Úvod do konceptu Asset Administration Shell

Študijný materiál ÚAMT FEI STU Bratislava

Ing. Tomáš Sedláček, tomas.sedlacek@stuba.sk

29. decembra 2025

Abstrakt

Tento študijný materiál je určený pre študentov, ktorí sa oboznamujú s architektúrou a komunikačnými štandardmi v kontexte Industry 4.0. Hlavnou výzvou tejto digitálnej priemyselnej transformácie je sémantická interoperabilita rozličných zariadení, senzorov a softvérových systémov. Materiál predstavuje kľúčový štandard, Asset Administration Shell (AAS), ktorý rieši práve tento problém transformáciou akéhokoľvek fyzického či digitálneho aktíva na plnohodnotný Komponent Industry 4.0.

Text logicky vedie čitateľa od motivácie a kontextu Industry 4.0 cez detailnú štruktúru AAS až po praktické aspekty spojené s týmto konceptom. Predstavuje metódy komunikácie a oboznamuje s nástrojmi pre prácu a návrh digitálnych opisov aktív práve pomocou AAS. Cieľom je poskytnúť ideálny a pútavý odrazový mostík pre hlbšie štúdium digitálnych dvojčiat a ich implementáciu v moderných priemyselných systémoch.

Obsah

1 Úvod – Prečo potrebujeme AAS? 4
1.1 Motivácia a problém interoperability v Industry 4.0 4

2 Komponent Industry 4.0 5
2.1 Od Aktíva (Asset) ku Komponentu I4.0 6
2.2 Prepojenie s modelom RAMI 4.0 7

3 Štruktúra AAS - Čo sa skrýva vo vnútri? 8
3.1 Základná štruktúra AAS 8
3.2 Sila Submodelov 9
3.3 SubmodelElementy 9
3.4 Sémantický identifikátor - kľúč k pozorumeniu 12

4 Komunikácia AAS 13
4.1 Statická výmena pomocou súboru .aasx 13
4.2 Dynamická výmena cez API 14
4.3 Priama Peer-to-Peer komunikácia 14

5 Tvorba AAS – Nástroje pre digitálne modelovanie 15
5.1 Nástroje na návrh a modelovanie AAS 15
5.2 Platformy pre implementáciu AAS a ich prevádzku 16

6 Záver 18

Zoznam obrázkov

1 Znázornenie významu syntaktickej interoperability 4
2 Znázornenie významu sémantickej interoperability 5
3 Komponent Industry 4.0 6
4 Referenčný model architektúry Industry 4.0 - RAMI 4.0 7
5 Kritériá pre Komponent Industry 4.0 8
6 Príklad štruktúry navrhovaného AAS pri výskumných činnostiach na ÚAMT 9
7 Príklad atribútu Property (vlastnosť) s idShort = temperature, ktorý je referencovaný na externý IRDI element 11
8 Príklad vlastnosti ‘max. supply voltage‘ z knižnice ECLASS s IRDI: 0173-1#02-AAC965#010 12
9 Spôsoby výmeny dát prostredníctvom Asset Administration Shell:
4.1 Statická výmena pomocou súboru .aasx
4.2 Dynamická výmena cez API
4.3 Priama Peer-to-Peer komunikácia 13
10 Rozdelenie využívaných API rozhraní v rámci komunikačnej vrstvy z RAMI 4.0 14
11 Prostredie aplikácie AASX Package Explorer v režime editácie AAS 15
12 Mapovanie konkrétneho AAS realizovaného OPC UA serverom 17

Zoznam tabuliek

1 Tabuľka kľúčových atribútov prvku Property 10

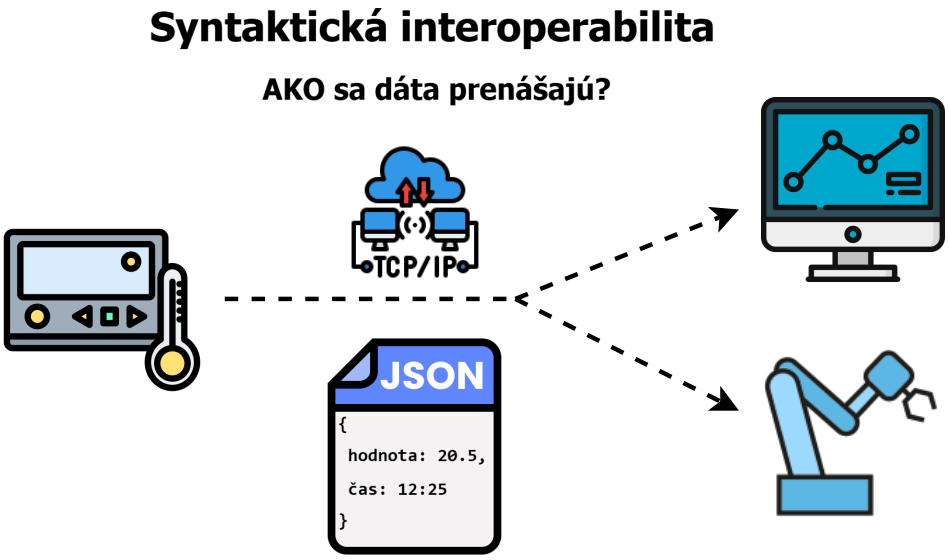
1 Úvod – Prečo potrebujeme AAS?

Súčasná éra priemyselnej výroby, označovaná ako **Industry 4.0**, prináša so sebou viacero kľúčových aspektov, ktorých hlavným cieľom je transformovať výrobné prostredia na moderné a plne digitalizované podniky. Dosiahnutie tohto stavu však naráža na zásadný problém modernej priemyselnej komunikácie – nedostatok **interoperability**, ktorý vedie k dátovému chaosu. Možné riešenie prináša práve koncept AAS objasnený v tomto učebnom texte.

