EULYNX 컨셉

1. 서론
   1. 릴리스 정보
   2. 인프린섬
   3. 목적 이 문서는 EULYNX의 개념과 범위를 제시합니다. 표준화의 근거, EULYNX의 목표를 설명하고 EULYNX가 정의한 모듈식 시스템 개념을 소개합니다.
   4. 적용 표준 및 규정 EULYNX에서 사용되는 적용 가능한 표준 및 규정 목록은 EULYNX 참조 문서 목록[Eu.Doc.12]에 나와 있습니다.
   5. 적용 문서

이 문서에 입력으로 사용되거나 관련된 문서의 현재 버전은 EULYNX 문서화 계획[Eu.Doc.11]에 나열되어 있습니다. 문서 간의 관계는 부록 A1 문서화 계획 및 구조[Eu.Doc.11\_A1]에 표시됩니다.

1.6 용어 및 약어 용어와 약어는 EULYNX 용어집[Eu.Doc.9]에 나열되어 있습니다.

1.7 변동성 관리

이 문서는 전체 EULYNX 시스템에 유효합니다. 이 문서에서는 가변성 관리를 사용하지 않습니다.

1.8 객체 유형의 정의 이 문서에서는 개체 유형에 대한 다음 정의가 적용됩니다.

* "Req": 필수 요구 사항을 나타냅니다.
* "정보": 이는 사양을 이해하는데 도움이 되는 추가 정보를 나타냅니다. 이러한 객체는 추가 요구 사항을 지정하지 않습니다.
* "머리말" ‑ 이것은 장의 제목을 나타냅니다.

2 EULYNX의 근거

2.1 목표

* EULYNX의 주된 목표는 신호 시스템의 기술 인터페이스를 표준화하여 시장을 개방하고, 혁신을 가속화하고, 규모의 경제 효과를 얻는 것입니다.
* EULYNX는 표준화 목표를 달성하기 위해 인프라 관리자 간의 긴밀한 협력을 위한 프레임워크를 제공합니다.

2.2 표준화의 이점 및 제약

특히 유럽 수준에서 기술 시스템의 표준화는 상호 운용성을 관리하고, 효율성을 개선하고, 따라서 전체 생태계의 비용을 줄이는 가장 강력한 조치입니다. 신호 시스템의 경우 이 표준화는 다양한 국가 운영 규칙, 상업적 이익, 언어 및 기타 차이점을 고려합니다.

2.3 라이프 사이클 접근 방식

(INFO)

* Lifecycle 비용 목표와 공유 시장 접근 방식은 유럽 인프라 관리자(IM)의 목표입니다.
* 개별 IM의 요구 사항을 유럽의 조화된 요구 사항으로 수렴하는 동시에 기술 시스템을 경쟁적인 방식으로 변경, 유지 관리, 갱신 및 업데이트해야 합니다.
* 이를 통해 인프라 관리자는 시스템 통합자로서 시스템 Lifecycle동안 다양한 하위 시스템에 대한 다양한 공급업체를 선택할 수 있는 위치에 놓이게 됩니다.
* 이러한 접근 방식은 새로운 프로젝트에서 또는 기존 시스템 기능이나 인프라 레이아웃을 수정할때 따라야 합니다. 또한 Maintence 관련 활동도 이로부터 이익을 얻을 것입니다.
* IM은 공급업체 간의 경쟁을 개선하고 신호 시스템 혁신을 가속화하기 위해 시장 세력에 합류하여 Lifecycle 비용을 줄이는 것을 목표로 합니다.

(REQ)

각 하위 시스템의 수명이 다르므로 개별적으로 교체가 가능해야 합니다.

