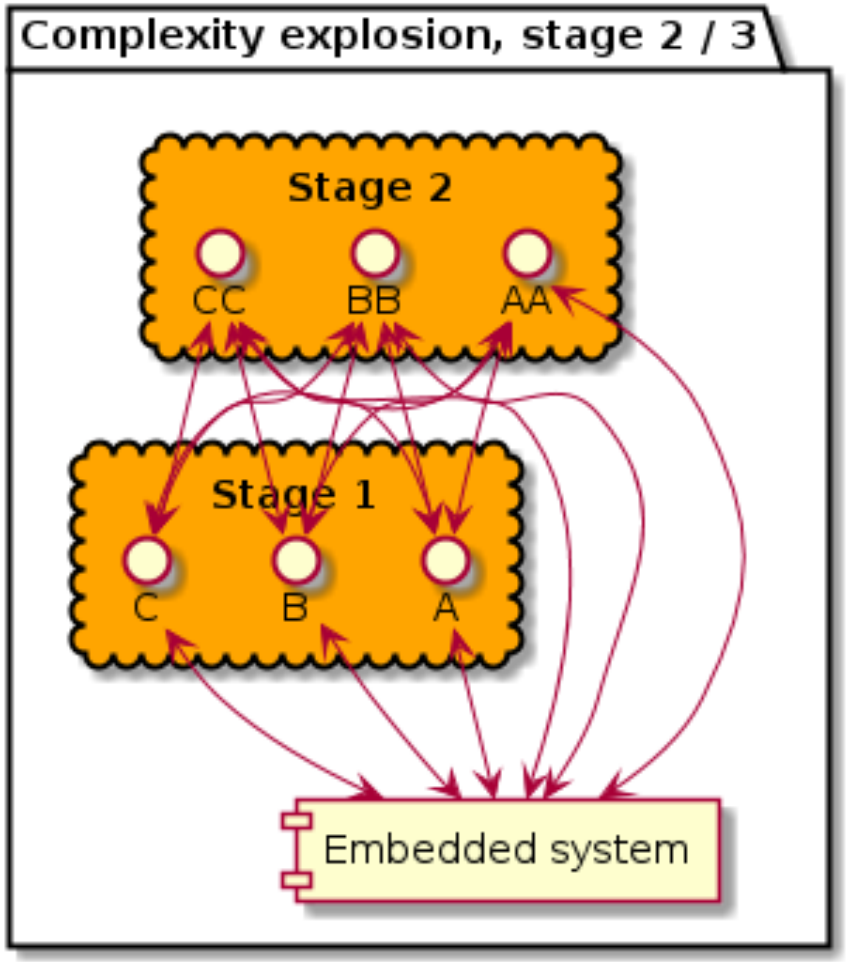
Exponential communication complexity, stage 1