1.1 Motivácia a problém interoperability v Industry 4.0

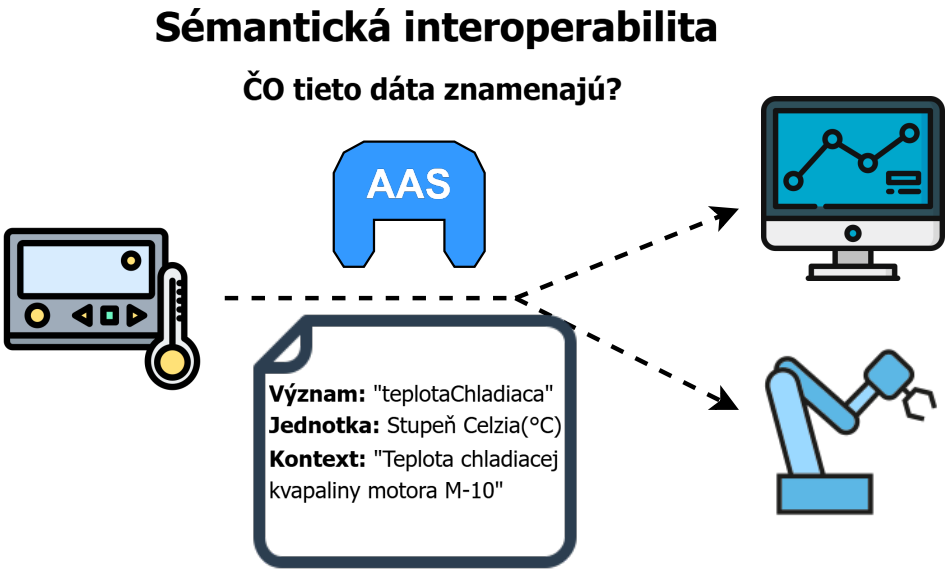
Ruka v ruke s konceptom Industry 4.0 ide aj rozmach **Priemyselného internetu vecí (IIoT)**, ktorý priniesol na trh obrovské množstvo zariadení schopných komunikovať medzi sebou ako aj so širokým okolitým svetom. Tieto zariadenia využívajúce najrôznejšie komunikačné protokoly sú schopné poskytovať obrovské množstvá dát do nadriadených systémov. Napriek tejto konektivite však nemožno povedať, že by boli všetky aspekty konceptu Industry 4.0 úspešne naplnené. Kľúčové je dôsledne porozumieť interoperabilite - teda schopnosti systémov vzájomne komunikovať, porozumieť si a spolupracovať. Túto problematiku je však nutné vnímať na dvoch základných úrovniach:

- **Syntaktická interoperabilita**: Schopnosť systému navzájom si vymieňať dáta v dohodnutom formáte prostredníctvom spoločného komunikačného protokolu. Táto úroveň je pomocou dnešných technológií dosiahnuteľná pomerne jednoducho.



Obr. 1: Znázornenie významu syntaktickej interoperability

- **Sémantická interoperabilita**: Schopnosť systémov správne a autonómne interpretovať význam (sémantiku) vymieňaných dát. Nestačí, že systém prijíma dáta z iného zariadenia, ale automaticky by mal vedieť, čo táto hodnota predstavuje.



Obr. 2: Znázornenie významu sémantickej interoperability

Absencia sémantickej interoperability vedie k neefektívnej a rigidnej architektúre, ktorá má v praxi niekoľko kľúčových negatívnych dôsledkov:

- **Manuálna integrácia:** Každé nové zariadenie si vyžaduje prácu systémových integrátorov, ktorí musia manuálne naprogramovať a zmapovať význam dát nového zariadenia. Ide o mimoriadne neefektívnea nákladné riešenie náchylné na chyby.
- **Nízka flexibilita a škálovateľnosť:** Systémy sú vzájomne fixne prepojené. Pridanie alebo výmena akéhokoľvek komponentu tak nemusí byť rýchla a operatívna, ale môže ísť o zložitý proces.

Cieľom je dosiahnutie princípu **Plug&Produce**. Ide o koncept predstavujúci schopnosť zariadení autonómne sa integrovať do výrobného procesu a ihneď po zapnutí/pripojení do siete (**Plug**) začať poskytovať svoje funkcie a dáta v zrozumiteľnej forme (**Produce**).

Práve sémantická interoperabilita je pre dosiahnutie princípu Plug&Produce v priemyselnom prostredí kľúčová. Zariadeniam a systémom ponúka možnosť komunikovať univerzálnym, strojovo čitateľným jazykom, ktorý otvára dvere k skutočne flexibilnej a autonómnej výrobe.

Pre dosiahnutie týchto princípov bol v rámci nemeckej iniciatívy *Plattform Industrie 4.0* vyvinutý štandardizačný koncept, ktorý definuje jednotný digitálny opis akéhokoľvek priemyselného aktíva. Tento koncept je známy práve pod pojmom **Asset Administration Shell (AAS)** a detailne sa mu venujú ďalšie podkapitoly.

2 Komponent Industry 4.0

Pre dosiahnutie skutočnej interoperability a autonómie v Industry 4.0 bol zavedený tzv. **Komponent Industry 4.0** (I4.0 Component). Jeho základom je práve spomínaný Asset

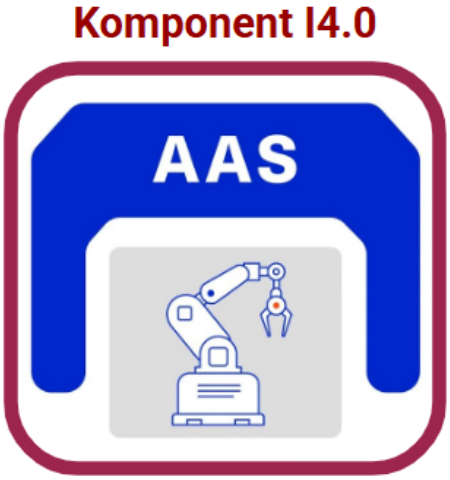
Administration Shell (AAS). Takto vytvorený komponent umožňuje aktívam sémanticky zrozumiteľne komunikovať s okolitým svetom.

Táto časť sa venuje detailnej analýze tohto kľúčového konceptu. Definuje, ako samotné AAS transformuje bežné fyzické alebo digitálne aktíva (Asset) na plnohodnotný Komponent Industry 4.0. Následne zasadzuje tento dôležitý prvok do kontextu celkovej architektúry definovanej referenčným modelom **RAMI 4.0**.

2.1 Od Aktíva (Asset) ku Komponentu I4.0

Základným stavebným kameňom každého výrobného systému je aktívum (Asset). Podľa definície normy IEC je aktívum akýkoľvek fyzický, digitálny alebo nehmotný objekt, ktorý má pre organizáciu hodnotu. V kontexte priemyselnej automatizácie môže byť aktívom prakticky čokoľvek: od jednoduchého snímača, cez elektromotor, robotické rameno, softvérovú funkciu, až po celú výrobnú linku. Samotné aktívum však v digitálnom svete Industry 4.0 automaticky neznamená, že je schopné interoperabilnej komunikácie.

Kľúčové je práve zaobalenie takéhoto aktíva pomocou konceptu Asset Administration Shell (AAS). AAS si môžeme v praxi predstaviť ako **digitálny pas aktíva**. Tento digitálny pas, digitálny obal alebo schránka zapuzdruje reálne aktívum a sprístupňuje ho digitálnemu svetu prostredníctvom štandardizovaného rozhrania a dátovej štruktúry.



Obr. 3: Komponent Industry 4.0

Zapúzdrené aktívum pomocou AAS nazývame v praxi ako spomínaný **Komponent Industry 4.0**. Samotný komponent je teda viac než len fyzické zariadenie, je to celý systém, ktorý v sebe spája:

- **Reálny svet**: Samotné fyzické alebo logické aktívum.
- **Digitálny svet**: Jeho štandardizovaná digitálna reprezentácia - Asset Administration Shell.