(INFO) 연동 시스템 표준화(예: Euro‑Interlocking, INESS 및 ERTMS)에 관한 이전 유럽 이니셔티브의 결과는 기반을 제공합니다. 또한 여러 시장에서 결과를 재사용할 수 있으므로 공급 산업에 기회를 제공합니다. 이는 모든 관련자에게 Win-Win 상황을 만들어냅니다

2.4 사양 방법

(INFO) 역사적인 이유로 많은 레거시 신호 시스템 간의 차이가 존재합니다. 그 결과, 운영 프로세스와 기술 장비와의 호환성에 직접적인 영향을 주지 않고는 서로다른 요구사항들을 조합롭게 통합하는 것이 종종 불가능해 보입니다. 따라서 표준화를 위한 전용 방법론이 정의됩니다. 참여자의 의견을 수집, 구조화, 분류합니다. 해당 분야의 전문가들은 시스템 아키텍처와 시스템 요구 사항을 제공하여 디지털화를 통한 비용 절감과 기존 제어‑명령 및 신호 하위 시스템과의 호환성이라는 목표를 공통 검토 및 검증을 통해 달성할 수 있습니다. 장기적으로 더 높은 수준의 표준화가 등장하여 신호 시스템의 Lifecycle 비용을 크게 줄일 수 있습니다. EULYNX는 MBSE(Model Based System Engineering)를 적용하고 있어, 명세서 품질 측면에서 선도적인 역할을 하고 있습니다.

2.5 EULYNX 대화 철도 부문은 이 아키텍처의 개선 및 진화와 EULYNX 웹사이트에 공개적으로 게시된 모든 EULYNX 표준에 적극적으로 기여할 수 있습니다. EULYNX는 이러한 대화를 적극적으로 촉진합니다.

3. EULYNX의 범위

* EULYNX는 제어 명령 및 신호 하위 시스템의 일반적인 Reference 아키텍처를 제공합니다.
* 이 Reference 아키텍처는 인프라 관리자와 유럽의 유사한 프로젝트에 참여하여 특정 공급업체와 협의하여 지정됩니다. 이 아키텍처에서는 시스템이 의도한 기능, 즉 하위 시스템 간의 데이터 교환을 달성하는 것을 목표로 프로세스를 고려합니다. 필요한 경우 정의에는 설계에 구현될 하위 시스템(하드웨어, 소프트웨어, 시설, ao)도 포함됩니다. 범위에는 보안도 포함됩니다.

(REQ) Reference 아키텍처는 [EN 50126]에 따라 시스템의 전체 Lifecycle에 적용됩니다.

(INFO) Reference 아키텍처는 이해관계자들의 요구사항과 관심사를 다룹니다:

* 유럽의 기반시설 관리자(Infrastructure Managers),
* 신호 시스템 및 하위 시스템 공급자,
* 열차 운영사(Train Operators),
* 안전 관련 당국(Safety Authorities),
* 유럽 철도 기관(European Railway Agency),
* 기타(사용자, 지정기관, 독립 안전 평가 기관, 엔지니어링 사무소, 추가 표준화 기구, 계약자 등).

4 시스템 개념

4.1 Reference Architecture의 전략

(INFO) Reference Architecture는 공급자들의 개발 및 미래 솔루션을 지원하여 유럽의 미래 지향적인 철도 인프라를 달성할 수 있도록 해야 합니다.

(REQ) EULYNX는 철도 시스템의 최상위 계층이 중앙 집중식 교통 제어 계층으로 구성되어 있으며, 이 계층이 중앙에서 운영과 용량을 제어한다고 가정합니다. 그 아래 계층에서는 연동 시스템(interlocking systems)과 무선 블록 센터(radio block centres)가 열차 경로의 잠금 및 해제를 안전하게 수행합니다. 가장 하위 계층에서는 선로 측 객체 (wayside objects)가 이 작업에 기여하며, 이는 포인트(분기기), 신호기, 건널목, 가동 교량 등을 포함합니다.

(REQ) 모듈식 개념 내에서 정보와 에너지 공급의 분리는 기본적인 패러다임입니다. 분산 에너지 공급을 위한 IP 네트워크와 "power-bus"

는 분산 시스템에서 정보 공급을 위한 "서비스 지점"과 에너지 공급을 위한 "파워 지점"을 결정합니다.

