

ISO TR 24529 - Intelligent transport systems - Systems architecture- Using UML in ITS Standards and deliverables

Application Area: [ITS Architecture](#)

Number of pages: 23

Zavedení normy do ČSN: převzetím originálu

Extract Creation Year: 2012

Standard Topic Group: Použití UML

Standard Topic: Architektura systémů ITS

Topic Description: Používání UML v normách a dokumentech ITS

Introduction, Explanation of Starting Points
Description of Architecture, Hierarchies, Roles, and Object Relationships
Description of Process / Function / Method of Use
Použití UML při návrhu norem pro ITS
Description of Interfaces / APIs / System Structure
Protocol / Algorithm / Computation Definition
Definition of Data Representation / Physical Meaning
Definition of Constants / Ranges / Restrictions

Introduction

Cílem této technické [zprávy](#) je poskytnout návod na použití [UML](#) při vývoji norem pro inteligentní dopravní systémy. [UML](#) poskytuje mezinárodně standardizovanou formu [modelu](#) systému, snadno interpretovatelnou kdekoli na celém světě, což umožňuje konzistentní popis z více uživatelských pohledů. Jazyk [UML](#) se velmi často používá při návrhu architektury systému nebo při návrhu a vývoji náročných softwarových systémů. Bohužel tento jazyk stále není plně chápán mnoha účastníky, kteří nejsou zároveň vývojáři softwaru, protože používá velké množství nezvyklého žargonu, nezbytného pro přesnost, ale nepochopitelného pro odborné i laické čtenáře. Existují proto určitá rizika při používání [UML](#), nicméně jeho výhody jsou obecně hodnoceny jako převažující nad nevýhodami.

Poznámka: Extrakt uvádí vybrané kapitoly popisovaného dokumentu a přejímá původní číslování kapitol.

Application

Tato technická [zpráva](#) by měla být vodítkem pro vývojáře ITS norem a systémů, kteří uvažují o využití [UML](#). Tato [zpráva](#) není normou a poskytuje spíše obecná doporučení než závazné [požadavky](#). [Zpráva](#) je navržena tak, aby poskytovala údaje a vysvětlení těm, jenž vytváří mezinárodní normy ITS a těm, kteří vytváří specifikace, implementace a instalace inteligentních dopravních systémů. Tato technická [zpráva](#) poskytuje pokyny a nevyjadřuje [požadavky](#) na shodu.

1. Scope

Předmětem této technické [zprávy](#) je poskytnout vodítko pro využití [UML](#) při návrhu mezinárodních norem a technických [zpráv](#) souvisejících dokumentů.

2. Associated Standards

ISO 14813 (všechny části), Inteligentní dopravní systémy – [Model referenční architektury](#) pro obor ITS (části 1 až 6)

ISO 14817, Informační a řídicí systémy v dopravě – [Požadavky](#) na ITS/TICS centrální [datové registry](#) a ITS/TICS [datové slovníky](#)

[ISO / TR 17452](#), Inteligentní dopravní systémy – Používání [UML](#) (unifikovaného jazyka) pro definování a dokumentaci rozhraní ITS

ISO / IEC 19501, Informační technologie – [Otevřené](#) distribuované zpracování – Unified Modeling Language ([UML](#)) verze 1.4.2

[ISO / TR 25102](#), Inteligentní dopravní systémy – [Architektura systému](#) – Formulář pro forma pro případy použití ITS

3. Terms and Definitions

[aktér; aktor](#) (actor) – koherentní sada [rolí](#), které mohou uživatelé dané entity hrát při interakci s touto entitou

[architektura](#) (architecture) – (ITS) soubor koncepcí a pravidel pro systém, který popisuje vzájemný [vztah](#) mezi subjekty v celém systému, a to [nezávisle](#) na hardwarovém a softwarovém prostředí

[komunikační architektura](#) (communications architecture) – rámec, který říká projektantům, jak [prvky](#) hardware a software harmonicky fungují při použití stejných protokolů a technik bezdrátového rozhraní (kde je to vhodné)

[logická architektura](#) (logical architecture) – [definice](#) procesů (aktivit a funkcí), které jsou požadovány, aby poskytovaly požadované služby pro uživatele

[model](#) ([model](#)) – znázornění [objektu](#), z něhož byly vyčleněny důležité [prvky](#) odstraněním nedůležitých detailů, aby přitom byl zachován vzájemný [vztah](#) mezi klíčovými [prvky](#) celku