AAS je teda technickou špecifikáciou na implementáciu **Digitálneho dvojčaťa (Digital Twin)**, ktorej hlavným cieľom je zabezpečiť sémantickú interoperabilitu. Hlavné prínosy tejto transformácie sú:

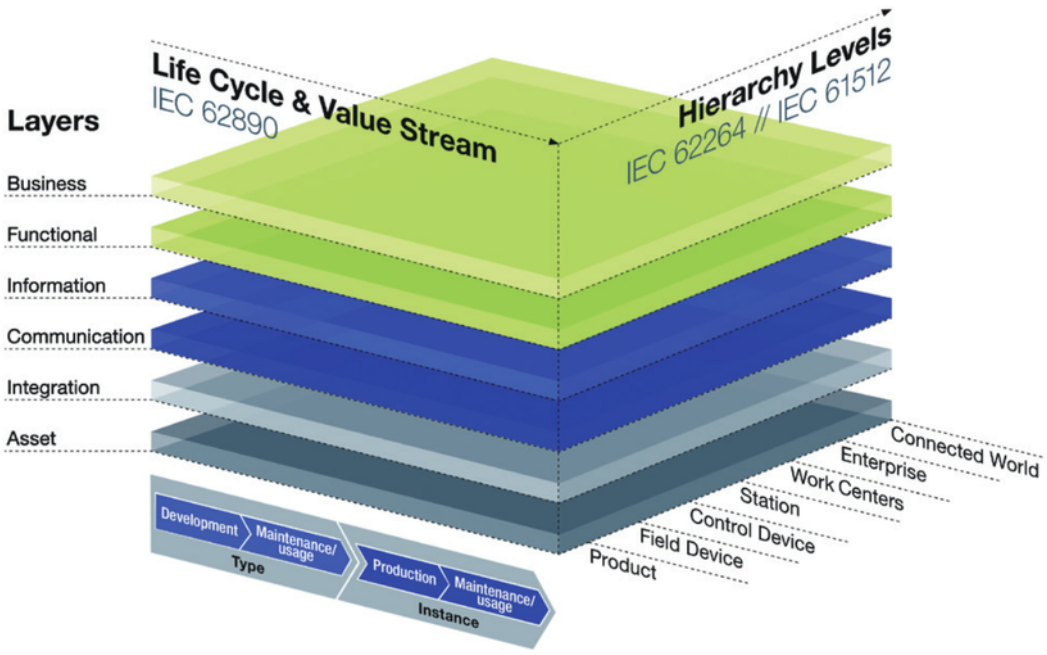
- **Štandardizácia**: AAS definuje jednotnú štruktúru, obsah a správanie digitálnej reprezentácie, vďaka čomu si komponenty rozumejú bez ohľadu na výrobcu.

- **Kompletnosť**: Všetky relevantné informácie o aktíve sú združené na jednom mieste - od identifikačných údajov a technických parametrov, cez dokumentáciu, až po dynamické prevádzkové dáta a ponúkané služby.
- **Jednotné rozhranie**: AAS poskytuje štandardizovaný prístup k informáciám, čo dramaticky zjednodušuje integráciu do nadradených systémov a umožňuje automatizované spracovanie dát.

Týmto spôsobom sa z akéhokoľvek, aj "hlúpeho" aktíva, môže stať inteligentná, autonómne komunikujúca jednotka pripravená na zapojenie do flexibilných dynamických výrobných procesov.

2.2 Prepojenie s modelom RAMI 4.0

Komponent Industry 4.0 a s ním previazaný Asset Administration Shell nie sú izolované nápady, ale ide o koncepty pevne ukotvené v hlavnom referenčnom modeli spojeného s Industry 4.0. Ide o pojem známy ako referenčný model architektúry Industry 4.0, označovaný skratkou **RAMI 4.0** (Reference Architectural Model Industry 4.0). Tento trojrozmerný model bol vytvorený s cieľom poskytnúť spoločnú a štruktúrovanú mapu pre všetky aspekty Industry 4.0. Zabezpečiť sa tak má vzájomné porozumenie naprieč vrstvami a účastníkmi v nich - od výrobcov komponentov až po systémových integrátorov.



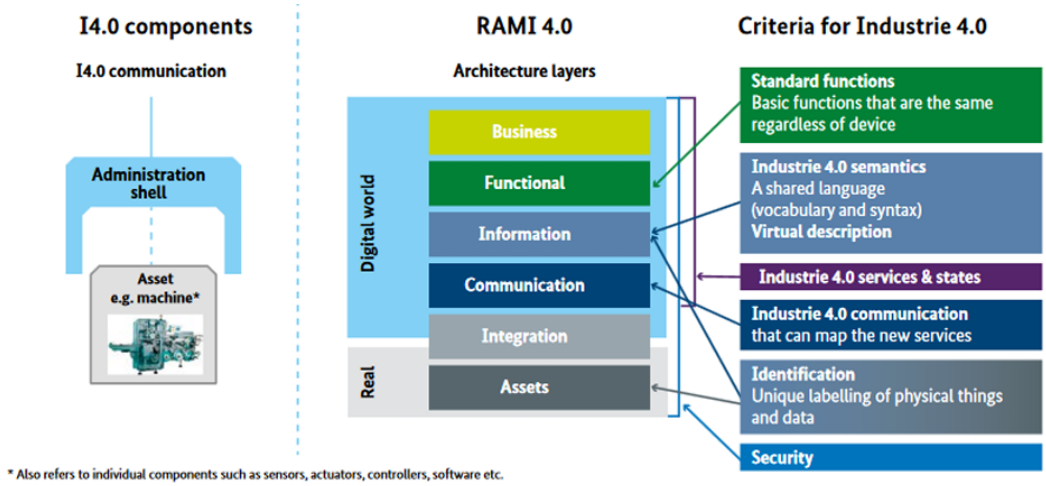
Obr. 4: Referenčný model architektúry Industry 4.0 - RAMI 4.0

RAMI 4.0 je definovaný tromi osami, ktoré spoločne opisujú všetky dôležité aspekty Industry 4.0:

1. **Vertikálna os (Layers - Vrstvy):** Predstavuje funkčnú dekompozíciu celého podniku, od najnižšej vrstvy aktív (assetov) až po najvyššiu podnikovú vrstvu.
2. **Ľavá horizontálna os (Life Cycle & Value Stream - Životný cyklus a dátový tok):** Predstavuje životný cyklus zariadenia alebo produktu na základe normy IEC 62890.

3. **Pravá horizontálna os (Hierarchy Levels - Hierarchické úrovne):** Definuje hierarchické usporiadanie v rámci výrobného podniku na základe normy IEC 62264.

Pre lepšie pochopenie vzájomného prepojenia je na obrázku nižšie priblížené lepšie mapovanie Komponentu Industry 4.0 do modelu RAMI 4.0, teda jeho osi Layers (vrstiev) a konkrétnych častí tejto osi. Tie sú pre dôsledné začlenenie rozdelené na digitálny svet (Digital world), čo je práve doménou samotného Asset Administration Shell. Časť reálneho sveta (Real world) je zase reprezentovaná už samotným aktívom (Asset), pričom ich spojením dochádza k realizácii celého Komponentu Industry 4.0.



Obr. 5: Kritériá pre Komponent Industry 4.0

Obrázok vyššie teda približuje konkrétny kontext a zároveň definuje kritériá, ktoré musí komponent spĺňať na to, aby bol považovaný za platný Komponent Industry 4.0.

