**Application Software Layer:**

→La **couche application** est la couche supérieure de l’architecture, elle est dédiée à l’implémentation des fonctionnalités spécifiques au système embarqué (comme le contrôle moteur, le système de divertissement, la gestion du climat, etc.)

* **Objectif principal**:

→ La couche application est conçue pour répondre ***aux besoins spécifiques de l'utilisateur final*** ou du système en fournissant des fonctionnalités automobiles spécifiques. Elle permet une certaine modularité et portabilité, en isolant la logique métier de l'infrastructure matérielle.

* **Composants principaux** :
  + **Application Software Components (ASWCs)** :

→ Ce sont des composants logiciels qui contiennent la logique fonctionnelle spécifique à l'application. Chaque composant est divisé en unités plus petites appelées **Runnable Entities**, qui sont les éléments exécutables définis pour des tâches spécifiques.

* + **Interfaces d'Application**:

→ Elles facilitent la communication entre les différents composants logiciels ***en utilisant les ports et interfaces définis*** dans AUTOSAR. Ces interfaces sont standardisées pour assurer la compatibilité et l'interopérabilité entre les composants.

* + **Communication** :

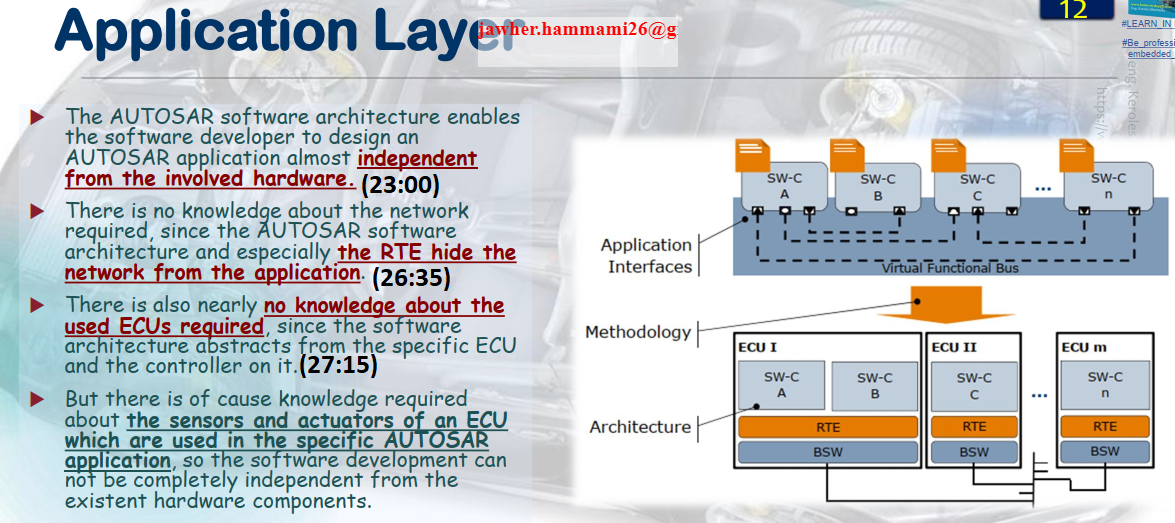
La couche application interagit avec la **couche de base AUTOSAR** (Basic Software Layer) via la couche de services (RTE - Runtime Environment), qui gère l'acheminement des données et les appels de fonction entre les composants de l'application et le matériel, et assure aussi la communication entre les différents composants.

