

# 지능형 자동차 서비스를 위한 차량 SW 플랫폼

전부선, 이정환, 한태만  
한국전자통신연구원

## 요약

과거 단순한 이동수단이었던 자동차는 다양한 안전과 편의적 서비스를 제공하게 되면서 다양한 활동이 수행되는 생활의 한 공간으로 자리매김하고 있다. 이에 따라 차량을 선택하는데 있어 소비자 취향에 맞는 다양한 부가 서비스를 제공하는지가 중요한 요소로 작용함에 따라 자동차 산업에 IT 기술의 접목에 대한 필요성이 대두되었다. 이러한 흐름에 따라 자동차 IT 융합이 가속화되고 있으며, 편의성, 안전성, 편리성을 위해서 차량 내에 많은 ECU의 사용이 필수적으로 요구되고 있다.

이러한 자동차의 부품에 소프트웨어가 탑재됨에 따라 공통적으로 사용되는 모듈에 대하여 재사용성, 부품 호환성 등의 요구가 생겨나게 되었고, 이를 해결하고자 하는 움직임이 차량 SW 플랫폼 표준화를 통해서 활발하게 이루어지고 있다.

본 고에서는 자동차 서비스 개발을 위한 대표적인 차량 전장용 SW 플랫폼인 AUTOSAR 플랫폼과 차량 내 헤드유닛을 통해 멀티미디어 및 텔레매틱스 서비스를 제공하는 차량용 인포테인먼트 시스템을 위한 표준 플랫폼인 GENIVI 플랫폼에 대하여 살펴보고자 한다.

## I. 서론

단순한 이동수단으로써 기계적 장치일 뿐이었던 자동차는 긴급 구난, 자동순항시스템, 추돌방지, 차선이탈경보 등의 능동적 안전 시스템을 도입하고, 첨단 운전자 지원 시스템(ADAS, Advanced Driver Assistance System)과 외부 통신망과의 연계 등을 통해 다양한 운전자 친화적 서비스를 제공하는 제 3의 생활 공간으로 발전하고 있다.

이에 따라 자동차 산업도 부품 모듈의 개별 제어 수준에 머물던 단계에서 점차 통합 모듈 제어 방식으로 진화되고 있으며, 신제품을 개발할 때에도 외관과 디자인 위주에서 안전, 편의와 관련된 서비스 개발위주로 변화하고 있어, 미래에는 차량의

80% 이상의 혁신적인 서비스가 전기전자 시스템을 기반으로 할 것으로 예상하고 있다.



그림 1. 자동차 서비스 패러다임의 변화

자동차 산업은 OEM이라 불리는 완성차 업체와 부품 개발업체로 나뉘는데, 완성차 업체들은 공통 부품 모듈을 자사의 다른 차량 모델에도 적용할 수 있기를 원하며, 부품 개발 업체들은 자신들이 개발한 부품을 여러 완성차 업체에 납품할 수 있기를 원한다. 즉 모듈의 재사용성, 호환성 문제가 불거지게 되는데 이러한 문제를 해결하기 위하여 세계 굴지의 완성차 업체, 부품 공급 회사 및 IT 기술 업체들이 협력하여 소프트웨어의 재사용성과 안전성 및 응용 소프트웨어의 하드웨어 의존성 제거 등을 목표로 차량용 소프트웨어 플랫폼 표준화를 진행 하고 있다.

본 고에서는 다음 두 장을 통해서 자동차 전장 소프트웨어의 표준 플랫폼인 AUTOSAR와 차량용 인포테인먼트 플랫폼인 GENIVI에 대한 동향과 주요 이슈를 살펴보고자 한다.

## II. 자동차 전장 SW플랫폼

### 1. AUTOSAR 개요

AUTOSAR는 AUTomotive Open System ARchitecture의 약자로서, 재사용성/분산 구조에 적합하도록 생성된 자동차 전장 SW 플랫폼이다.[1] 점점 복잡해지는 전장SW 기술 발전에 대응하기 위해, OEM과 공급자 간 전장SW 재사용(reuse) 및

호환성(exchangeability)을 개선하여 자동차 생산 비용 절감 및 새로운 전장 기능 개발 발판을 마련하는 것을 그 목적으로 한다.

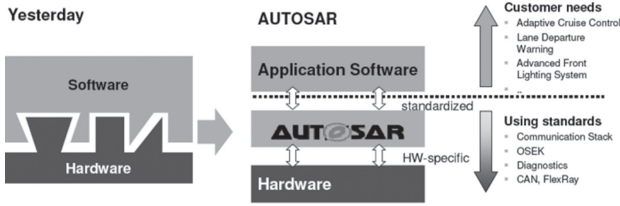


그림 2. AUTOSAR 비전

AUTOSAR 표준단체는 코어 파트너, 프리미엄 멤버, 관련 멤버의 3 단계 회원 자격 구조로 이루어져 있으며, 2011년 12월 현재 9개의 코어 파트너, 48개의 프리미엄 멤버, 81개의 관련 멤버 및 18개의 개발 멤버로 총 156개의 멤버로 구성되어 있다. 국내는 현대기아자동차, 한국전자통신연구원 이 프리미엄 멤버로 활동하고 있으며, 대성전기, 만도, 대구경북과학기술연구원이 관련 멤버로 활동 중이다.