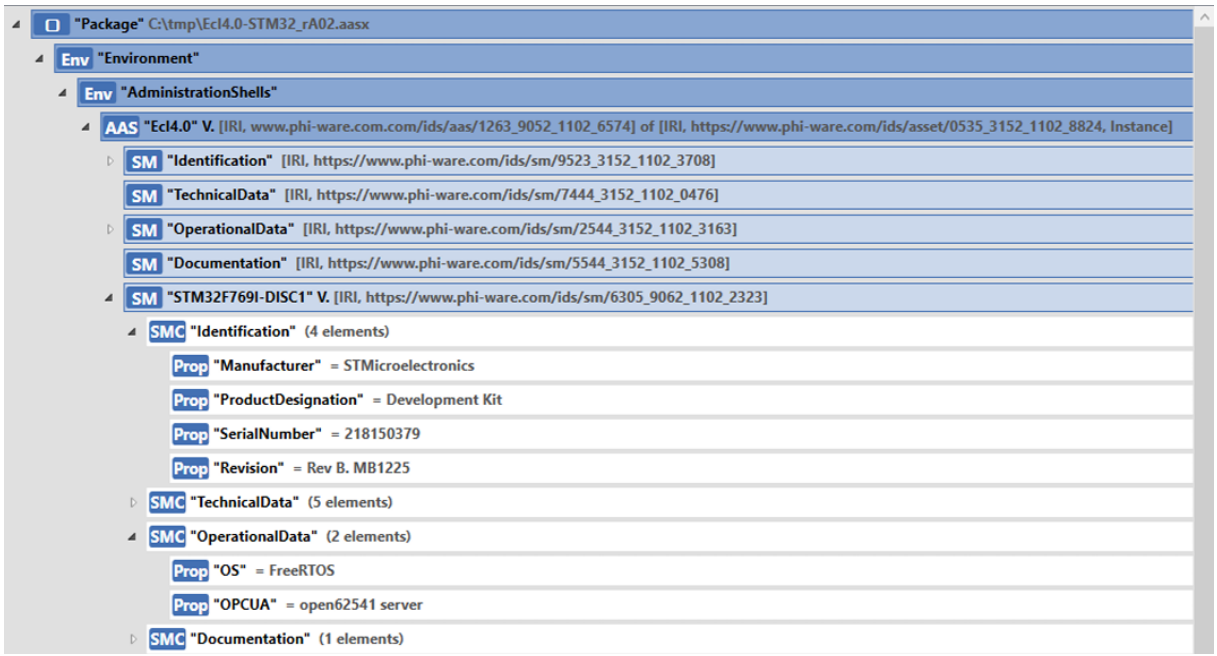
3 Štruktúra AAS - Čo sa skrýva vo vnútri?

Asset Administration Shell si možno predstaviť ako štandardizovaný kontajner pre všetky informácie o aktíve. Jeho sila spočíva v modulárnej a hierarchickej štruktúre, ktorá umožňuje detailne a sémanticky presne popísať akýkoľvek priemyselný komponent.

3.1 Základná štruktúra AAS

Podľa špecifikácie je AAS abstraktným dátovým modelom popísaným pomocou UML (Unified Model Language), ktorý je nezávislý od konkrétnej technológie jeho implementácie (XML, OPC UA, JSON, AutomationML a pod.). Jeho štruktúra pozostáva zo základných častí:

- **AAS:** Hlavný kontajner obsahujúci jedinečný identifikátor a názov reprezentujúci dané aktívum.
- **SM (Submodel)**: Kľúčový prvok, ktorý štruktúrovane zoskupuje informácie (atribúty) o jednom konkrétnom aspekte aktíva (napr. technické dáta, dokumentácia a pod.).



Obr. 6: Príklad štruktúry navrhovaného AAS pri výskumných činnostiach na ÚAMT

- **Asset:** Referencia na konkrétne aktívum, ktoré dané AAS popisuje v digitalizovanej podobe.

Jednotlivé Submodely, atribúty, kolekcie či assety sú vzájomne prepojené vzťahmi a referenciami, čo umožňuje pomocou AAS štruktúrovane popísať aj veľmi komplexné aktíva.

3.2 Sila Submodelov

Konkrétne Submodely zoskupujúce atribúty môžu byť ďalej hierarchicky štruktúrované, pričom môžu obsahovať už konkrétne atribúty daného Submodelu, ale aj kolekcie SMC (SubmodelElement Collection), ktoré ďalej hierarchicky rozdeľujú štruktúru daného Submodelu. Každý takýto Submodel predstavuje štandardizovaný pohľad na určitú oblasť informácií o aktíve.

Príklady bežne používaných Submodelov:

- **Identifikácia:** Základné údaje o výrobcovi, sériové číslo, typové označenie.
- **Dokumentácia:** Odkazy na manuály, certifikáty, technické výkresy, 3D modely.
- **Prevádzkové dáta:** Dynamické hodnoty z prevádzky meniace sa v reálnom čase.
- **Technické dáta:** Fyzikálne a technické vlastnosti ako rozmery, hmotnosť, spotreba a pod.