- Flat sub-system architecture
  - Good so far



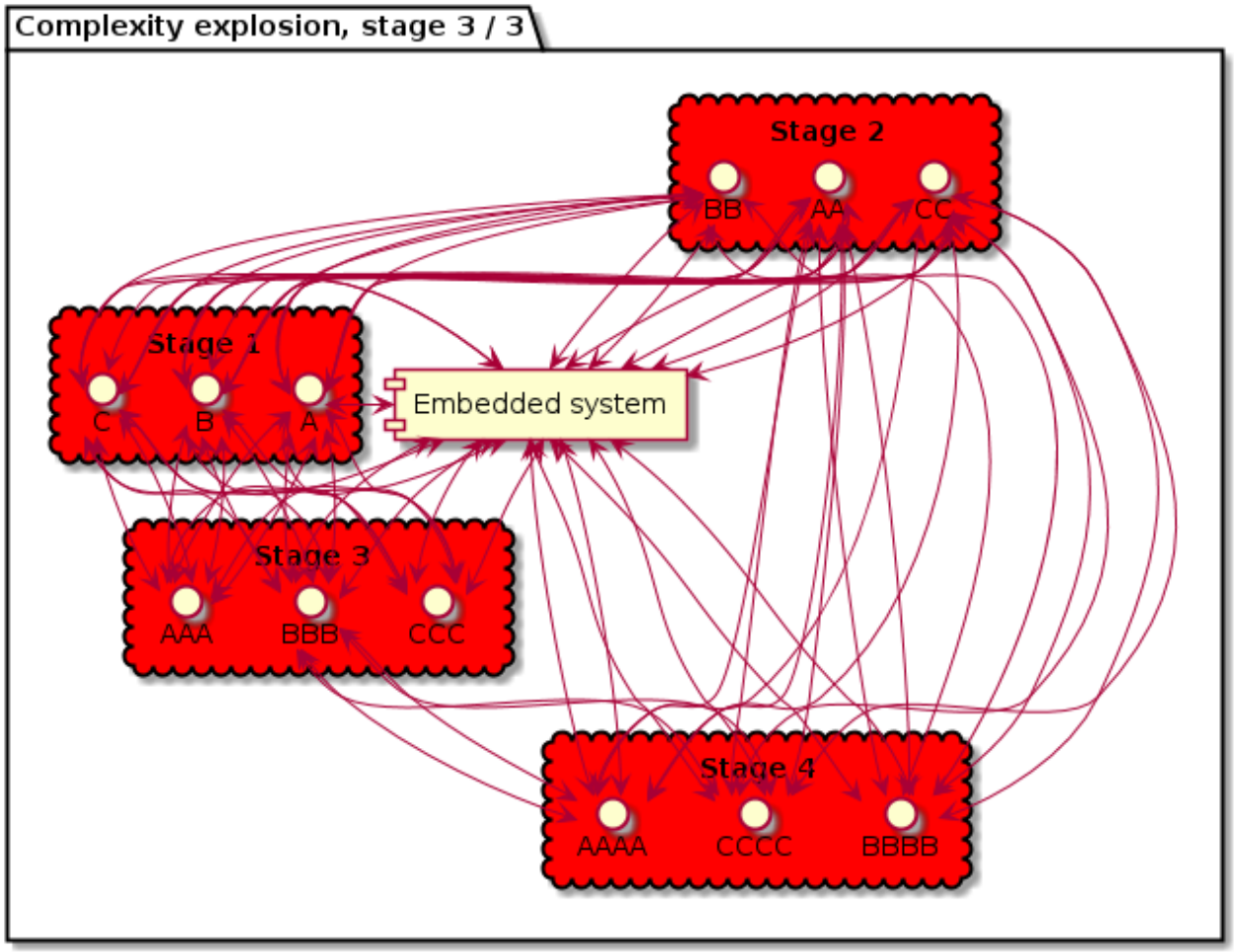
Exponential communication complexity, stage 2

