

The Siemens logo is displayed in a bold, teal, sans-serif font.

Ingenuity for life

The Siemens Digital Industries Software logo is presented in a white, sans-serif font on a dark blue rectangular background.

차량용 임베디드 애플리케이션 개발 조정

애플리케이션 제공 및 모니터링

개요

차량용 소프트웨어 애플리케이션은 각 차량 버전에 장착된 특정 기능과 하드웨어에 맞게 구성 및 보정되어야 하며, 이로 인해 수백 가지 버전이 파생될 수 있습니다. 설상가상으로 OEM은 이러한 애플리케이션을 차량 제조 시, 그리고 차량을 판매한 후에도 추적하고 유지관리해야 합니다. 소프트웨어 애플리케이션 개발 조정을 위한 통합 플랫폼을 활용하면 전체 차량 버전에 정확한 고품질의 애플리케이션을 효과적으로 제공할 수 있습니다. 더불어 사후 애플리케이션을 모니터링도 할 수 있습니다.

Piyush Karkare
Siemens Digital Industries Software

소개

차량 전기화, 커넥티비티, 자동화 및 공유 모빌리티는 첨단 운전자 보조 시스템 (ADAS), 배터리 관리, V2X (Vehicle-to-Everything) 통신 등과 같은 기능을 구현하기 위한 매우 정교한 소프트웨어의 필요성을 부각시킵니다 (그림 1). 이러한 각 기능은 복잡한 소프트웨어 애플리케이션의 조합이 필요하며, 이로 인해 차량 내 소프트웨어 복잡성이 크게 증가합니다. 최신 차량 소프트웨어 콘텐츠는 보통 1억 5천만 개 이상의 코드 라인으로 이뤄져 있습니다.

복잡성 증가 외에도 자율 주행, 커넥티비티, 전기화 및 공유 모빌리티 (ACES)라는 자동차 산업의 메가 트렌드로 인해 소프트웨어와 전기 및 전자 아키텍처가 변화하고 있습니다. 이러한 아키텍처는 더 적은 수의 도메인 제어 장치 (DCU)와 표준화된 기본 소프트웨어 계층이 더 많은 차량 기능을 관리하며 중앙 집중화되고 있습니다. 표준 소프트웨어 플랫폼을 구축하면 OEM이 제품 포트폴리오 전체에서 소프트웨어 기능을 확장하는 데 도움이 될 뿐만 아니라, 또 다른 복잡성 계층을 더해 차량 소프트웨어 빌드를 개발 및 제공할 수 있습니다. 자동차 소프트웨어 엔지니어는 최종 소비자에게 간단하고 직관적인 사용자 경험을 제공하기 위해 이러한 복잡성을 관리해야 합니다.

오늘날 자동차 시장의 혁신은 운송 기술을 새롭게 구상하는 대신 독립적인 개인 모빌리티의 재해석에 초점을 맞추고 있습니다. 소비자는 자유롭고 폭넓게 선택할 수 있는 이동 수단 옵션을 요구합니다. 이는 오늘날 차량 커스터마이제션 옵션 증가로 이어졌으며, 미래에는 대응력 있고 유연한 모빌리티 시스템의 개발로 이어질 것입니다. 자동차 OEM은 이미 광범위한 옵션 기능 카탈로그를 통해 차량 커스터마이제션 옵션을 확장하고 있습니다. 자율 주행, 커넥티비티 및 공유 모빌리티의 트렌드가 확산되면서 제조업체는 소비자의 요구를 충족하기 위해 다양한 차량 구성을 생산해야 합니다. 이러한 각 차량 빌드를 지원하려면 소프트웨어 구성을 개발해야 합니다. 소프트웨어 엔지니어는 최소한의 애플리케이션 구성으로 차량 빌드를 극대화하기 위해 소프트웨어 구성을 최적화해야 합니다.