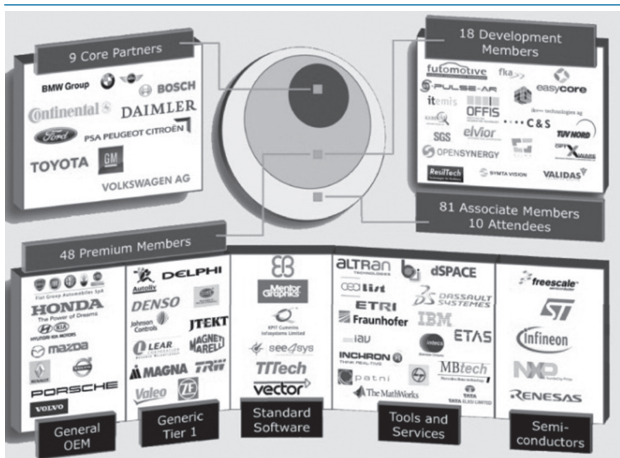


그림 3. AUTOSAR 주요 회원사

AUTOSAR는 2003년 AUTOSAR가 결성된 이후 지속적으로 표준을 제정·갱신하고 있다.

AUTOSAR에서는 자동차 도메인을 바디, 새시, 파워트레인, HMI, 멀티미디어/텔레매틱스 및 안전 분야로 나누고 각 워크 패키지 별로 표준화를 단계에 따라 Phase 1, 2, 3로 나누어 진행 중에 있다.

AUTOSAR 표준화 작업을 간단히 살펴보면 표준의 기본 틀을 개발한 후에 안정적인 구조의 방법론을 토대로 선별적인 표준 강화 작업을 진행한다. 이후 시장 상황에 맞추어 각각의 다른 버전의 표준을 지속적으로 유지 보수하고 있다.

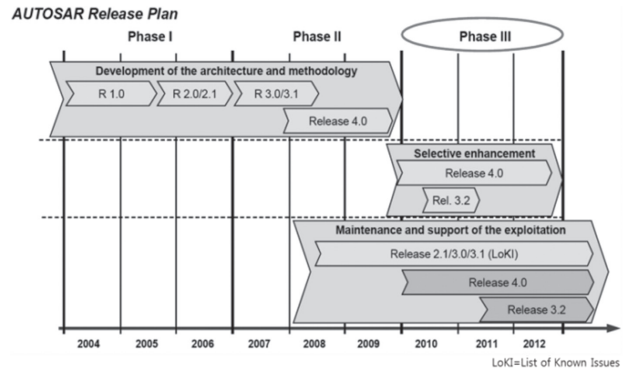


그림 4. AUTOSAR 표준화 진행 일정

표 1. AUTOSAR Release

협회 결성	시기	내용
협회 결성	2003.6	BMW, Bosch, Continental, Daimler Chrysl-er, Volkswagen, Siemens VDO 초기 코어 멤버 결성
Release 1.0	2005.1	Release 1.0 배포 (23개 BSW 완성)
Release 2.0	2006.5	Release 2.0 배포 (42개 BSW 완성) : SW architecture 및 methodology/ template의 기본 기능을 제공
Release 3.0	2007.12	Release 3.0 배포 (47개 BSW 완성) : Application Interface 카테고리의 7개 Spec을 개발함으로써 응용 인터페이스 제안
Release 3.1	2008.8	R3.0에 OBDII 서비스를 포함한 버전
Release 3.2	2011.5	Partial Networking 및 bug fixing 관련한 시장 요구사항을 반영하여 통신 스택을 강화한 버전
Release 4.0	2009.12	Release 4.0 배포 (안전성 개념 강화, 통신 스택 강화, 총 87개의 BSW 제공)
Release 4.1.1	2013.3 예정	Release 4.1.1 배포 (Backward compatibility, Acceptance Test 추가)

AUTOSAR에서는 최근 버전이 이전 버전을 갱신하고 버전간의 호환성을 보장하는 일반적인 버전 개념과는 상이하다. 각각의 버전은 각기 다른 특성을 포함하는 것이 특징이다. 물론 상위 버전은 하위 버전의 서비스 컴포넌트를 포함하기는 하지만 이전 버전과의 호환성을 보장하지는 않는다. 업계에서도 최신 버전을 채택하기 보다는 자사가 제공하는 서비스의 특징에 따라 알맞은 버전을 선택하여 사용하는 것이 추세이다. 이와 같은 이유로 Phase 3에서도 여전히 3.x 버전의 표준화 작업이 지속적으로 진행되고 있다. 2011년 5월 3.2.1 버전이 배포된 것도 같은 맥락이며 이 버전은 4.0 버전과 호환성을 보장한다.

## 2. AUTOSAR SW 구조

AUTOSAR 구조는 크게 AUTOSAR SW-C, VFB(RTE),

BSW의 3 계층으로 나누어진다.

중간에 VFB(Virtual Function Bus)라 하여 응용 소프트웨어 (SW-C)와 HW/시스템 관련 소프트웨어 (BSW, Basic Software)를 분리함으로써, HW 및 플랫폼에 독립적인 응용 서비스를 개발할 수 있도록 하는 것이다.