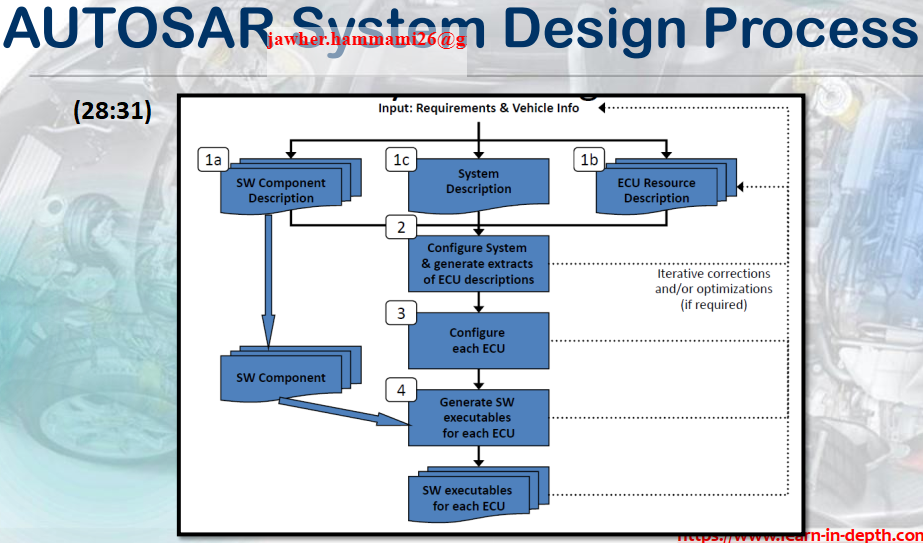
(5:30)

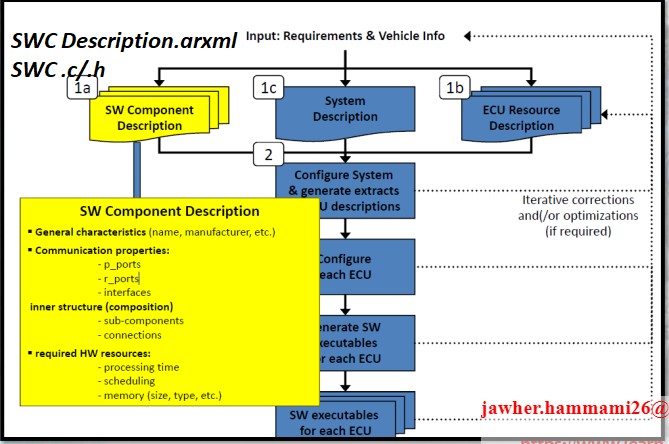
→ Dans la couche RTE (Runtime Environment) d’AUTOSAR, la décision de répartition des composants logiciels entre les différents **ECUs** (Electronic Control Units) est implémentée. Voici comment cela fonctionne :

* **Communication locale** : Si deux composants logiciels (SWCs) doivent communiquer et se trouvent sur le même ECU, ils échangent des données en utilisant une **mémoire partagée** (shared memory). Cela permet une communication rapide et directe entre eux.
* **Communication inter-ECUs** : Si les composants logiciels se trouvent sur des ECUs différents, la communication passe par l’écriture et la lecture dans la **mémoire de l’autre ECU**. Le transfert des données entre les ECUs est géré par les mécanismes de communication définis par AUTOSAR.
* **Fichier RTE généré** : La logique de communication est automatiquement générée dans le fichier RTE.c/.h, basé sur la description des composants et leur configuration, qui sont définies dans le fichier de description RTE. Ce fichier contient les informations nécessaires pour que le système sache comment chaque composant communique, que ce soit via la mémoire partagée ou par le réseau entre ECUs.
* **Différence entre VFB et RTE:** (23:00)

→ La **VFB** est une abstraction permettant de modéliser la communication entre composants logiciels **sans considérer** le matériel, tandis que la **RTE** est sa mise en œuvre concrète dans le système réel, gérant la communication en fonction de la configuration matérielle et de la répartition des composants sur les ECUs.







1) **SWC Description.arxml** :

→ Ce fichier est un **fichier XML** (AUTOSAR XML) qui définit la structure, les interfaces, les ports, et les dépendances du SWC.

→ Il contient toutes les informations nécessaires pour configurer le SWC dans le système AUTOSAR, notamment la définition des runnables, des ports (fournisseur/récepteur), et des types de données échangées.

**2) Fichiers .c/.h pour l'implémentation** :

→ Les fichiers **.c** et **.h** contiennent le **code source** et les **déclarations** pour l'implémentation des fonctionnalités définies dans le fichier .arxml.

→ Le fichier. h déclare les fonctions et structures du SWC, tandis que le fichier. c contient l'implémentation réelle des runnables et des fonctions qui réalisent les traitements spécifiques.

**→ La suite pages 15 & 16**