고객이 직접 사용하는 기능은 대부분 전기, 전자 및 소프트웨어 관련입니다. 여기에는 보통 첨단 운전자 지원 시스템 (ADAS), 업그레이드된 인포테인먼트 시스템, 구성 가능한 실내 주변 조명, 외부 조명 및 모바일 장치 연결이 포함됩니다. 이러한 기능은 차량 제작에 맞게 구성해 볼륨 마진을 극대화할 수 있습니다. 그러나 엔진 제어, 배터리

관리, 토크 관리, 제동, 조향 같은 기본적인 차량 시스템은 주로 소프트웨어 제어 및 전자 장치를 사용해 운전자의 입력을 실행으로 전환합니다. 이로 인해 차량 구성마다 차량에 장착된 기능과 이러한 기능을 수행하는데 사용되는 시스템 아키텍처의 영향을 받는 소프트웨어의 요구사항이 제각각입니다.



그림 1: 자율 주행, 커넥티비티, 전기화, 공유 모빌리티라는 메가 트렌드로 인해 자동차 시장 내 소프트웨어 및 전자 제품의 비중이 높아지고 있습니다.

소프트웨어 엔지니어는 개별 애플리케이션에 요구되는 정교함을 처리하면서 각 차량 버전에 맞게 전체 소프트웨어 빌드뿐 아니라 각 애플리케이션의 구성 및 보정도 관리해야 합니다. 또한 각 소프트웨어 구성은 해당 버전에 맞게 소프트웨어를 올바르게 보정하고 올바른 차량 빌드에 배포되어야 합니다.

소프트웨어 아키텍처가 중앙 집중화되면서 애플리케이션 구성과 보정 프로세스도 달라지고 있습니다. 소프트웨어가 하드웨어에서 분리되며 애플리케이션 구성은 하드웨어 및 소프트웨어 플랫폼 구축 조건에 따라 달라집니다. 소프트웨어 기능으로 애플리케이션을 제공하면서 하드웨어 통합에 대한 요구가 소프트웨어 복잡성으로 전환됩니다. 이러한 애플리케이션은 여러 ECU 추상화로 포팅할 수 있으며, 특정 맞춤형 차량 빌드에 맞게 구성할 수 있습니다. 애플리케이션과 하드웨어 간 기본 계층 소프트웨어 또는 펌웨어 계층을 사용하면

차량 플랫폼 또는 OEM 라인업 전체에 애플리케이션을 배포하는 방법과 위치를 추적하기가 더욱 어려워질 수 있습니다.

그러나 각 버전에 대한 애플리케이션 및 소프트웨어 빌드를 구성하고 배포하는 것만이 문제는 아닙니다. 주행 중인 차량에서 잠재적인 버그나 문제가 발생할 수 있으므로, OEM이 최대한 빨리 문제를 식별하고 해결해야 합니다. 소프트웨어 애플리케이션의 복잡성이 증가하고 여러 구성이 필요해지면서 차량용 임베디드 애플리케이션을 모니터링하는 것이 또 다른 과제가 되었습니다. 문제가 발생하면 OEM은 차량 버전에 미치는 결함의 잠재적 영향을 최대한 빨리 파악하고, 영향을 받는 버전에 구현할 솔루션을 개발할 수 있어야 합니다. 이는 무선 업데이트 및 차량 어셈블리 빌드의 경우에도 마찬가지로 복잡합니다.

애플리케이션 제공 및 모니터링의 주요 과제

이러한 특정 작업 세트는 차량 구조에 대한 최종 결과물을 처리하기 때문에 가장 중요합니다 (그림 2).

주요 과제

이 임베디드 소프트웨어 개발 단계에서 발생하는 일반적인 문제는 다음과 같습니다.

- **막바지 변경.** 이러한 변경 사항에 대한 제어를 유지하고, 시스템 수준의 변경 필요성을 완벽히 추적하는 것은 매우 중요하지만 간과되는 경우가 많습니다. 변경이 늦어지면 팀이 작업 기한에 맞추느라 서두르는 과정에서 미처 확인하지 못하는 새로운 결함이 발생할 수 있으며, 이는 결국 작업 범위 변동, 지연 및 비용 부담이 큰 다운스트림 재작업으로 이어집니다.
- **도메인 간 변경.** 막바지 변경은 소프트웨어만의 문제가 아닙니다. 전기 시스템, 기계 시스템 및 기타 시스템은 최종 검토와 승인이 완료된 후에 변경해야 하는 경우도 있습니다. 완전히 커밋돼야 하는 고정된 데이터도 변경됩니다. 이러한 도메인 간 변경을 면밀히 주시하고, 소프트웨어 제공을 위해 함께 업데이트하는 것은 추가 위험 가능성을 상당히 증가시키는 매우 복잡한 작업입니다. 모든 엔지니어링 팀은 변경, 시스템 수준 하향으로 인한 변경의 영향, 백업을 명확히 파악해야 합니다.