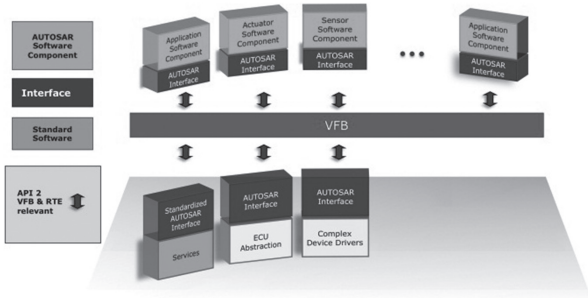


그림 5. AUTOSAR Layered Architecture

각 AUTOSAR SW-C는 응용 SW 기능의 일부를 구현하며, ECU에 매핑 되는 기본 단위이며, 포트와 인터페이스를 통하여 상호 송수신 할 신호와 데이터를 정의하고 정의된 규격에 따라 태스크들의 동작으로 메시지들을 교환한다.

VFB(Virtual Functional Bus) 혹은 RTE(Run Time Environment)는 각 SW-C 사이 및 SW-C와 BSW 사이의 정보 교환을 위한 통신의 중추적인 역할을 한다. RTE는 하부 많은 서비스 계층의 컴포넌트들을 추상화하여 API들을 제공하여 향후에 ECU 보드가 바뀌더라도 상위 제작된 소프트웨어 컴포넌트 수정 변경 없이 사용할 수 있도록 가상 버스 개념을 접목하고 있다.

BSW는 Basic Software의 약자로서, VFB 하단에 존재하며 다양한 ECU에 공통으로 적용 가능하게 만드는 Layer로 HW와 SW를 분리 시키는 역할을 한다. 이러한 요소로 AUTOSAR는 “component based development”가 가능해진다.

BSW의 표준 계층으로 Service Layer, EAL (ECU Abstrac-

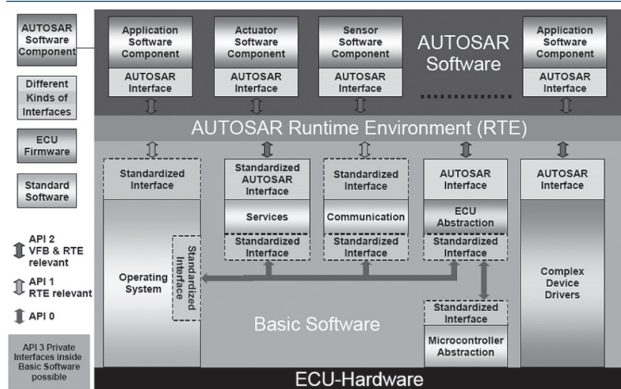


그림 6. AUTOSAR SW 아키텍처

tion Layer), MCAL (Micro-controller Abstraction Layer) 그리고 CDD (Complex Device Driver)로 구성된다[4].

BSW의 Service 계층은 OS, 네트워크, 메모리, 검증, ECU 상태관리 등의 서비스 기능을 수행한다. EAL은 ECU 내부의 장치들과의 인터페이스를 제공하며, ECU에 독립적인 상위 계층의 설계를 제공한다. MCAL은 상위 계층에서 마이크로 컨트롤러의 레지스터를 직접 조작하는 것을 피하게 해주며, 디지털 입출력, 아날로그 디지털 변환, 파형변환, 직·병렬 변환 등으로 구성된다.

이러한 계층화에 반하여 기존 동작중인 안정화된 장치들을 AUTOSAR 플랫폼과 호환성을 가지고 동작될 수 있도록 CDD 라는 개념으로 수용하고 있다. AUTOSAR에서는 CDD를 제공함으로 기존에 안정적으로 동작되던 장치들을 최소한의 수정으로 지속적으로 서비스할 수 있도록 Backward Compatibility 를 제공하려고 노력하고 있다.

### 3. AUTOSAR 의 적용

<그림 7>에서는 전장 응용 서비스의 AUTOSAR 개발 방법론을 보여주고 있다. AUTOSAR SW 개발은 시스템 설정단계와 ECU 설정단계로 나누어진다.

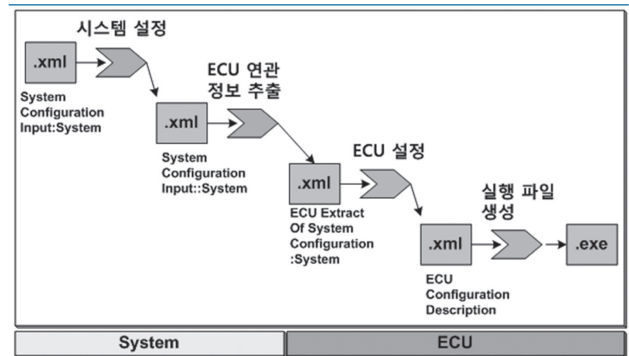


그림 7. AUTOSAR 표준 개발방법론

시스템 설정 단계에서는 SW-C 및 이들 간의 포트/인터페이스 연결 상태, 주고받는 데이터 타입 등을 기술하는 SW-C 명세서(Component Description), 각 ECU의 HW 구성을 기술하는 ECU 자원명세서(Resource Description), 그리고 버스 시그널, 토폴로지 등 시스템 제약명세서 (Constraint Description)를 작성한다.

각 SW-C 내부에는 응용 SW 구현을 위한 태스크 동작 정의 및 트리거 조건을 정의한다. 다음은 SW-C를 각 ECU에 매핑하고 네트워크 설계를 하여 시스템 설계명세서(System Configuration Description)를 기술한다.