3.3 SubmodelElementy

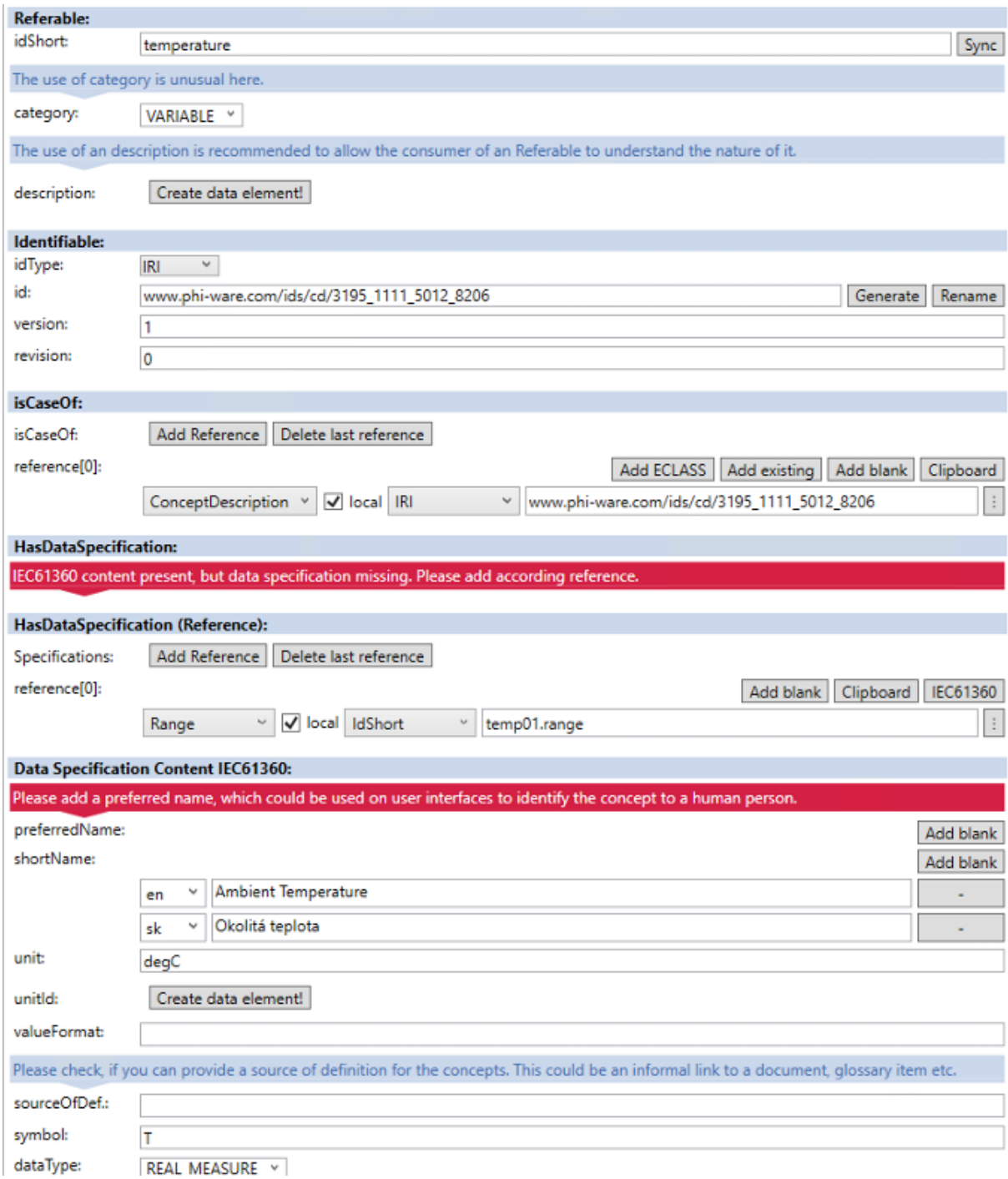
Konkrétne Submodely sa skladajú zo základných prvkov, nazývaných **SubmodelElementy**, ktoré sú nositeľmi samotných dát. Tieto prvky sú v podstate digitalizovanými atribútmi reálneho objektu. Najzákladnejším a najpoužívanejším atribútom je Property (vlastnosť), pričom zoznam jednotlivých SubmodelElementov je pomerne rozsiahly:

- AnnotatedRelationshipElement
- BasicEventElement
- Blob
- Capability
- DataElement
- Entity
- EventElement
- File
- MultiLanguageProperty
- Operation
- Property
- Range
- ReferenceElement
- RelationshipElement
- SubmodelElementList
- SubmodelElementCollection

Samotné atribúty sú špecifickým mapovaním konkrétnych hodnôt a identifikátorov, ktoré daný SubmodelElement nesie. Dôležitá je tak štandardizácia jednotlivých atribútov tak, aby došlo k automatickému rozpoznaniu týchto údajov iným systémom. Jednotlivé SubmodelElementy nemusia mať jednotnú štruktúru konkrétnych atribútov, no pre lepšiu predstavu si uveďme zjednodušený príklad kľúčových atribút napr. pre prvok **Property**:

Atribút	**Popis**	**Príklad**
idShort	Krátky, v rámci submodelu unikátny identifikátor (názov premennej)	MaxSpeed
valueType	Dátový typ hodnoty (napr. xs:int, xs:string, xs:double)	xs:int
value	Samotná hodnota vlastností	3000
semanticId	Unikátny odkaz na sémantickú definíciu v externom slovníku	IRDI: 0173-1#02-BAA122#006
displayName	Ľudsky čitateľný názov	Font Size
description	Voliteľný, ľudsky čitateľný popis	Maximálne povolené otáčky motora

Tabuľka 1: Tabuľka kľúčových atribútov prvku Property



Obr. 7: Príklad atribútu Property (vlastnosť) s idShort = temperature, ktorý je referencovaný na externý IRDI element

3.4 Sémantický identifikátor - kľúč k pozorumeniu

Ako naznačuje samotný názov tohto atribútu, semanticId je najdôležitejším prvkom pre dosiahnutie sémantickej interoperability. Poskytuje jednoznačný, strojovo čitateľný význam pre každý prvok v AAS. Samotnú sémantiku daného atribútu je možné definovať dvoma spôsobmi, pričom oba vždy odkazujú na konkrétny popis daného konceptu, tzv. ConceptDescription:

1. **Globálne - pomocou externých slovníkov**:
   - Používa odkaz na medzinárodné štandardizované slovníky, napr. ako ECLASS, v ktorej má každá vlastnosť unikátny identifikátor IRDI (International Registration Data Identifier).
   - **Príklad:** 0173-1#02-AAB822#006 globálne a jednoznačne definuje vlastnosť ‘maximálny menovitý prevádzkový prúd motora‘ (max. rated operating current of the motor). Akýkoľvek systém, ktorý pozná štandard ECLASS, presne vie, čo táto hodnota znamená, bez ohľadu na jej názov (idShort).

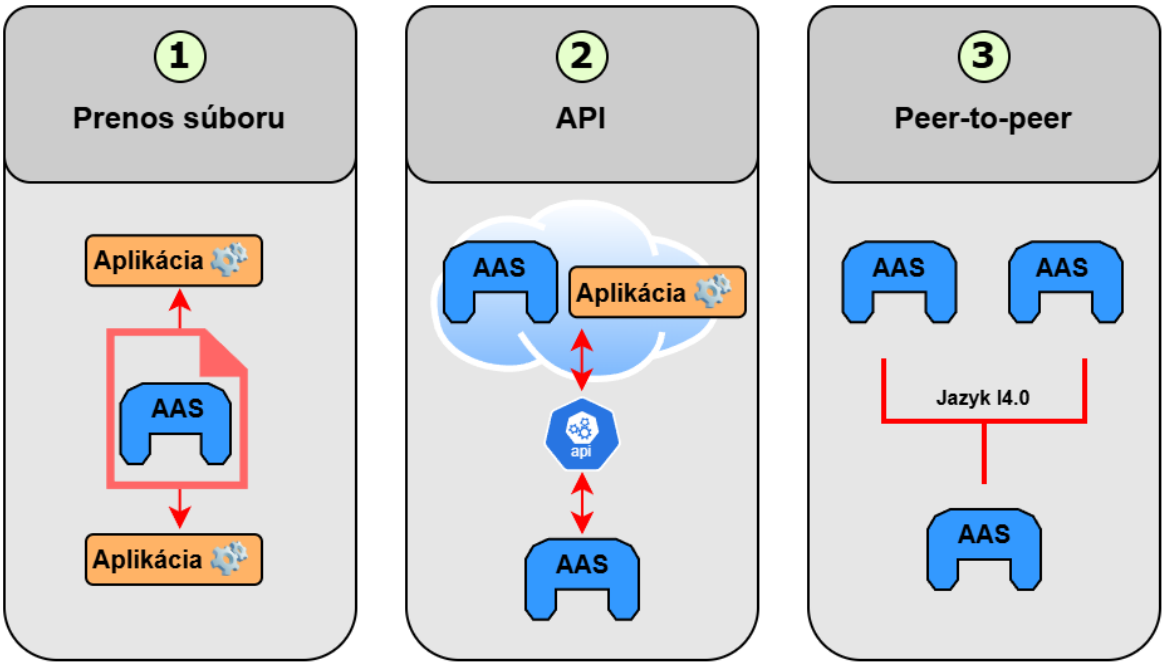
Obr. 8: Príklad vlastnosti ‘max. supply voltage‘ z knižnice ECLASS s IRDI: 0173-1#02-AAC965#010

2. **Lokálne**:
   - Ak pre danú vlastnosť neexistuje globálny štandard, je možné jej sémantiku definovať referenciou na vlastný, lokálny repozitár s popismi typu ConceptDescription. Takáto referencia tak slúži pre vzájomné porozumenie systémov napr. v rámci jedného výrobného podniku alebo v jednej konkrétnej spoločnosti.