- Flat sub-system architecture
  - Things start to get messy



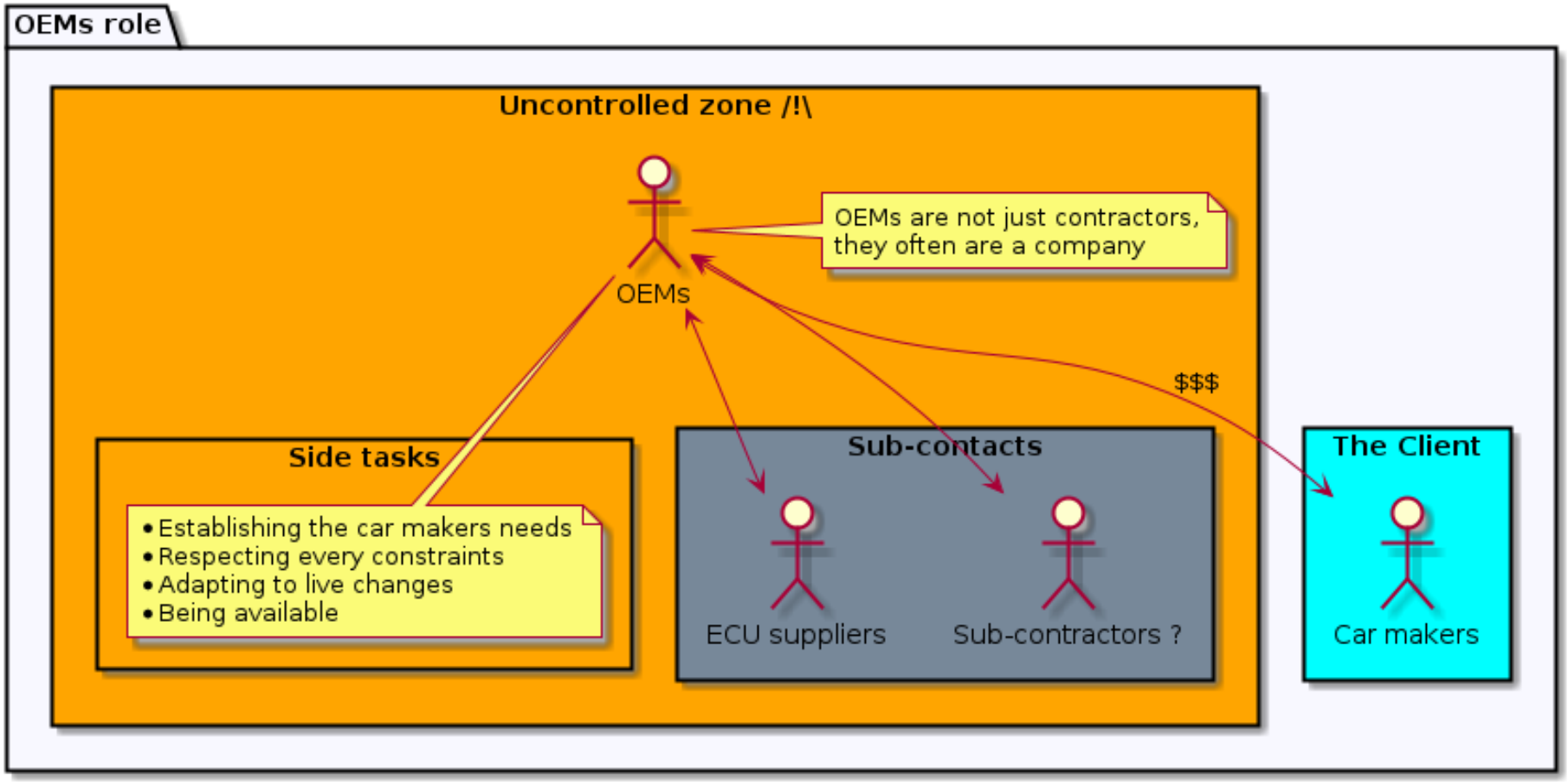
Exponential communication complexity, stage 3