다음 단계인 ECU설정 단계에서는 시스템 설계명세서로부터 각 ECU 정보를 추출하여 ECU 설정을 하며, 태스크 정의 및 할당, RTE 생성, BSW 설정을 통해 ECU 설계 명세서(Configuration Description)를 기술한다. 응용 SW와 함께 RTE, OS, Communication 등의 AUTOSAR SW 모듈 코드를 생성하고, 컴파일, 링크를 거쳐 실행 파일을 만들어 ECU 응용서비스를 구현한다. 구현된 동작 가능한 결과물은 설정된 ECU에 올려 시험할 수 있다.

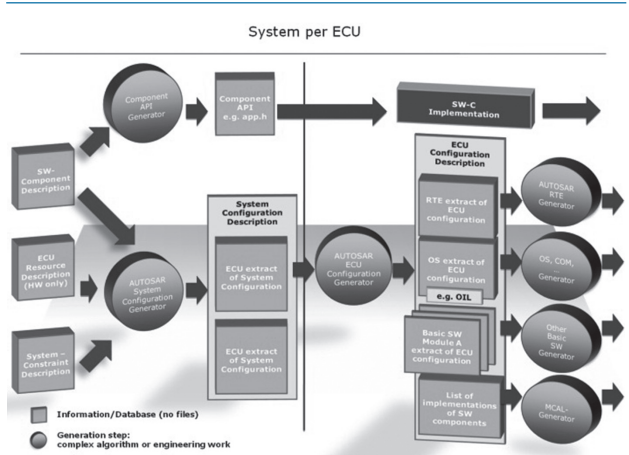


그림 8. AUTOSAR 방법론에 따른 SW 설계

이 때, AUTOSAR에서 정의한 표준에 맞게 공통 모듈인 BSW 모듈을 설정하게 되는데, ECU의 환경과 지원 기능에 따라 다양한 설정이 가능하며, AUTOSAR에서는 각 BSW 모듈에 대한 명세 및 설정에 대한 표준을 제공하고 있다. AUTOSAR에서는 BSW 모듈 설정에 있어 표준이 되는 Meta Model을 제공하는데, 해당 정보는 AUTOSAR EcuParamDef.arxml 이라는 파일에서 제공된다. 이 메타 모델에서는 각 모듈과 속성들에

Module Definition: Can	Module Definition: Can
MODULE-DEF	((SHORT-NAME, LONG-NAME?, DESC?, CATEGORY?, ADMI
UID	ECUC:f7ca73a3-b05b-4f6a-a64d-9a6e8da3314e
SHORT-NAME	Can
DESC	((L-2*))
LOWER-MULTIPLICITY	0
UPPER-MULTIPLICITY	*
SUPPORTED-CONFIG-VARIANTS	((SUPPORTED-CONFIG-VARIANT*))
CONTAINERS	((CHOICE-CONTAINER-DEF   PARAM-CONF-CONTAINER-DEF
PARAM-CONF-CONTAINER-DEF	Parameter Container Definition: CanConfigSet
UID	((SHORT-NAME, LONG-NAME?, DESC?, CATEGORY?, ADMI
SHORT-NAME	ECUC:e28c636a-f5f1-4503-b0ad-aed7fbed80d7
DESC	CanConfigSet
LOWER-MULTIPLICITY	((L-2*))
UPPER-MULTIPLICITY	1
MULTIPLE-CONFIGURATION-CONTAINER	true
SUB-CONTAINERS	((CHOICE-CONTAINER-DEF   PARAM-CONF-CONTAINER-DEF
PARAM-CONF-CONTAINER-DEF	Parameter Container Definition: CanController
UID	((SHORT-NAME, LONG-NAME?, DESC?, CATEGORY?, ADMI
SHORT-NAME	ECUC:s79e27f6-cf20-496c-8587-edf4d9d2f01b
DESC	CanController
LOWER-MULTIPLICITY	((L-2*))
UPPER-MULTIPLICITY	1
MULTIPLE-CONFIGURATION-CONTAINS	false
PARAMETERS	((BOOLEAN-PARAM-DEF   DERIVED-BOOLEAN-PARAM-DEF
REFERENCES	((CHOICE-REFERENCE-PARAM-DEF   FOREIGN-REFERENCE-P
SUB-CONTAINERS	((CHOICE-CONTAINER-DEF   PARAM-CONF-CONTAINER-DEF
PARAM-CONF-CONTAINER-DEF	Parameter Container Definition: CanHardwareObject
UID	((SHORT-NAME, LONG-NAME?, DESC?, CATEGORY?, ADMI
SHORT-NAME	CanGeneral
PARAM-CONF-CONTAINER-DEF	((SHORT-NAME, LONG-NAME?, DESC?, CATEGORY?, ADMI

그림 9. AUTOSAR Meta Model

대하여 CONTAINER, LOWER-/UPPER-MULTIPLICITY, SUB-CONTAINERS 등의 모듈 및 속성들에 대한 제약 사항을 정의하고 있다.

AUTOSAR Meta Model에서 정의하고 있는 제약조건에 준하여 적절히 설정된 BSW는 XML 기반의 AUTOSAR 데이터 저장 방식인 ARXML 파일로 저장된다. 이러한 데이터는 AUTOSAR를 지원하는 모든 플랫폼과 도구에서 공통적으로 사용 가능하기 때문에, 데이터의 공유 및 전달을 표준화 할 수 있다.