(REQ) Reference 아키텍처는 폐쇄형 및 개방형 네트워크를 사용하여 IP 기반 통신을 허용합니다.

(REQ) Reference 아키텍처는 다른 산업에서도 사용되는 기성품(COTS) 및 대량 산업 솔루션을 기반으로 하는 시스템 설계를 지원합니다.

(REQ) 현대적인 아키텍처와 표준화된 진단 시스템을 제공함으로써 유지 관리 효율성과 시스템 최적화가 개선됩니다.

(REQ) EULYNX는 플러그 앤 플레이 구성 요소를 신호 시스템에 통합하는 것을 목표로 하는 표준 인터페이스와 정의된 기능 범주를 제공합니다. 소프트웨어 및 하드웨어 아키텍처를 포함한 매우 자세한 구현 처방은 이 표준의 범위에 포함되지 않습니다. 이는 가변성 관리 또는 제조업체가 다양한 설계 솔루션을 적용하려는 의도된 유연성과 충돌할 수 있기 때문입니다.

(REQ) EULYNX는 다음과 같은 표준화 목표를 우선순위에 따라 고려합니다

* 인터페이스에서 볼 수 있는 기능적 요구사항
* 인터페이스의 기술적 요구사항, 텔레그램 구조
* 하위 시스템 및 구성 데이터의 식별
* 유지보수 및 진단에 대한 요구사항
* 하위 시스템의 물리적 요구사항
* 환경 요구사항
* 전원 공급

(REQ) 표준화 목표는 각 기반시설 관리자(IM)마다 가능한 시나리오에 따라 다를 수 있습니다:

* 하드웨어(객체 제어기)의 교체가 필요하지 않은 경우:  
  이는 주로 인접 시스템(RBC, CC, LX) 간의 인터페이스에 적용됩니다.
* 하드웨어(객체 제어기) 교체가 필요한 경우:  
  시스템 라이프사이클 동안 다양한 하위 시스템에 대해 여러 공급자를 선택할 수 있도록 하기 위한 목적입니다. 이 시나리오는 시스템 기능 수정 및 레이아웃 변경, 시스템의 부분적 갱신에 적용됩니다.
* 하드웨어(객체 제어기) 교체가 필요하며 유지보수 목적도 포함하는 경우:  
  이 시나리오는 시스템 라이프사이클 동안 다양한 하위 시스템에 대해 여러 공급자를 선택할 수 있도록 하는 것뿐만 아니라, 유지보수 목적도 포함합니다. 이는 수정 및 부분적 갱신을 지원합니다. 또한, 유지보수 관련 활동을 추가로 지원하기 위해, 이 시나리오는 공급자에 관계없이 개별 객체 제어기를 교체할 수 있도록 하여, 예를 들어 고장 난 객체 제어기를 다른 공급자의 제품으로 교체하는 등의 수정 작업을 가능하게 합니다.

4.2 모듈형 시스템 개념

(INFO) EULYNX Reference 아키텍처는 모듈식 시스템을 정의하며, 이는 하위 시스템과 표준화 및 명세화 될 인터페이스로 구성됩니다.

(REQ) 모듈식 구조는 변동성 관리(variability management)를 적용하여 서로 보완적이거나 상충할 수 있는 IM(기반시설 관리자)별 요구사항을 관리할 수 있도록 정의되어야 합니다.

(REQ) 모듈은 전적으로 EULYNX가 제공하는 인터페이스 및 기능 명세서에 기반하여 신호 시스템에 통합될 수 있도록 설계되어야 합니다.

(REQ) EULYNX Reference 아키텍처의 모듈식 구조는 분기기(points) 및 신호기(signals)와 같은 선로 측 객체(wayside objects)를 하위 시스템으로 정의하며, 이들은 표준화된 인터페이스를 통해 선로 측 객체를 명령하고 제어하는 컨트롤러를 포함합니다.