Práve vďaka sémantickým identifikátorom sú AAS nie len kontajnerom plným dát, ale aj nositeľom **jednoznačného významu**, čo je základným predpokladom pre autonómnu a inteligentnú komunikáciu v Industry 4.0.

4 Komunikácia AAS

Samotná existencia štruktúrovaného digitálneho modelu v podobe AAS nestačí. Aby mohlo AAS plniť svoju úlohu v prepojenom priemyselnom prostredí, musí existovať štandardizovaný spôsob, ako k jeho dátam pristupovať a ako ich vymieňať medzi rôznymi systémami a aplikáciami. Špecifikácia AAS definuje niekoľko základných mechanizmov výmeny informácií znázornených na obrázku:



Obr. 9: Spôsoby výmeny dát prostredníctvom Asset Administration Shell:
4.1 Statická výmena pomocou súboru .aasx
4.2 Dynamická výmena cez API
4.3 Priama Peer-to-Peer komunikácia

4.1 Statická výmena pomocou súboru .aasx

Súbor .aasx je štandardizovaný kontajnerový formát, v podstate archív (podobný .zip) obsahujúci celú štruktúru AAS (vo formáte XML alebo JSON) a všetky k nej priradené externé súbory (napr. PDF dokumentáciu, 3D modely, obrázky).

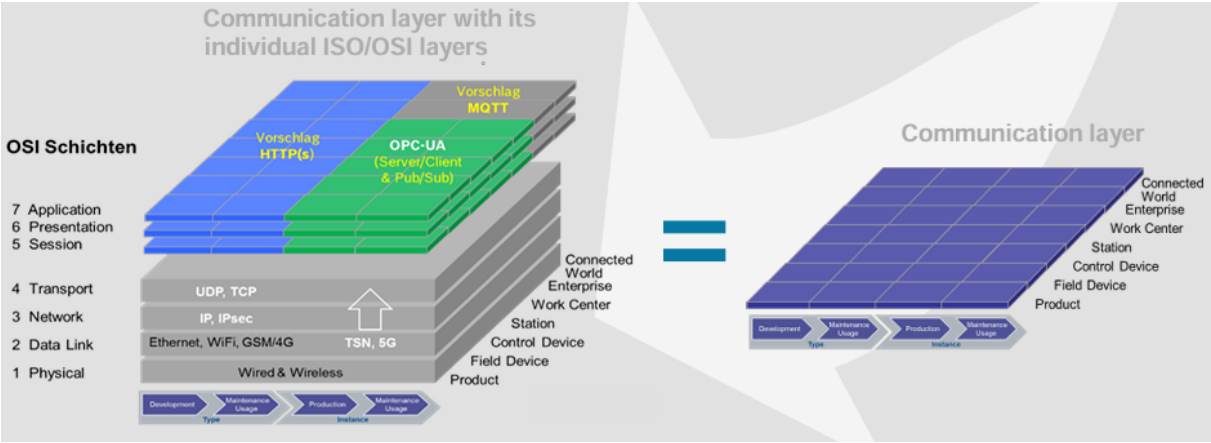
- **Obsah:** Jeden .aasx súbor typicky obsahuje jednu alebo viac AAS a s nimi súvisiace Submodely a dáta.
- **Použitie:** Tento formát je ideálny pre statickú výmenu kompletného digitálneho dvojčaťa medzi rôznymi fázami životného cyklu alebo medzi partnermi, napr.:
  - Výrobca zariadenia dodá zákazníkovi .aasx súbor obsahujúci kompletnú technickú dokumentáciu, konfiguračné parametre a celú špecifikáciu daného zariadenia popísanú pomocou AAS.
  - Prenos digitálneho modelu medzi inžinierskymi nástrojmi alebo medzi rôznymi systémami.

– Archivácia stavu daného aktíva.

- **Nevýhoda:** Nie je vhodný pre výmenu dynamických dát.

4.2 Dynamická výmena cez API

Pre prístup k aktuálnym a dynamickým dátam a funkciám slúži aplikačné programovacie rozhranie (API). AAS je v takomto prípade 'hostovaná' na serveri (napr. Eclipse BaSyx AAS Server), ktorý umožňuje externým aplikáciám (klientom) čítať hodnoty (napr. technické údaje, aktuálne dáta), zapisovať hodnoty (nastaviť cieľovú rýchlosť) alebo volať operácie definované v AAS.



Obr. 10: Rozdelenie využívaných API rozhraní v rámci komunikačnej vrstvy z RAMI 4.0

Špecifikácia AAS definuje štandardizované API rozhrania, pričom najčastejšie používané technológie sú:

- **REST (Representational State Transfer):** Webové API vhodné pre integráciu s IT systémami a cloudovými aplikáciami. Poskytuje prístup k dátam AAS prostredníctvom štandardných REST metód.
- **OPC UA (Open Platform Communications Unified Architecture):** Široko rozšírený štandard pre priemyselnú komunikáciu (IEC 62541). Umožňuje mapovať štruktúru AAS priamo do OPC UA informačného modelu. Poskytuje robustnú, bezpečnú a sémanticky bohatú komunikáciu vhodnú pre výrobné prostredie.
- **MQTT (Message Queuing Telemetry Transport):** Odľahčený protokol založený na princípe Publish/Subscribe, vhodný pre IoT zariadenia a prenos telemetrických dát.

4.3 Priama Peer-to-Peer komunikácia

Pokročilejší model umožňujúci priamu komunikáciu medzi jednotlivými Komponentmi Industry 4.0 bez centrálneho sprostredkovateľa. Komunikácia prebieha prostredníctvom štandardizovaných interakčných vzorov a spoločného **Jazyka Industry 4.0 (I4.0 Language)**. Tento prístup je základom pre autonómne systémy, kde si komponenty môžu napríklad samy vyjednávať úlohy alebo zdieľať informácie priamo medzi sebou.