MODULE-CONFIGURATION	
UID	DCE:5944d518-580b-4654-a723-bfcc8a28a94
SHORT-NAME	Can
DEFINITION-REF	
IMPLEMENTATION-CONFIG-VARIANT	VARIANT-PRE-COMPILE
CONTAINERS	
CONTAINER	
CONTAINER	
UID	DCE:e842c464-8a45-4221-bcda-1f24804ce3f9
SHORT-NAME	CanConfigSet
DEFINITION-REF	
SUB-CONTAINERS	
CONTAINER	
UID	DCE:174cd0d1-a0a8-4c2f-8cc6-d76ad0e3b35c
SHORT-NAME	ECS
DEFINITION-REF	
DEST	PARAM-CONF-CONTAINER-DEF
	/AUTOSAR/Can/CanConfigSet/CanController
PARAMETER-VALUES	
ENUMERATION-VALU	
DEFINITION-REF	
DEST	ENUMERATION-PARAM-DEF
	/AUTOSAR/Can/CanConfigSet/CanController/CanBusoffProcessing
VALUE	INTERRUPT
BOOLEAN-VALUE	
DEFINITION-REF	
DEST	BOOLEAN-PARAM-DEF
	/AUTOSAR/Can/CanConfigSet/CanController/CanControllerActivation
VALUE	true
INTEGER-VALUE	
DEFINITION-REF	
DEST	INTEGER-PARAM-DEF
	/AUTOSAR/Can/CanConfigSet/CanController/CanControllerBaudRate
VALUE	0

그림 10. BSW 설정 파일 (ARXML)

이러한 AUTOSAR 방법론에 따라 현재 자동차 완성차 업체에서는 규격이 적용된 자동차를 적용하고 있으며, 선두주자인 BMW가 2006년에 시험 적용한 이후, AUTOSAR의 9개 핵심 멤버들은 자사의 차량에 2012년까지 단계적으로 AUTOSAR 플랫폼을 적용하기로 공표했다.

2011년 2,500만개의 AUTOSAR 기반 ECU가 생산되었으

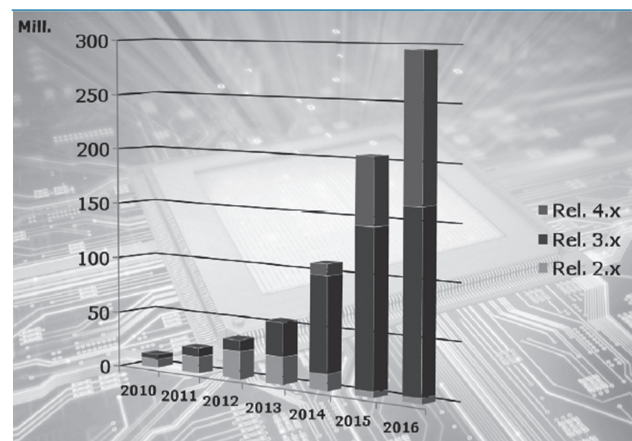


그림 11. AUTOSAR ECU 규모

며, 2016년에는 전체 ECU의 25% 이상을 차지하는 약 3억개의 ECU가 AUTOSAR를 기반으로 생산될 것으로 예측되고 있다 [5].

### Ⅲ. 차량용 인포테인먼트 플랫폼

#### 1. 인포테인먼트 플랫폼 개요

GENIVI는 GENEVA In-Vehicle Infotainment의 약자이며, GENEVA에서 최초 결성된 오픈소스 기반의 차량 멀티미디어 표준 SW 플랫폼이다.[2]



그림 12. 차량용 인포테인먼트 시스템

차량에 탑재되는 차량용 헤드유닛은 초기 단순한 cd, 라디오, 테잎 등을 재생시켜주는 매체에서 AVN(Audio/Vedio/Navigation)으로 진화하여, 최근에는 차량용 IT 융합 서비스 기술을 접목한 차량용 인포테인먼트 시스템(In-Vehicle Infotainment, IVI)으로 진화하는 추세이다. 텔레매틱스 개념 이후 등장한 인포테인먼트 시스템은 정보와 엔터테인먼트를 합친 개념으로써 3G, WiFi, 블루투스 등의 무선 통신을 이용하여, 위치추적 및 원격 차량진단, 사고감지, 교통정보 및 홈네트워크와 사무 자동화, 인터넷 접속, 스마트폰 등의 외부 기기 및 인터넷과의 연동 등 다양한 IT 융합 서비스들이 가능하다는 특징이 있다.

이러한 인포테인먼트 시스템을 위하여 BMW, ARM, Visteon 등을 주축으로 GENIVI 연합이 결성되었으며 차량용 멀티미디어 플랫폼 표준화를 시작하였다.

현재 세계 자동차 시장은 비개방형 플랫폼에서 오픈 소스 기반의 플랫폼을 적용하는 방향으로 변화하고 있다. 기존에는 차량 모델별, 등급별, 제조사별 천차만별로 특정 업체에 종속적인 플랫폼에서 개발하던 것에서, 국제 표준 플랫폼을 기반으로 개발함으로써 재사용성 및 안정성을 높이고, 개발기간을 단축하며, 다양한 차량 IT 융합 서비스 개발이 편리하다는 장점을 취할 수 있기 때문이다.