[organizační architektura](#) (organizational architecture) – rámec, do kterého jsou obchodní procesy zavedeny a který zajišťuje, že hlavní kvality organizace se uplatní v rámci obchodních procesů probíhajících uvnitř organizace

[fyzická architektura](#) (physical architecture) – vysokoúrovňová struktura procesů a datových toků v [logické](#) architektuře

[referenční architektura](#) (reference architecture) – seznam funkcí a některé indikace jejich rozhraní (nebo API) a jejich interakce vzájemné a s funkcemi umístěnými mimo předmět [referenční architektury](#)

[vztah; vazba](#) (relation/relationship) – způsob, jakým se dvě entity vzájemně ovlivňují včetně [závislosti](#)

[požadavek](#) (requirement) – stanovení potřeby uživatele, typicky vyjádřené jednou větou, aby se mohla posléze použít při ověřování shody

[scénář](#) (scenario) – posloupnost kroků, které se provedou pro změnu [stavu](#) předcházejícího [scénáři](#) do [stavu](#) následujícího hned po provedení [scénáře](#)

[systémová architektura](#) (system architecture) – [systémová](#) architektura je rámcem pro instalaci ITS; je jedním vysokoúrovňovým popisem hlavních [prvků](#) nebo [objektů](#) a jejich vzájemnými vazbami

[šablona](#) (template) – rámec, který lze opakovaně použít pro splnění [požadavků](#), které jsou si do jisté míry podobné

[případ užití](#) (use case) – reprezentace sady interakcí mezi entitami mimo systém a systémem, která končí poskytnutím obchodní hodnoty

Další termíny a zkratky z oboru ITS jsou obsaženy ve [slovníku ITS terminology](#).

4. Abbreviations

ITS - inteligentní dopravní systémy (intelligent transport system)

KAREN - základní architektura vyžadovaná pro evropské sítě (keystone architecture required for European networks)

MDD - [modelem řízený vývoj](#) ([model driven developments](#))

OMG - Standardizační skupina pro správu objektů, například pro vývoj a standardizaci jazyka [UML](#) (object management group)

POM - procesně orientovaná metoda (process oriented methodology)

SEI - institut softwarového inženýrství (software engineering institute)

TICS - informační a řídicí systémy dopravní telematiky (transport information and control systems)

TR - technická [zpráva](#) (technical report)

UML - [unifikovaný modelovací jazyk](#) (unified modelling language)

Další termíny a zkratky z oboru ITS jsou obsaženy ve slovníku ITS terminology (www.ITSterminology.org).

5 Souvislosti

Tato technická [zpráva](#) vznikla z práce WG 1 při zpracování všech částí normy pro ITS architekturu ISO 14813 a řady dalších mezinárodních norem a technických [zpráv](#). Jazyk [UML](#) je standardizován mezinárodní normou ISO / IEC 19501, a proto by měl být podporován jako jedna z modelovacích technik. Cílem technické [zprávy](#) je podnítit diskusi o použitelnosti, silných a slabých stránkách a doporučit nejlepší postupy pro použití [UML](#) v ITS normách.

Jazyk [UML](#) je intenzivně využíván pro specifikaci a modelování náročných softwarových systémů v souladu s normou. [UML](#) rozhodně není jediným dostupným nástrojem, který je k dispozici. Existují i jiné počítačové techniky modelování architektury a mají své výhody a nevýhody ve srovnání s [UML](#). Oblast ITS se stále vyvíjí a jazyk [UML](#) nabízí potřebnou flexibilitu, dobře přizpůsobenou tomuto prostředí. To nejdůležitější je, že kompozitní popis splňuje všechny [požadavky](#) uživatelů i rozhraní a poskytuje vhodný základ nejen pro původní návrh, ale i pro další vývoj systému.

6 Diskuze

Architektura systému identifikuje hlavní [aktory](#), rozhraní a komponenty a poskytuje základnu pro pochopení jejich vnitřních vazeb a interakcí. Bývá popisována v několika hlavních úhlech pohledu:

- [Referenční architektura](#)
- Logická (funkční) [architektura](#)
- [Fyzická architektura](#)
- [Komunikační architektura](#)
- [Organizační architektura](#)

Návrh [architektury](#) je rozhodující fáze ve vývoji každého komplexního systému, jako je právě ITS pracující v síti. Mezi klíčové otázky, které jsou řešeny v [architektuře](#), patří:

- Rozdělení do logických nebo funkčních entit jako jsou subsystémy, moduly a komponenty.
- Rozdělení odpovědnosti za chování systému v rámci [omezení](#) (daných nebo předpokládaných), pro dostupné a/nebo externí entity.
- Identifikace funkčních rozhraní a dalších [vztahů](#) mezi logickými jednotkami.
- Pracovní režimy systému, včetně poškozených a alternativní režimů.
- [Omezení](#) výkonnosti včetně předpokladů jak mohou být charakterizovány.
- Úroveň služeb, které bude možno podporovat.
- [Spolehlivost](#), udržitelnost, dostupnost a bezpečnost systému jako celku.
- Základna pro vývoj prostřednictvím rozšíření, integrace nebo nahrazení entit či rozhraní.