- Flat sub-system architecture
  - Game over



Small parenthesis on OEMs

Car makers <—— OEMs <—— ECU suppliers

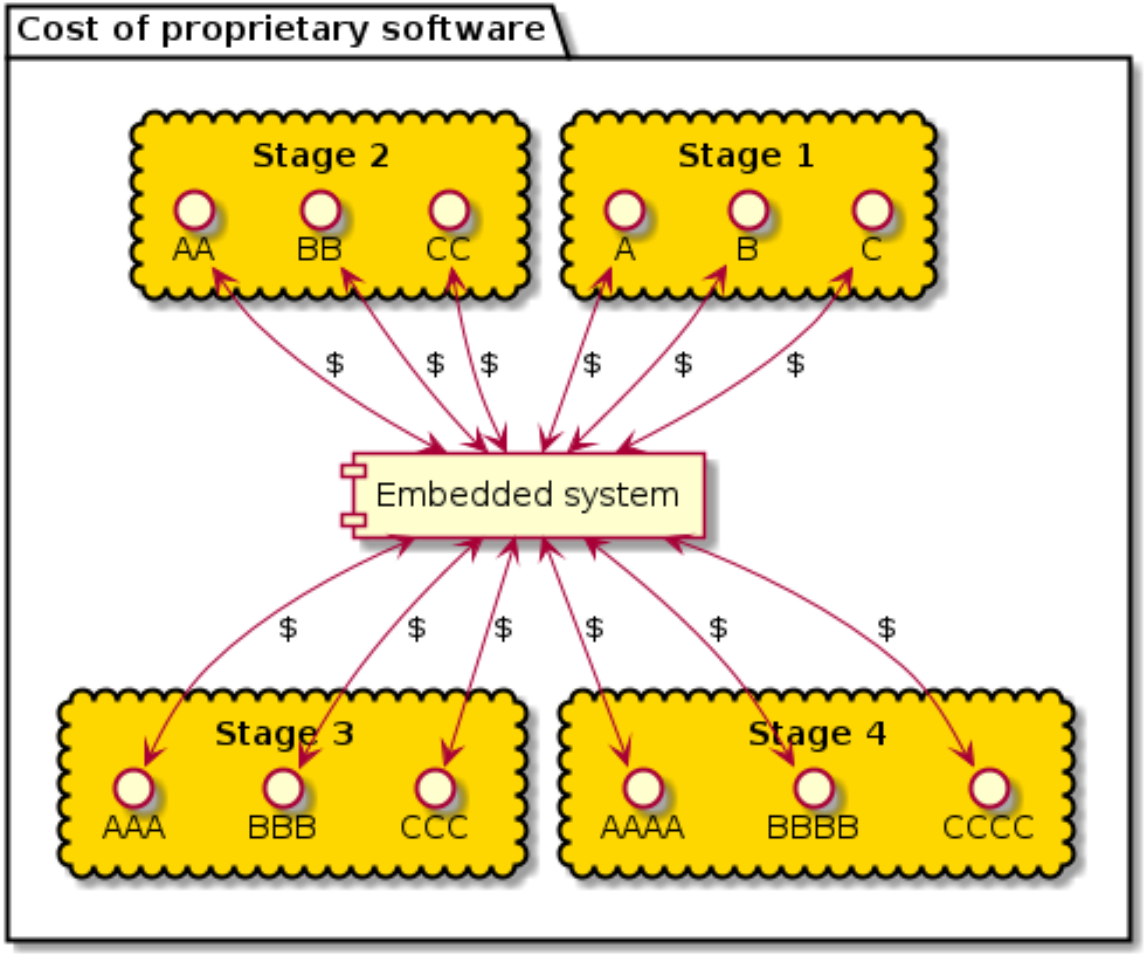


A parenthesis on a published standard

- AUTOSAR relies on
  - A **network of components** (see Simulink^{tm})
  - Regrouped and abstraction of sub-systems as features
- AUTOSAR is also providing
  - A communication architecture in-between ECUs
  - A micro system allowing a normalized API on ECUs
- It is also generating code to automatically handle communications

Technical solutions do exists, yet

- Potential multiplication of licenses (like AUTOSAR^{tm})
- Each OEM has to be dealt with individually still
- Mandatory design coordination between OEMs



Other limitations do exist

Design wise

- **Lack of dynamism**
  - No easy software redundancy
  - Deep component have to be recompiled in if changed
- **Heavily tied** components
  - Dependencies issues
  - Forces old patchworks to run along new code

Development cycle complexity

- **Difficulties to** build a **replay** environment
  - Increased release cycle, less coverage
- **Difficulties to optimize** the whole system
  - More computing power required
  - Less predictable requirements
- **Difficulties to validate** OEMs deliveries
  - Need to be done by hand, each time
  - Error prone, leads to un-diagnosed issues being brought in