기존의 IVI(In-Vehicle Infotainment) 플랫폼으로는 Microsoft, QNX, Microltron, Android, Linux 등이 있다. Microsoft는 시장에서 검증된 상품이지만 개발자 커뮤니티에

대한 개방성이 부족하고 고가라는 단점이 있고, QNX는 IVI에서 검증된 이력을 가지고 있고, 상대적으로 낮은 가격이지만, 미래의 확장성, 특히 고수준의 사양을 통합하는 차세대 시스템에 대한 지원이 부족한 상황이다.

스마트폰 시장 공략을 목표로 최근 자동차 회사들이 안드로이드 플랫폼을 탑재하여 IVI 시스템을 상용화하기도 하고, 유럽 자동차 시장을 중심으로 차량용 산업 표준 인포테인먼트 플랫폼인 GENIVI를 기반으로 한 제품 출시도 발표되고 있는 상황이다. 하지만 안드로이드가 IVI 시장의 대안으로 자리 잡은 것은 아니다. 안드로이드 뿐 아니라 다른 모바일 플랫폼은 차량용 플랫폼으로 대체될 수는 없다는 것이 자동차 업계의 대체적인 시각이다. 그 이유는 모바일 플랫폼이 차량용 플랫폼으로 적용되기 위해서는 차량용 특성을 고려한 플랫폼으로 변화되어야 하고 표준화 또한 진행이 되어야 하지만, 안드로이드 플랫폼은 모바일 시장에서 치열한 경쟁 중에 있기 때문에, 추가적으로 차량용 플랫폼으로 진화하기 위한 어떤 노력도 현재 이루어지지 못하고 있기 때문이다.

이와 같이 기존 플랫폼이 가지는 단점들로 인하여 새로운 플랫폼에 대한 요구가 늘어나고 있는 상황에서 GENIVI 연합은 차량용 인포테인먼트를 위한 멀티미디어 플랫폼 표준화를 위하여 BMW, ARM, Visteon 등의 창단 멤버를 주축으로 표준화를 시작하였으며 3년 정도가 지난 현 시점에 멤버의 수는 160개 이상으로 늘어날 정도로 그 활동이 활발하게 이루어지고 있다.

#### 2. GENIVI 협회 현황

MW, Intel, Visteon, GM, Magniotti Marelli, 윈드리버 등을 중심으로 2009년 초 설립되었으며, 자동차 산업별 회원사의 요구사항을 취합하여 산업 표준화를 진행하고 있다. GENIVI의 멤버는 현재 2012년 4월 기준으로 163개의 회원사가 가입이 되어 있으며, 주로 OEM, Tier-1, 반도체, OS 및 S/W 솔루션 업체 등으로 구성되어 있다.



그림 13. GENIVI 회원사

GENIVI Activity 구조는 기술적인 표준화 활동을 수행하는 전문가 그룹(EG, Expert Group)과 매니저 역할을 수행하는 PMO(Program Manager Office), 각 EG의 컴포넌트를 통합하는 SAT(system Architecture Team), 그 외 기술 외적인 업무를 수행하는 Marketing, Legal Workgroup, IT Infrastructure 및 각 취합된 요구사항에 대하여 Compliance 여부를 판단하는 GENIVI 이사회와 기술 고문으로 이루어져 있다.

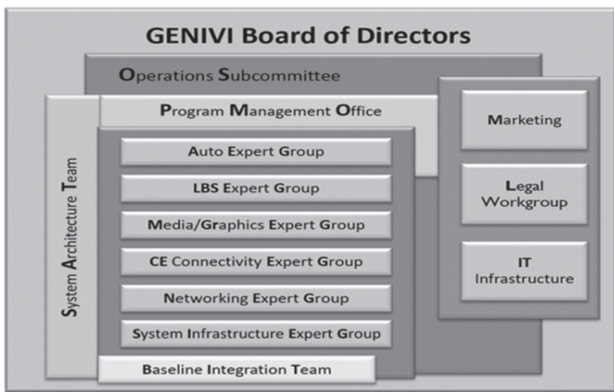


그림 14. GENIVI Activity 구조

국내에서는 ETRI가 최초로 2009년 하반기 Core 멤버로 가입을 한 후, 현대자동차, 삼성전자, 현대모비스, LG전자, 인포뱅크, 오비고, 슈어소프트 등이 참여하면서, 현재 GENIVI KOREA 지역 전문가 그룹을 발족하여 한국 자동차 시장의 요구사항을 도출하여 표준화 작업을 진행하고 있다.

GENIVI 협회의 멤버는 최근 1년 사이 그 수가 두 배로 늘었는데, 이는 GENIVI 멤버인 BMW, GM등이 GENIVI 멤버로 가입되지 않은 회사로부터는 납품을 받지 않겠다는 방침을 천명한 것과도 관련이 깊다.

### 3. GENIVI 플랫폼 구조

GENIVI는 개방형 IVI 플랫폼이며 공개 소프트웨어 기반의 자동차 헤드유닛 플랫폼 명세를 정의한다.

GENIVI에서는 어느 회사, 어느 플랫폼에서도 공통적으로 쓰이는 공통 미들웨어 부분을 표준화를 함으로써 기존의 소비적이던 플랫폼 경쟁에서 벗어나, 회원사의 목적에 따라 경쟁 영역에 집중함으로써 비용과 시간에 대한 효율적인 투자가 가능해진다.