- **애플리케이션 구성 관리에 대한 문제입니다.** 차량용 플랫폼은 고유한 공유 기능, 구성요소 및 임베디드 하드웨어가 혼합된 수십 개의 개별 차량 빌드를 생성합니다. 소프트웨어 도메인은 이러한 각 차량 빌드와 일치하도록 소프트웨어 빌드를 구성해야 하므로 복잡성이 크게 증가합니다. 소프트웨어 엔지니어는 각 차량에 제공되는 소프트웨어가 기능적 관점에서 완전함은 물론, 차량 빌드의 하드웨어와 완전히 호환되는지도 확인해야 합니다. 교정 및 부트 로더가 일치하는 올바른 구성을 제공해야 합니다. 이는 OTA 업데이트를 통해 현장에서 고객 차량을 업데이트할 때 특히 중요합니다.

또한 호환 가능한 애플리케이션을 적시에 높은 수준으로 제공할 수 있도록 엔지니어는 계획에서 테스트 및 문서화에 이르는 테스트 및 품질 보증 프로세스를 조정하고 후반 및 교차 도메인 변경을 관리하며, 이 데이터를 취합해 최종 차량 BOM을 공급할 수 있는 강력한 임베디드 애플리케이션 프로그램 개발 플랫폼을 필요로 합니다.

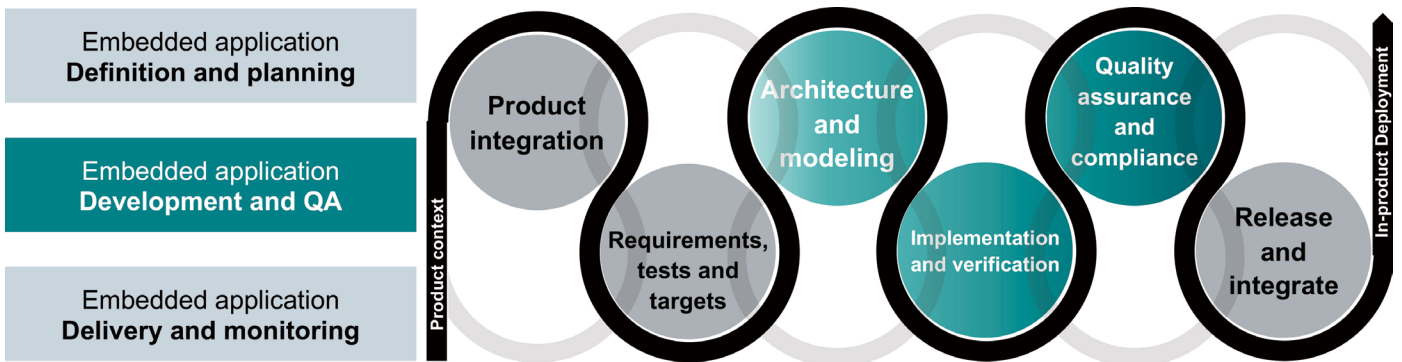


그림 2: 애플리케이션 제공 및 모니터링에는 품질 보증에서 변경 사항 구현, 최종 애플리케이션 바이너리 제공 및 모니터링, 소프트웨어 빌드 및 차량까지 포함됩니다.

통합 플랫폼을 이용한 임베디드 애플리케이션 제공 및 모니터링

최신 임베디드 애플리케이션 엔지니어링 플랫폼 솔루션을 사용하면 애플리케이션을 올바른 차량 빌드 BOM (설계, 릴리스, 구축 기준)으로 조정하고, 엔지니어는 릴리스 후 애플리케이션을 모니터링할 수 있습니다 (구축 및 서비스 기준 BOM). 이러한 솔루션을 통해 적절한 시스템 또는 제품 버전 구축조건 맥락에서 애플리케이션 빌드의 검증 및 확인을 조정할 수 있습니다. 여기에는 각 제품 버전에 의해 생성된 차량 하드웨어 및 구성 요소 생태계를 파악하고, 모든 관련 하드웨어 생태계에서 애플리케이션을 테스트하는 작업이 포함됩니다.