Break n°2, before presenting the new model

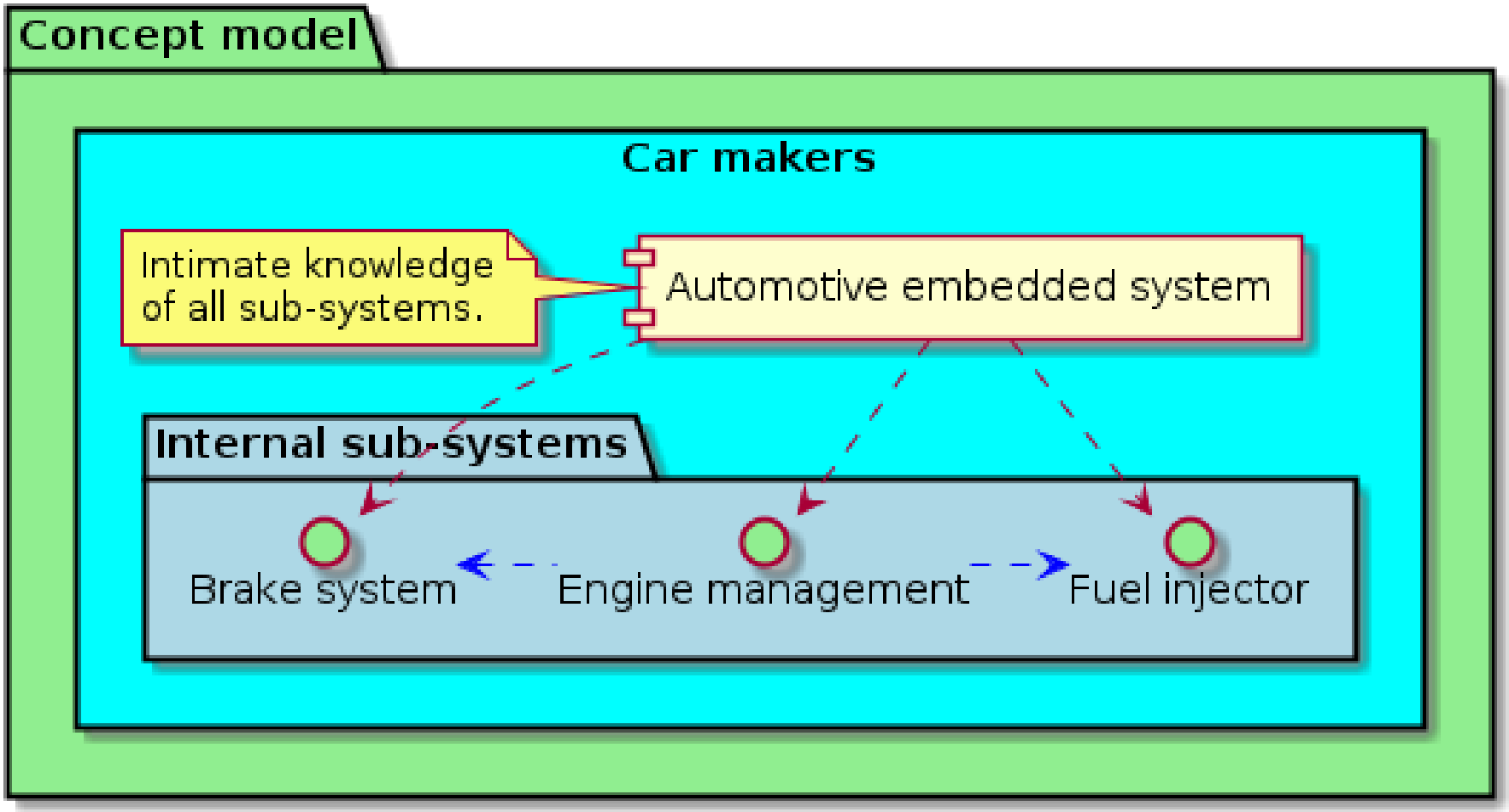
Subsection 3

Introduction to the concept model

Incremental list of modifications

1 Appropriation of sub-systems

- First, we learn how those sub-components work



This requires to handle more, but

1.1 OEMs are not in the loop anymore

- More flexibility regarding sub-systems development
- Intern communication, can be made on peer to peer
- Cheaper, no more overhead cost, nor licenses