<그림 15>에서 보여지듯이 화면 하단의 노란 박스가 표준화를 통해 공통으로 사용하는 GENIVI 플랫폼이며, GENIVI의 표준 Stack 위에 HMI, Apps, OSV Extensions, Libs 등의 각 회사별로 경쟁 영역이 올라가게 된다.

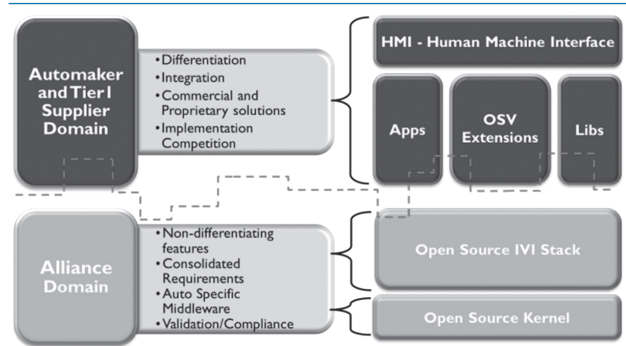


그림 15. GENIVI Target

이러한 구조를 통해 차량용 인포테인먼트 응용을 위해서 구축해야 할 플랫폼 개발에 대한 개발 비용을 감소할 수 있다. 기존 플랫폼 개발은 인포테인먼트 개발비용의 50% 이상을 차지하고 있었던 만큼 상당한 비용적 효과를 가져온다. 또한 응용 개발을 위해 필요한 미들웨어를 모두 개발할 필요가 없고 업체별로 필요한 요구는 표준화하여 GENIVI 컴포넌트화함으로써 개발 시간도 단축할 수 있다. 또한 국제 자동차 시장으로의 진입이 보다 자유로워 질 수 있고 오픈 소스 기반으로 많은 개발자를 확보할 수 있다. 결과적으로 자동차 특화 플랫폼을 공통으로 보유하여 소비적인 경쟁을 지양하여 이익을 극대화 하겠다는 의도인 것이다.

GENIVI는 오픈소스 기반으로 요구사항을 취합하여 산업표준화로 구성된 플랫폼이다. 따라서 기존의 오픈소스 성격과 달리 GENIVI 멤버에 한해서 플랫폼이 개방되며 이를 상용화하기 위해서는 GENIVI Alliance 정책에 준해야 한다. 또한 멤버가 가진 기술을 GENIVI플랫폼에 Compliance로 등록하기 위해서는 기술을 개방되 그에 따른 로열티에 대한 권리는 주장할 수 없도록 규정되어 있다.

현재 GENIVI 플랫폼은 Kernel(현재 Ubuntu 기반) 및 기본 플랫폼의 80%를 Linux Foundation에서 제공하는 오픈소스를 기반으로 하고, 그 외 기존 오픈 소스 모듈을 IVI에 맞도록 수정

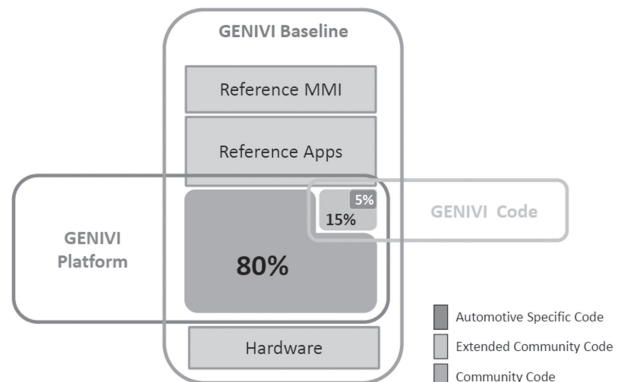


그림 16. 오픈소스 기반의 GENIVI 플랫폼 구조

보완한 부분이 약 15%, 그리고 차량 특화된 기능으로 GENIVI Alliance에서 자체 개발한 모듈이 약 5%정도를 차지하고 있다.

GENIVI는 현재 전문가그룹(EG)인 Automotive, CE Connectivity, HMI-Application Framework, Location -Based Services, Media and Graphics, Networking, System Infrastructure 등 총 7개의 Domain EG와 Korea, China 등 2개의 지역(Regional) EG를 중심으로 표준화 작업을 진행하고 있다. 각 EG에서는 워크 아이템에 대하여 요구사항을 도출하고, API spec을 확정하며, 이에 맞는 오픈소스가 있는지 찾아보고 요구사항을 제대로 충족하는지 PoC 테스트를 수행하여 적합할 경우 오픈 소스를 GENIVI Compliance로 채택할 수도 있고, 이에 요구사항에 맞추어 자체 개발 작업을 수행하기도 한다.

7개의 전문가 그룹에서 수행하는 표준화 작업은 다음과 같다 [3].

Automotive EG는 차량 진단 프로토콜 및 DLT(Diagnostic Log and Trace)에 대한 표준화, 노드간 통신 방식이나 AUTOSAR 연계, Virtualization 등의 ECU Architecture에 대한 표준화, Vehicle Network 및 Interface 분야에 대한 표준화를 진행하고 있다.