(REQ) 컨트롤러를 포함한 하위 시스템은 다음 목표를 지원하기 위해 충분한 지능(intelligence)을 갖추어야 합니다:

* 연동 시스템(interlocking system) 코어와 객체 컨트롤러(object controllers) 간의 기능 분배를 통해 조화를 이루고 유럽 전역에서의 구현을 가능하게 합니다. (강력한 코어만으로는 조화를 이룰 수 없습니다.)
* 보안 관련 측면을 효과적으로 관리합니다.
* 지능형 객체 컨트롤러를 통해 예방적 유지보수를 지원합니다.

(REQ) 인프라 관리자(IM)별로 추가적인 요구사항이나 상충되는 요구사항이 있을 경우, 이를 개별적인 구성 방식으로 관리해야 한다.

(REQ) EULYNX 아키텍처의 각 하위 시스템은 통신이 중단된 경우에도 안전한 방식으로 반응할 수 있어야 한다."

4.3 Reference Architecture

4.3.1 일반

(INFO)EULYNX 명세서는 필수적으로 지켜야 할 인터페이스 요구사항과, 도움은 되지만 선택적으로 사용할 수 있는 정보 객체로 구성됩니다.

(INFO) 다양한 유형의 아키텍처가 설명된다.

* 식별된 인터페이스를 포함한 하드웨어 및 네트워크 구조
* 식별된 인터페이스를 포함한 시스템 기능 및 논리적 동작 구조
* 성능, 보안, 신뢰성 등 기능 외적인 요구사항을 정의하는 구조
* 실제 개발 및 배포를 위한 소프트웨어 및 하드웨어 구성 방식

(INFO) 한 아키텍처 유형을 변경하면 다른 하나 이상의 아키텍처 유형에 영향을 미칠 수 있습니다.

(INFO) EULYNX 표준화는 시스템의 기본적인 운영 방식을 정의하지만, 세부적인 운영 개념이나 전체적인 통합은 다루지 않습니다. 대신, 시스템이 어떻게 동작할지, 어디에 배치될지, 운영 시간 및 성능 요구 사항에 대한 기본적인 방향을 제시합니다.

4.3.2 물리적 아키텍처

(INFO) 국가 전체 네트워크에 걸쳐 분포된 많은 하위 시스템이 중앙 집중식. 하위 시스템과 결합됩니다. 물리적 아키텍처는 이러한 하위 시스템을 표현하며, 이는 시스템을 구성하는 노드(node), 그들의 연결성(connectivity), 그리고 링크(links)로 나타냅니다.

(INFO) 물리적 아키텍처는 특정 요구사항 집합을 충족하는 적합한 시스템을 위해 필요한 부분 또는 요소의 배열, 상호 작용 및 상호 의존성을 관리하는 최소한의 규칙 집합을 제공합니다.

(INFO) 물리적 아키텍처를 기반으로 엔지니어링 명세를 도출할 수 있으며, 기존 인프라에 연결할 수 있어야 합니다.

4.3.3 기능적 아키텍처

(INFO) EULYNX 시스템의 기능적 아키텍처는 각 서브시스템과 인접 시스템에 어떤 기능을 할당할지에 대한 표준을 정의하고, 이와 관련된 인터페이스는 'EULYNX 시스템 정의[Eu.Doc.7]' 문서에 있습니다.

4.3.4 비기능적 아키텍처

(INFO) 표준 비기능적 측면은 둘 이상의 하위 시스템 또는 인터페이스에서 역할을 하는 요소들로, 다음과 같은 표준 요구사항을 포함합니다:

* RAMS(신뢰성, 가용성, 유지보수성, 안전성)
* 성능(Performance)
* 모니터링(Monitoring)
* IP 기반 통신(IP-based Communication)
* 보안(Security)

4.3.5 구현 아키텍처(Implementation architecture)

(INFO) 구현 아키텍처는 시운전, 유지보수, 철거를 포함합니다.