Zlepšení interoperability je náplní řady technických [zpráv](#) včetně [ISO TR 17452](#), která pojednává o použití [UML](#) pro [definici](#) a dokumentaci rozhraní v ITS. Je důležité si uvědomit, že systém nemůže být charakterizován pouze svým chováním na rozhraní nebo protokolech, ale také vyžaduje interoperabilitu [datových konceptů](#).

Pomocí [UML](#) se stanovují [požadavky](#) uživatelů, zapouzdřené v [případech užití](#), které jsou ve [vztahu](#) s jinými [požadavky](#) jiných [případů užití](#), a také s dalšími komponentami [UML](#) (pokud jsou využity nástroje CASE). Nicméně není nutné, aby [případy užití](#) musely být vyjádřeny v jazyce [UML](#), pokud to není účelné a jednoznačné, jak je popsáno v [ISO / TR 25102](#).

Vrstvení [architektury](#) a souvisejících norem je nutné, ale nestačí k definování systému dostatečně podrobně tak, aby návrh a implementace mohla být potvrzena jako důvěryhodná. Musí být brán ohled i na další pohledy a ne všechny z nich jsou spojeny s normami. Jinými slovy, je vždy prostor pro definování nových norem pro řešení nových [vztahů](#), rozhraní a systémů, nebo popis nových pohledů na stejné subjekty.

[Datový registr](#) podporuje vytvoření elementů dříve definovaných ve formálních normalizačních dokumentech postupně definovaných a zveřejňovaných. Tento proces má za cíl poskytovat rychlejší a vstřícnější konsensus o normách aplikovaných v zamýšlené doméně. V tomto přístupu jsou entity zachyceny ve formě modelových fragmentů [UML](#), které jsou vhodné pro harmonizaci a syntézu pro vytváření rozsáhlejších [modelů](#).

V této kapitole jsou též uvedeny předpoklady pro přijetí [logické architektury](#) uživateli. Hlavními předpoklady jsou čitelnost a srozumitelnost technických výstupů [UML](#) a popisů [případů užití](#) pro laiky, stručnost a jednoduchost dokumentů, méně technický a méně upovídaný jazyk.

7 Důsledky (uživatelského přijetí) pro použití [UML](#)

Mezi významné výhody, plynoucí z přijetí standardizovaného přístupu k návrhu [architektury](#), patří jednoznačnost, srozumitelnost, přesnost, rozšiřitelnost a flexibilita pro přijetí nevyhnutelných změn. [UML](#) nabízí užitečné a mezinárodně uznávané prostředky pro dosažení tohoto cíle v rámci [systémové architektury](#).

Stejně jako u všech norem je vždy možné vybrat podmnožinu normativních [prvků](#) a [požadavků](#) jako 'profil' a zadat tento profil pro použití ve stanovených případech ITS norem. Přejemnějším by to mohlo omezit rozsah problémů, které by vznikly vždy při použití nejnovější verze [UML](#), a tak by se snížila pravděpodobnost zpětné nekompatibility, kdy pozdější [modely](#) nemohou být připojeny k předchozím [modelům](#), protože protějšek nemá ve starším [modelu](#) některé z doplněných funkcí.

Existuje několik nástrojů pro modelování [UML](#) a tak je také užitečné provést srovnávací hodnocení [vlastností](#), výhod a nákladů. Co je důležitější je používání společného formátu pro [modely](#) tak, aby modelové části z jednoho nástroje bylo možno použít pro pokračování vývoje v jiném nástroji.

[UML](#) nemůže zcela pokrýt všechny [požadavky](#) na modelování složitých sítí a distribuovaných systémů informačních a komunikačních technologií, ale nabízí nejkompaktnější a flexibilní nástroj, který je k dispozici.

Associated Terms

- [template](#)
- [scenario](#)
- [reference architecture](#)
- [use case](#)
- [requirement](#)
- [request](#)
- [organisational architecture](#)
- [model](#)
- [communications architecture](#)
- [physical architecture](#)
- [relation](#)