CE Connectivity EG는 사용자 기기(CE Device)들에 대한 연동 기능에 대한 표준화 작업을 수행하고 있으며, 주로 블루투스, 연결 장비 관리 기능 등에 대한 연구를 진행하고 있다.

LBS EG는 위치 기반 서비스(Location-Based Services)를 위한 미들웨어에 대한 표준화를 진행하고 있으며, Positioning API, Navigation API, Map Data Access 등에 대한 연구를 진행하고 있다.

Media and Graphics EG는 말 그대로 오디오 및 멀티미디어, 그래픽에 대한 관리 기능을 표준화하고 있다.

Networking EG는 차량과 다른 차량간의 통신, 혹은 IT Infrastructure에 대한 연결에 대하여 표준화를 진행하고 있으며, 주로, 브라우저 프레임워크, 다운로드/업로드 매니저, 서비스 프레임워크, 연결 관리 기능 등에 대한 연구를 진행하고 있다.

System Infrastructure EG는 시스템이 시작하면서부터 종료할 때까지 Lifecycle 전반에 걸친 관리 구조를 표준화 하고 있으며, 커널 및 파일 시스템, IPC나 IDL, Persistence, Personalization 에 대한 연구도 함께 진행하고 있다.

HMI/Application Framework EG는 2012.4월 새로이 추가된 7번째 EG로써, 그동안 경쟁영역이라고 판단되어 GENIVI 플랫폼의 영역 밖으로 여겨던 HMI Framework과 Application Framework을 정의할 예정이다.

GENIVI 플랫폼은 2009년 12월 GENIVI 1.0 출시를 시작으로 현재 2012 4월 5번째 release인 Discovery 버전이 출시되었

고, 매 해 4월과 10월에 All Member Meeting을 진행하며 이때를 맞추어 매해 두번씩 버전업을 발표하고 있다.

## IV. 결론

현재 세계 자동차 시장은 비개방형 플랫폼에서 오픈 소스 기반의 플랫폼을 적용하는 방향으로 변화하고 있다. 기존에는 차량 모델별, 등급별, 제조사별 천차만별로 특정 업체에 종속적인 플랫폼에서 개발하던 것에서, 국제 표준 플랫폼을 기반으로 개발함으로써 재사용성 및 안정성을 높이고, 개발기간을 단축하며, 다양한 차량 IT 융합 서비스 개발이 편리하다는 장점을 취할 수 있기 때문이다.

AUTOSAR와 GENIVI에서 제안하는 플랫폼은 이러한 차량 산업계의 흐름에 편승한 여러 기술들을 제안하고 있으며, 국내에서도 산업체와 연구소를 중심으로 표준화 작업 및 실제 제품 개발에 활발하게 참여하고 있다.

AUTOSAR와 GENIVI의 표준을 주도하고 있는 자동차 제조사들이 시장을 크게 선점하고 있는 상황에서, 해당 업체들이 향후 AUTOSAR 또는 GENIVI 기반의 부품만을 납품 받겠다고 공언한 것은 간단히 무시할만한 일이 아니다. 표준화된 소프트웨어 플랫폼을 적용하여 개발하는 것은 자동차 제조사에게도, 부품을 납품하는 부품업체에게도 개발 기간과 비용을 줄이고, 재사용성 및 대체 가능성을 높여준다는 점에서 모두에게 환영받을 만한 일이다.

이러한 추세로 볼 때, AUTOSAR 기반의 ECU가 탑재된 스마트카에서 GENIVI 기반 인포테인먼트 시스템으로부터 다양한 지능형 서비스를 제공받는 시대가 멀지 않은 것으로 보인다.

## 참고 문헌

- [1] AUTOSAR Home page, <http://www.autosar.org>
- [2] GENIVI Home page, <http://www.genivi.org>
- [3] GENIVI Wiki Page (members only), <https://collab.genivi.org/wiki/display/genivi/GENIVI+Home>
- [4] AUTOSAR Layered Software Architecture, AUTOSAR, Dec. 2007.
- [5] AUTOSAR - The worldwide Automotive Standard for E/E systems Brochure

## 약 력



전 부 선

1998년 충남대학교 컴퓨터과학 학사  
 2000년 충남대학교 컴퓨터과학 석사  
 2000년~2000년 삼성전자 중앙연구소 연구원  
 2000년~현재 ETRI 자동차융합플랫폼연구팀 선임 연구원  
 관심분야: 자동차SW플랫폼, 소프트웨어 엔지니어링



이 정 환

1993년 순천향대학교 공학사  
 2009년 연세대학교 공학석사  
 1995년~2000년 LG전자 중앙연구소 주임연구원  
 2000년~2007년 아이비트 수석연구원  
 2007년~2011년 로텍 기업연구소 연구소장  
 2011년~현재 ETRI 자동차융합플랫폼연구팀 선임 연구원  
 관심분야: TCP/IPv4, IPv6, USN, AUTOSAR



한 태 만

1985년 경북대학교 전자공학사  
 2004년 충남대학교 컴퓨터공학 석사  
 2011년 충남대학교 컴퓨터공학 박사수로 후 과정  
 1986년 삼성전자 근무  
 1987년 LG전자 근무  
 1988년~현재 ETRI 자동차융합플랫폼연구팀 팀장  
 관심영역: 자동차SW플랫폼, 임베디드시스템, 시스템공학, SW 품질 인증