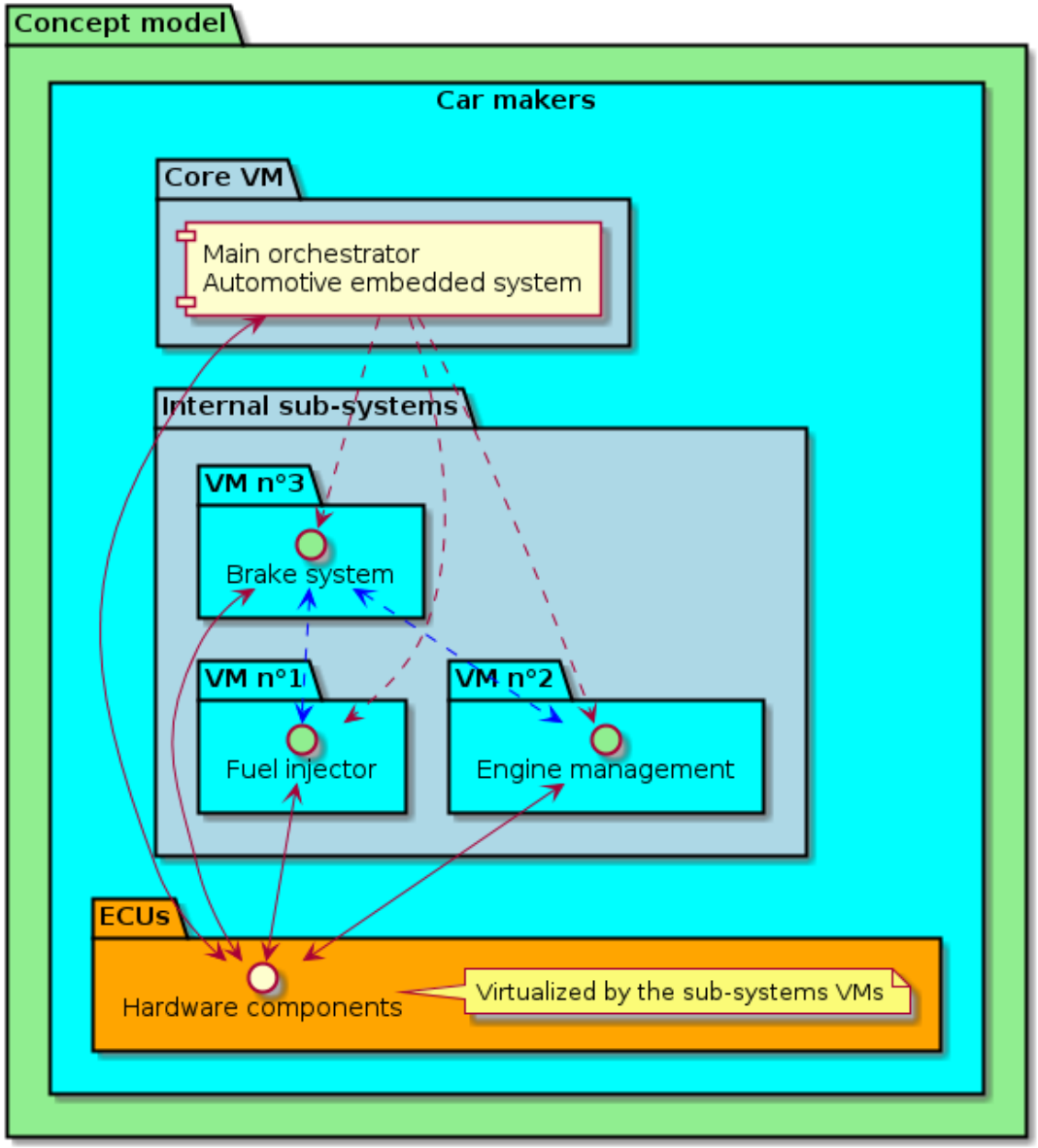
1.2 Sub-systems can be worked on directly

- The orchestrator can manage sub-systems completely
- Sub-systems can be reworked to work with each-others

1.3 Resources can now be shared

- Development resources, libraries
- Physical resources, ECUs, computing power, disk space

2. Isolation of every components



3. Truly exploiting the benefits of the model

- Finally, we capitalize on that and start to do real things

3.1 With technical features like

- Load balancing | Dynamic management | Security
- Static optimizations | A/B benchmarks | Redundancy