이를 통해 엔지니어는 모든 하드웨어 및 시스템 구축조건을 충족하는 완전히 검증되고 확인된 빌드를 배포할 수 있습니다. 검증 작업의 전체 기록을 유지하므로 컴플라이언스 및 제품 책임 비용도 줄어듭니다.

구현 및 검증

애플리케이션 개발 및 품질 보증 프로세스 중에 애플리케이션 코드가 지속적으로 생성, 테스트, 업데이트 및 다시 테스트되어 애플리케이션이 설계된 대로 작동하는지 확인하고 설계가 올바른지 검증합니다. 애플리케이션 제공 및 모니터링 작업 맥락에서 엔지니어는 애플리케이션이 차량에 제공할 준비가 되도록 코드를 완성하고 커밋하는 데 주력합니다 (그림 3). 소프트웨어 테스트는 주로 MiL (Model-in-the-Loop) 및 SiL (Software-in-the-Loop) 테스트에서 HiL (Hardware-in-the-Loop) 및 ViL (Vehicle-in-the-Loop)로 전환됩니다. 이를 통해 가상(HiL) 및 물리적 (ViL) 하드웨어를 사용하여 애플리케이션을 검증하고 실제 시험 트랙에서 전체 차량까지도 검증합니다.

엔지니어는 구현이 차량 하드웨어 및 시스템 구축조건과 일치하는지 지속적으로 확인해야 합니다. 다른 관련 애플리케이션의 변경 사항도 고려해야 합니다. 지속적으로 시스템과 하드웨어의 구축조건 및 요구사항에 맞게 조정하는 것은 코드 변경 사항을 식별하고 적용할 때 특히 어렵습니다. 감사 가능한 변경 관리 프로세스는 변경 사항을 제어하고 애플리케이션 구축조건 및 요구사항을

완전히 추적하는 데 중요합니다. 또한 이러한 프로세스는 MiL, SiL, HiL 및 ViL 테스트에 따라 코드를 업데이트할 수 있는 충분한 유연성을 지원해야 합니다.

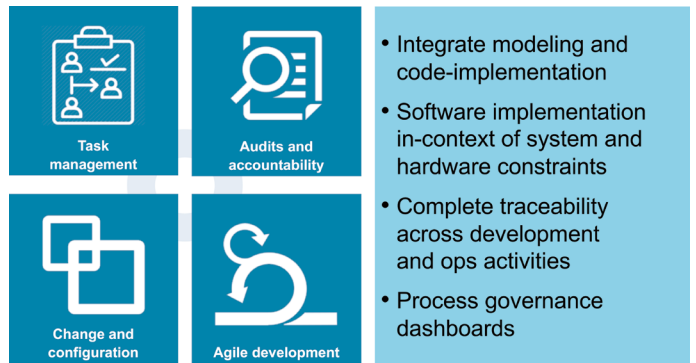


그림 3: 엔지니어는 코드 완성 과정에서 애플리케이션 기능을 지속적으로 검증 및 확인해야 합니다.

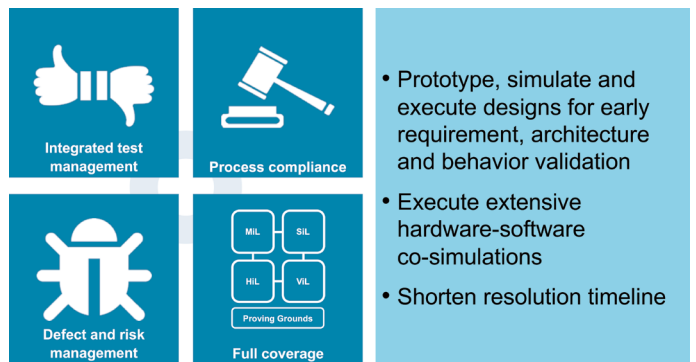


그림 4: 품질 보증을 통해 시스템 하드웨어 맥락에서 임베디드 애플리케이션을 검증 및 확인합니다.