구현 아키텍처에는 시운전, 유지관리 및 해체가 포함됩니다.

(INFO) 계획을 고려할 때, 구현은 다음과 같이 나눌 수 있습니다:

* 데이터 준비(Data Preparation): 신호 시스템의 특정 애플리케이션을 위한 계획 및 설계와 관련됩니다.
* 안전 보증 요구사항(Safety Assurance Requirements): 안전 사례(safety cases), 검증(verification), 확인(validation)을 포함합니다.
* 롤 아웃 계획(Roll Out Plan):
* 무엇을, 어디서, 언제 구현할지에 대한 일반적인 설명으로, 정부, 대중, 언론이 이해할 수 있는 수준으로 작성됩니다.
* 각 기반시설 관리자의 책임 하에 있습니다.
* 마이그레이션(Migration):
* 무엇을, 어디서, 언제 구현할지에 대한 상세 설명으로, 어느 기능이 어디서, 언제 구현될지 이해할 수 있도록 합니다.
* 종종 여러 단계, 상태 전이, 다양한 단계 이름을 포함하며, 전문가 수준에서 작성됩니다.
* 각 기반시설 관리자의 책임 하에 있습니다.
* 통합(Integration):
  + 무엇을, 어디서, 언제 구현할지에 대한 설명으로, 기술적, 기능적, 운영적 문제가 어떻게 조율되는지 다룹니다.
  + 각 기반시설 관리자의 책임 하에 있습니다.

5 시스템 개발 프로세스

(INFO) 인프라 관리자는 시스템 개발 프로세스의 1~4단계와 9~10단계에서 주요 책임을 집니다. 또한 5단계에서는 인프라 관리자와 산업이 공동으로 책임을 집니다. “[EN 50126참고]”

(INFO) EULYNX의 시스템 개발은 시스템 엔지니어링 프로세스 [Eu.Doc.27] 문서에 자세히 정의되어 있습니다.

(INFO) 인프라 관리자의 책임은 다음을 포함합니다:

* + 요구사항 관리 (requirements management)
  + 형식적 모델링 (formal modelling)
  + 도구 선택 (tooling selection)
  + 구성 관리 (configuration management)
  + 변동성 관리 (variability management)
  + 검증 및 유효성 검사 (verification and validation)
  + 변경 관리 (change management)

(REQ) EULYNX 인터페이스 사양의 개발은 다음과 같은 절차로 진행됩니다:

* + 관련된 모든 파트너(철도 운용사, 제조사 등)의 요구사항을 종합적으로 반영하라.
  + 그런 다음 시스템공학에서 널리 쓰이는 “Sys ML” 모델링 언어를 사용하여 인터페이스의 동적 모델(상호작용, 상태변화 등)을 생성합니다.
  + 이를 특정 소프트웨어 도구(Cameo, Enterprise Architect등)를 사용해 모델의 정확성과 실행가능성을 검증합니다.
  + Eu.Doc.7은 EULYNX의 전체 아키텍처와 하위 시스템 간 상호작용을 정의한 핵심 문서로, 모델링 시 이 표준을 엄격히 따릅니다.

요약: EULYNX 인터페이스의 사양은 파트너들로부터 수집한 요구 사항을 기반으로, SysML 모델링 언어를 사용해 동적 모델로 생성되며, 이를 통해 인터페이스가 올바르게 작동하는지 확인합니다.

(INFO) EULYNX 개발 방법 간략 설명:

* + 기능(Function): 시스템이 수행해야 할 주요 기능을 정의합니다.
  + 사용 사례(Use cases): 각 기능에 대한 구체적인 사용 사례를 설정하여, 실제 상황에서 시스템이 어떻게 동작할지 파악합니다.
  + 모델링(Modelling): 기능과 사용 사례를 바탕으로 시스템을 모델링하여, 시스템의 설계를 시각적으로 표현합니다.
  + 검증(Validation): 모델이 예상대로 작동하는지 확인하고, 시스템의 요구 사항을 충족하는지 검증합니다.