3.2 And a motivation boost

- It is always rewarding to work on challenging features

Investigations to do next on virtualization

Status

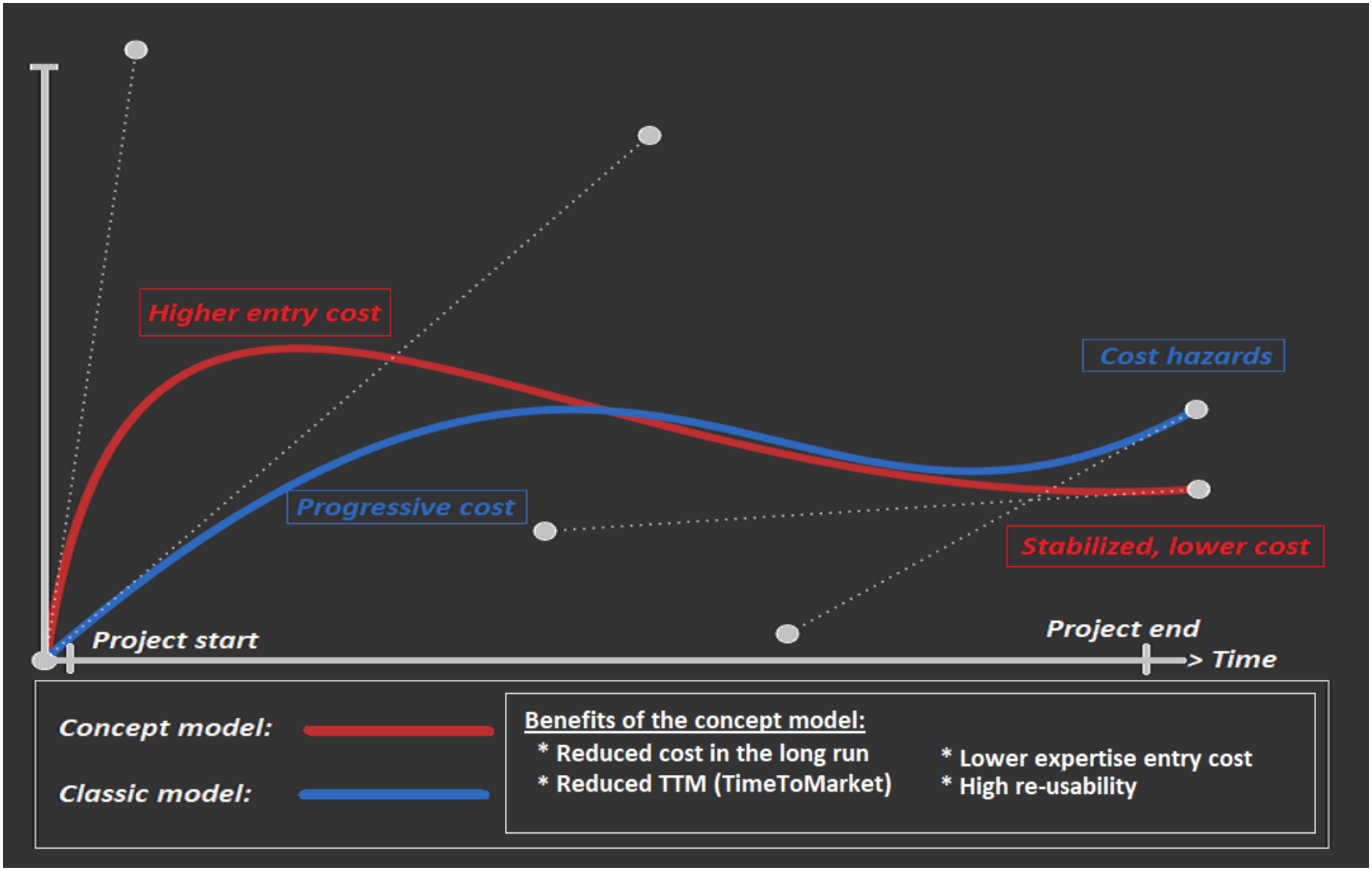
- Virtualization is a **mature technology**

Some questions on the subject

1. Is it **compatible** with automotive real-time systems constraints ?
2. How much **does it cost** ? performance wise ?
3. In practice, **does it fit** well ?
4. Is it **easy enough** to manipulate ? debug ? configure ?

[X] **This will have to get analyzed further down the research**

Last point, project timeline & costs differences



Everyone good so far ?

Any questions before we go to the last part of the presentation ?

- ☒ Part 1/3: Introduction
- ☒ Part 2/3: Research description and models
- ☐ Part 3/3: Administrative details, timeline and future work

Section 3

Administrative details, and research timeline

A research as part of a thesis

- 3 years PhD in the industry
- Publishing and hosting conferences
- Sharing knowledge

Research goals

Short term

- Digging **sub-systems incorporation**
- Introducing **low level virtualization**
- Challenging a **simplification of the hardware** interface
- **Challenging the** existing **model** deeper

Mid term

Macro goals

- Standalone **prototype** realization
- Answering a **client project** call
- Maybe even a **small team** to boost the results
- Submitting **Patents**

Technical goals

- Extend virtualization
- Incorporate other sub-systems
- **Crushing that complexity issue**

External parties, and sub-contractors

- Providing new needs and constraints
- Establishing the work with ECU suppliers

Long term

Macro goals

- Publishing results
- Founding a new project
- Having a production opportunity

Technical goals

- Fleet of VMs
- Development of a VMs orchestrator
- Designing generic ECUs
- Compatibility with other models

Final goal

- Releasing a fully fledged product !
- And validating my PhD :)

A wide, yet focused research

- Milestones are clear
- Possibility to iterate step by step
- Known technologies, less deviation risks
- Expertise do exist on the subject

What's now ?

Work to be done

- Research on the state of the art
- Bibliographic work
- Testing the virtualization
- Validating all the show stoppers
- Submitting a project proposal

Thank you for attending

- Any questions ?