품질 보증

품질 보증 및 컴플라이언스 인증은 보통 개발 과정 전반에서 이어지는 프로세스입니다. 품질 보증의 목표는 시스템 하드웨어 맥락에서 애플리케이션을 검증 (설계된 대로 작동하는지) 및 확인 (작동하도록 설계됐는지)하는 것입니다 (그림 4). 소프트웨어 엔지니어는 설계 요구사항과 아키텍처, 모델이 생성되는 동안 테스트

케이스, 테스트 계획 및 테스트 벡터를 정의합니다. 엔지니어의 과제는 이 모든 데이터를 다양한 추상화 수준에서 관련 작업 항목으로 조정하는 것입니다. 소프트웨어는 개별 기능부터 차량 수준 기능까지 모든 추상화 수준에서 테스트해야 합니다. 또한 테스트가 실행될 때 항목이 필요에 따라 수행되고 있는지, 결함이 있고 수정이 필요함을 표시하기 위해 결과를 관련 작업 항목과 연관시켜야 합니다.

하지만 엔지니어가 소프트웨어 아키텍처와 모델, 실제 코드를 개발하고 테스트하는 과정에서 새로운 데이터는 계속해서 생성됩니다. 테스트 라운드마다 소프트웨어 엔지니어는 테스트 케이스, 테스트 실행 전략, 테스트 벡터 및 각 작업 항목의 기타 요소를 상세하게 기술하는 테스트 계획을 실행해야 합니다. 테스트가 완료되면 문제를 (잠재적으로 특정 조건에 따라 자동 또는 반자동으로) 기록하고 시험 결과를 관련 작업 항목에 다시 연결해 해당 문제를 해결해야 합니다. 이러한 데이터 연결은 테스트 커버리지를 보여 주며, 요구사항에서 구현에 이르는 완전한 추적성을 지원하는 데 매우 중요합니다.

이 데이터는 추적이 필요하고 해당 소프트웨어 빌드 또는 구성에 연결되어야 합니다. 그간 엔지니어는 이러한 복잡한 데이터 관계를 수동으로 유지했고, 이로 인해 엔지니어링 작업에 시간과 노력을 집중할 수 없었습니다.

애플리케이션 릴리스 및 통합

애플리케이션을 릴리스하려면 소프트웨어 엔지니어가 특정 제품 BOM에 맞게 애플리케이션 빌드를 조정해야 합니다 (그림 5). 가장 중요한 것은 각 애플리케이션 빌드가 배포될 차량이나 시스템 버전의 맥락에서 준비되어야 한다는 점입니다. 이를 위해 엔지니어는 차량 버전의 특정 하드웨어 및 시스템 구속조건 하에서 애플리케이션을 완전히 검증 및 확인해야 합니다. 이는 잘못된 소프트웨어 테스트 및 업데이트를 비롯하여 버전 하드웨어 또는 시스템 구성의 변경으로 인해 매우 어려워집니다.

릴리스 시 각 애플리케이션은 해당 애플리케이션이 설치된 차량의 구축 및 서비스된 BOM을 추적할 수 있어야 합니다. 이러한 추적성은 고객 소유 차량이 현장에서 사용될 때 애플리케이션을 추적하는 데 중요하므로, 애플리케이션을 차량 수명 중 최신 상태로 유지해야 합니다.

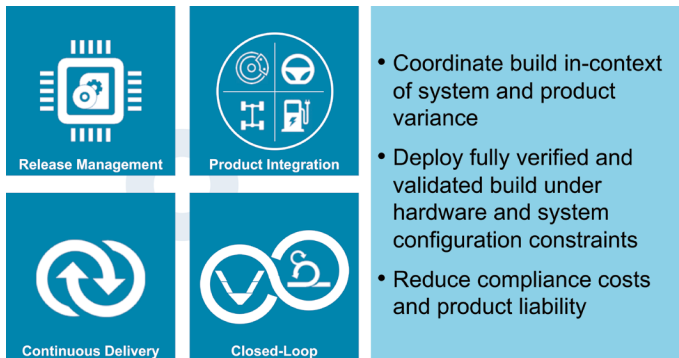


그림 5: 애플리케이션을 릴리스하고 통합하려면 애플리케이션 빌드가 특정 제품 BOM과 일치해야 합니다.