6 변동성 관리(Variability Management)

(REQ) 유럽 국가들 간의 철도 시스템에 따른 차이를 고려하여, EULYNX 시스템이 다양한 요구 사항을 충족할 수 있도록 변동성을 관리해야 한다.

(INFO) EULYNX 클러스터에 참여하는 각 인프라 관리자(Infrastructure Manager, IM) 가 해당 서브시스템 개발에 관련된 IM-specific 요구 사항을 제공한다. 이러한 요구 사항들은 국가별 출처(예: Notified National Technical Rules, IM-specific 요구 사항, 표준 등)와 추적 가능해야 하며, 그 추적의 책임은 각 인프라 관리자에게 있습니다.

(INFO) 변동성 관리(Variability management)는 EULYNX에서 어떻게 처리될지에 대해 [Eu.Doc.28] 문서에 상세히 정의되어 있다

7 데이터 준비

(REQ) 인프라 관리자들이 외부 파트너와 데이터를 쉽게 교환하고, ERTMS와 호환되는 인터락킹 시스템을 계획하는 데 필요한 표준화된 데이터 형식을 제공하는 것을 목표로 합니다.

(INFO) 각 인프라 관리자는 이미 자신만의 데이터 형식과 툴을 개발했기 때문에, 새로운 시스템에 이를 적용할 수 있도록 표준 데이터 형식으로 변환하는 과정이 필요하다는 것입니다. 이는 데이터 호환성과 시스템 통합을 원활하게 하기 위해 필수적입니다.

8. 확신

8.1 보증 프로세스

(REQ) EULYNX 보증 활동은 다음으로 구성됩니다.

* 아키텍처와 모델링 프로세스가 적절하고 완전하다는 독립적인 확인.
* 각 EULYNX 클러스터에 적용되는 일반적인 안전 보증 프로세스는 작업 프로세스와 해당 작업의 출력이 모두 완전하고 각 파트너 멤버가 사용하기에 적합하다는 것을 보여줍니다.
* 보증 계획 및 프로세스에 대한 독립적인 검토
* 보증 결과에 대한 독립적인 감사

(INFO)

* EULYNX는 Reference 구현(CSM Reference 시스템 경로[CSM])에서 이전 경험을 재사용하도록 권장합니다.
* Reference 아키텍처와 시스템 엔지니어링 프로세스는 각 클러스터 구성원의 전문가적 판단을 통해 생성되어야 합니다. 이 두 클러스터의 결과에 대한 독립적인 검토는 문서 검사와 클러스터 프로젝트 구성원의 경력 설명 검토를 통해 수행됩니다.
* 인터페이스 클러스터 프로젝트 출력의 보증은 클러스터 리더가 각 단계의 성공적인 완료를 기록할 수 있도록 하는 체크리스트를 통해 용이하게 됩니다. 각 클러스터는 프로세스의 올바른 적용을 보장하기 위해 독립적인 감사를 받게 됩니다.

8.2 EULYNX 출력

공급 산업에 대한 출력은 요구 사항 측면에서 완전하며 EULYNX 호환 인터페이스로 생성된 장비에 대한 안전 사례의 유효한 입력으로 사용될 수 있습니다

[EU.Doc.7] 기능적 측면에서 서브시스템과 인접시스템에 어떤 기능을 할당할지에 대한 표준

[Eu.Doc.9] 용어 및 약어

[Eu.Doc.11] 입력으로 사용되거나 관련된 문서의 현재 버전은

[Eu.Doc.12] 적용 가능한 표준 및 규정 목록

[EU.Doc.27] 시스템 엔지니어링 프로세스

[EU.Doc.28] 변동성 관리(Variability management)

\*이 문서는 EULYNX(eulynx.io)에서 제공하는 “20191212-EULYNX-Concept-Eu\_Doc\_6-v2\_0-2\_A.pdf”를 참고하였습니다.