결론

자동차 애플리케이션 개발은 초기 정의 및 계획 단계부터 올바른 구성으로 올바른 차량에 양질의 애플리케이션을 제공하는 마지막 난관에 이르는 다면적이며 까다로운 프로세스입니다. 이 과정에는 여러 도메인 팀과 관계자 간의 협업이 필요합니다. 차량용 임베디드 애플리케이션 소프트웨어 개발을 위한 통합 디지털 플랫폼은 소프트웨어, 하드웨어 및 시스템 엔지니어링을 비롯한 여러 도메인에 걸친 협업과 가시성을 촉진합니다. 또한 기업은 이러한 플랫폼을 사용해 검증된 소프트웨어 구성요소를 재사용 해 새롭고 다양한 애플리케이션 개발을 가속화할 수 있습니다.

애플리케이션 개발, 제공 및 모니터링 전반에 걸쳐 임베디드 애플리케이션 개발 플랫폼을 활용해 다양한 도구와 연결하고 코드 성능 커버리지 데이터를 제공하며, 메소드 및 코딩 표준 (예: MISRA-C)을 준수할 수 있습니다.

소프트웨어 엔지니어링 플랫폼에서 제공하는 포괄적인 추적 기능을 통해 모니터링 및 업데이트가 가능한 페루프 시스템을 구현합니다. 소프트웨어 엔지니어가 배포된 소프트웨어를 모니터링할 때 제품 또는 차량 수준에서 개념 수준 요구사항, 사양 및 기타 데이터 아티팩트에 이르는 추적 기능을 활용해 신속히 재설계하고 대응할 수 있습니다. 이러한 대응은 현장에서 발생하는 문제를 해결하고 소비자 안전을 강화하며, 결함 차량으로 인한 A/S 비용을 줄이거나 없애는 데 매우 중요합니다.

자동차는 점점 전기화될 뿐만 아니라 디지털화되어 갑니다. 소비자에게 가장 중요한 차량 기능은 점차 소프트웨어를 통해 제어하는 비중이 높아질 것입니다. 더 연결된, 지능적 차량에 대한 소비자의 요구가 늘어나는 가운데, OEM은 소프트웨어 개발 프로세스를 발전시켜야 합니다. 전체 애플리케이션 개발 프로세스를 취합하고 협업하며 추적 및 제어하는 단일 플랫폼을 통한 혁신으로 이 새로운 시장에서 경쟁 우위를 확보할 수 있습니다.

Siemens 솔루션이 귀사에 어떤 도움을 줄 수 있는지 자세히 알아보려면 [siemens.com/aes](https://www.siemens.com/aes)에서 블로그, 백서, 팟캐스트, 제품 동영상, 웨비나, 솔루션 기능, 인포그래픽 등을 확인해 보십시오.

Siemens Digital Industries Software

본사

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

미주 지역

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

유럽 지역

Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

아태 지역

Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

Siemens Digital Industries Software 소개

Siemens Digital Industries Software는 엔지니어링, 제조 및 전자 설계가 미래와 만나는 디지털 엔터프라이즈를 실현하기 위한 혁신에 박차를 가하고 있습니다. Siemens Digital Industries Software의 솔루션은 규모를 막론한 기업이 조직에 새로운 인사이트와 기회, 혁신을 촉진할 자동화 수준을 제공하는 포괄적 디지털 트윈을 생성할 수 있도록 지원합니다. Siemens Digital Industries Software 제품과 서비스에 대한 자세한 사항은 sw.siemens.com 를 방문하시거나 [LinkedIn](#), [Twitter](#), [Facebook](#) 및 [Instagram](#) 계정 팔로우를 통해 확인하실 수 있습니다. Siemens Digital Industries Software – Where today meets tomorrow.

[siemens.com/software](https://www.siemens.com/software)

© 2019 Siemens. 관련 Siemens 상표 목록은 [여기](#)서 확인할 수 있습니다. 기타 모든 상표는 해당 소유자에 귀속됩니다.

81257-83224-C1-KO 12/20 LOC