5 Tvorba AAS – Nástroje pre digitálne modelovanie

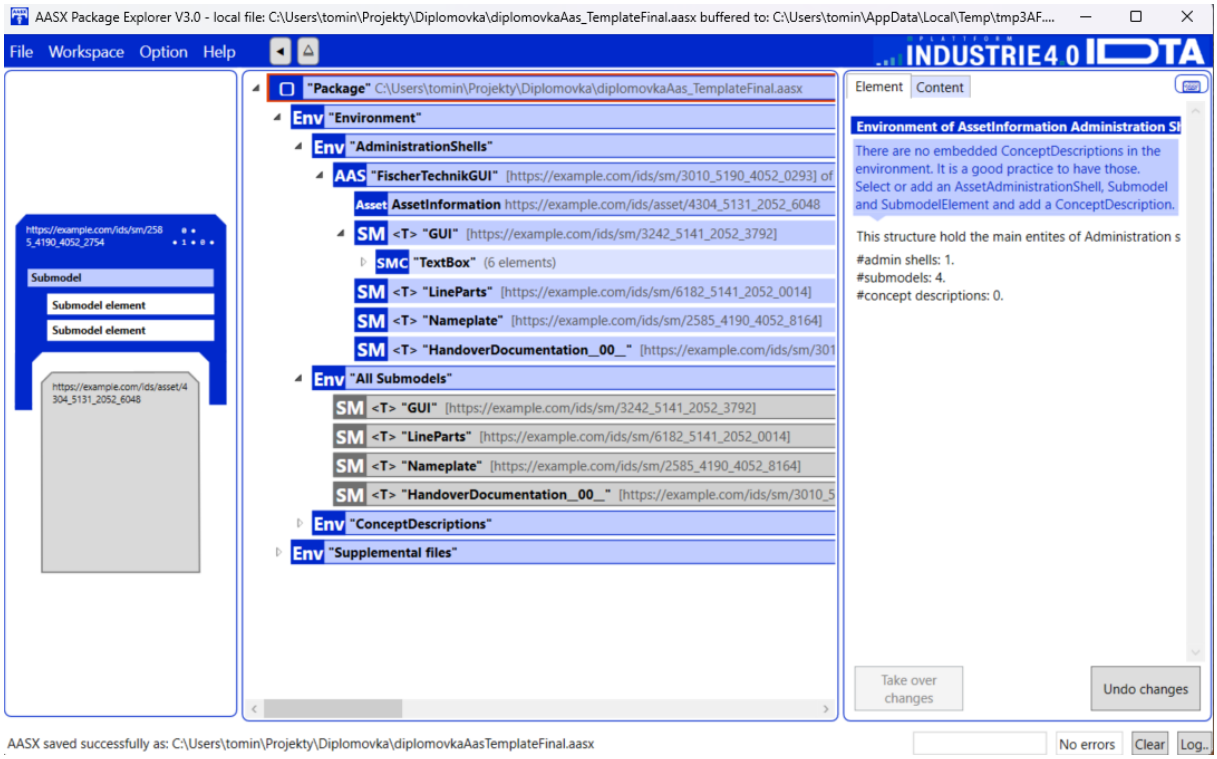
Pre reálnu implementáciu konceptu AAS v priemyselnom prostredí existuje viacero softvérových riešení, ktoré umožňujú prácu s nimi a ich samotnú tvorbu. Úlohy jednotlivých nástrojov sa zväčša prelínajú, teda umožňujú nie len ich návrh a modelovanie štruktúry, ale zároveň umožňujú aj ich reálnu implementáciu, napr. v podobe kontajnerizovaného serverového riešenia. Aj napriek prelínaniu funkcionalít týchto nástrojov si ich však logicky rozdelíme na časť s nástrojmi pre návrh a časť s nástrojmi pre implementáciu.

5.1 Nástroje na návrh a modelovanie AAS

Primárnym cieľom týchto nástrojov je umožniť používateľovi definovať AAS štruktúry, Submodely a ich atribúty podľa oficiálnej špecifikácie tak, aby boli AAS interoperabilné naprieč priemyselnými aplikáciami. Medzi najznámejšie nástroje patrí **AASX Package Explorer**, za ktorým stojí samotná organizácia IDTA (Industrial Digital Twin Association).

Medzi najväčšie výhody tohto nástroja patrí:

- Intuitívne ovládanie a jednoduchá editácia celej hierarchie AAS bez potreby programovania alebo čítania/úpravy zložitejšej XML štruktúry.
- Pracuje s formátom .aasx, ktorý slúži ako štandardizovaný kontajner pre prenos AAS vrátane pridružených súborov.
- Umožňuje aj export modelu do formátov ako XML, JSON, AutomationML a pod., pričom umožňuje aj export do vizualizačných formátov, napr. ako UML.



Obr. 11: Prostredie aplikácie AASX Package Explorer v režime editácie AAS

Pre návrh a modelovanie AAS existujú aj ďalšie nástroje, ktoré však zároveň slúžia aj ako kontajnerizované serverové riešenie pre ich správu. Ich úprava je tak vo väčšine prípadov technicky zložitejšia a neumožňujú jednoduché grafické používateľské rozhranie ako spomenutý AASX Package Explorer.

5.2 Platformy pre implementáciu AAS a ich prevádzku

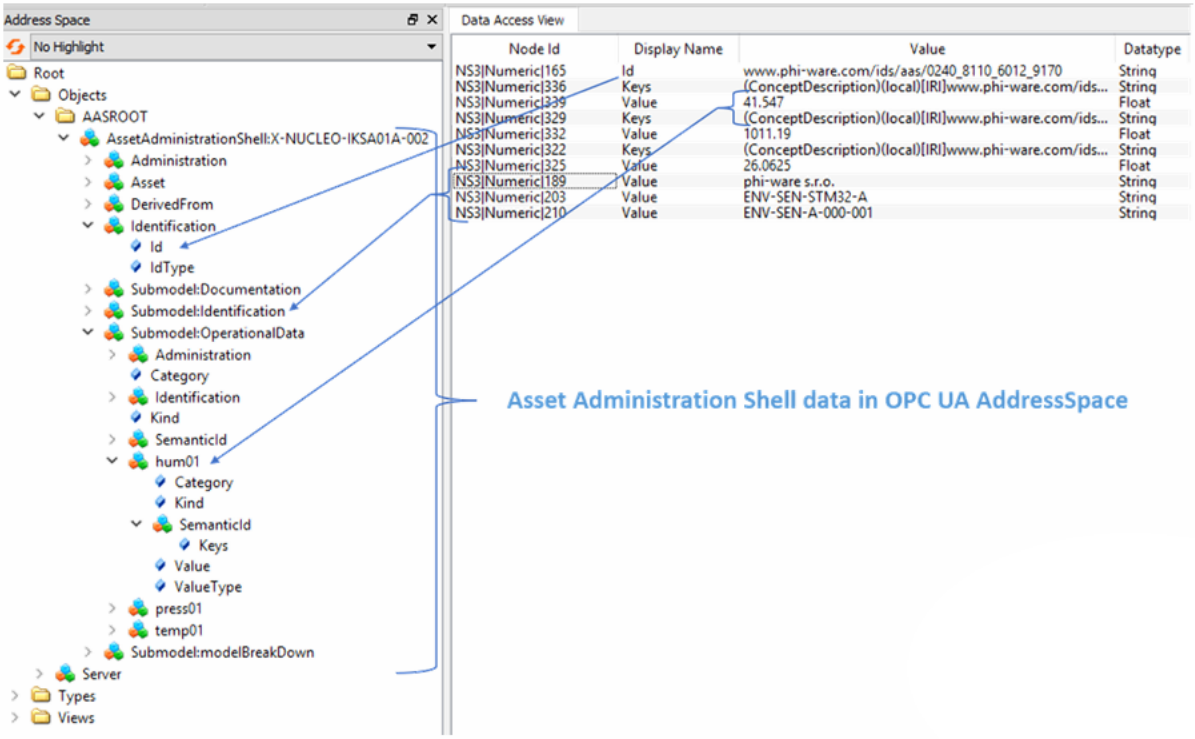
Ako bolo spomenuté vyššie, existujú nástroje ponúkajúce serverové riešenie, ktoré slúži pre 'oživenie' samotných AAS, ale aj ako nástroj pre ich editáciu (takže funkcie implementácie a návrhu sa prelínajú). Medzi aktuálne najznámejšie implementačné platformy patria nástroje Eclipse BaSyx, AASX Server, SAP i40 aas (už archivované) a NOVAAS. Z pohľadu praktickej implementácie sa na Ústave automobilovej mechatroniky najviac osvedčili dve riešenia, ktorých výhody sú zhrnuté v nasledujúcich bodoch:

Eclipse BaSyx

- **Charakterstika:** Komplexná open-source softvérová platforma a referenčná implementácia pre prácu s AAS. Je navrhnutá pre serverové nasadenie (lokálne alebo v cloude).
- **Kľúčové komponenty:** Obsahuje AAS Server, ktorý dokáže hostovať vytvorené AAS modely, ako aj AAS Registry pre vyhľadávanie AAS v sieti a sadu ďalších užitočných nástrojov.
- **Podporované API:** Štandardne ponúka prístup k hostovaným AAS cez REST API, no ponúka aj vlastné aplikačné riešenia pre OPC UA či prepojenia na ďalšie databázové systémy.
- **Technológie a nasadenie:** Implementácie sú dostupné v jazykoch Java, .Net Core, C++, Python, TypeScript a Rust. Nasadenie je realizované pomocou Docker kontajnerov, čo zjednodušuje spustenie na rôznych platformách.
- **Výhody:** Poskytuje robustné riešenie pre správu viacerých AAS, podporuje databázové úložisko pre perzistentné ukladanie AAS modelov. Je kompatibilný s AASX Package Explorer.
- **Nevýhody:** Náročnejšie na výpočtový výkon, takže nie je vhodný pre implementáciu na embedded zariadeniach.

OPC UA

- **Charakteristika:** OPC UA (IEC 62541) nie je len komunikačným protkolom, ale vďaka svojmu flexibilnému informačnému modelu môže slúžiť ako priama platforma pre implementáciu štruktúry AAS. Existuje štandardizované mapovanie medzi AAS metamodelom a OPC UA informačným modelom.
- **Implementácia:** Namiesto hostovania AAS na externom serveri sa štruktúra AAS vytvorí priamo v rámci adresného priestoru OPC UA servera. Toto mapovanie je možné realizovať napríklad exportom z AASX Package Explorer do formátu OPC UA Nodeset2.xml a následným načítaním dát do OPC UA servera.



Obr. 12: Mapovanie konkrétneho AAS realizovaného OPC UA serverom

- **Výhody:**
  - **Vstavané systémy (Embedded):** OPC UA servery môžu mať veľmi nízke hardvérové nároky, čo umožňuje ich implementáciu priamo na embedded zariadeniach (napr. mikrokontrólery), čím môžeme vytvoriť tzv. **Vstavaný Komponent Industry 4.0**.
  - **Priame prepojenie s aktívom:** OPC UA server bežiaci na zariadení môže efektívne prepojiť fyzické signály s ich digitálnou reprezentáciou v AAS, čím pokrýva integračnú vrstvu modelu RAMI 4.0.
  - **Jednotná komunikácia:** Celá komunikácia prebieha výhradne cez OPC UA, čo zjednodušuje celkovú architektúru.
- **Nevýhody:** Implementácia priamo v OPC UA môže byť komplexnejšia na úvodné nastavenia a integráciu v porovnaní s použitím hostovaného servera. Zároveň nemusí byť mapovanie vzťahov AAS prvkov do OPC UA úplne triviálne, s čím treba počítať pri implementácii.

6 Záver

Tento študijný materiál predstavil koncept Asset Administration Shell (AAS) ako kľúčový prvok pre dosiahnutie sémantickej interoperability v kontexte Industry 4.0. Text jasne predstavil hlavnú výzvu moderných priemyselných systémov, ktorou je schopnosť zariadení a systémov od rôznych výrobcov nie len vymieňať si dáta, ale aj rozumieť ich významu.

Kľúčové charakteristiky a prínosy AAS možno zhrnúť nasledovne:

- Slúži ako **štandardizovaná digitálna reprezentácia** akéhokoľvek aktíva.
- Transformuje samotné aktívum na **Komponent Industry 4.0**.
- Využíva **modulárnu štruktúru založenú na Submodeloch** pre detailný popis rôznych aspektov aktíva.
- Zabezpečuje **jednoznačný význam dát** pomocou sémantických identifikátorov (semanticId).
- Výmena informácií môže prebiehať staticky (pomocou .aasx súboru) a dynamicky prostredníctvom API alebo Jazyka Industry 4.0.
- Pre návrh a prácu s AAS existujú nástroje ako AASX Package Explorer, alebo priamo implementačné platformy Eclipse BaSyx či OPC UA.
- Je základným stavebným pilierom pre dosiahnutie konceptu **Plug&Produce** v priemyselnom prostredí.

Užitočné videonávody

Pre lepšie pochopenie práce s nástrojom AASX Package Explorer a konceptom AAS sa odporúča zhliadnutie série oficiálnych videonávodov (v anglickom jazyku) priamo od vývojárov tohto nástroja: **Screencasts about Asset Administration Shell (AAS)**

Literatúra

[1] Pribiš, R.; Beňo, L.; Drahoš, P. Asset Administration Shell Design Methodology Using Embedded OPC Unified Architecture Server. *Electronics* **2021**, *10*, 2520. Dostupné online: `https://www.mdpi.com/2079-9292/10/20/2520`

[2] ZVEI - German Electrical and Electronic Manufacturers' Association. *Examples of the Asset Administration Shell for Industrie 4.0 Components - Basic Part*. White Paper, April 2017. Dostupné online: `https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2017/April/Asset_Administration_Shell/ZVEI_WP_Verwaltungschale_Englisch_Download_03.04.17.pdf`

[3] Plattform Industrie 4.0. *Asset Administration Shell Discovery in an Industrie 4.0 System Environment*. Discussion Paper, April 2024. Dostupné online: `https://lni40.de/wp-content/uploads/2024/04/202404_Discussion_Paper_Discovery.pdf`

[4] OPC Foundation. *OPC UA for Asset Administration Shell (Version 1.0.0)*. Reference Document. Dostupné online: `https://reference.opcfoundation.org/I4AAS/v100/docs/`

[5] Bitkom; VDMA; ZVEI. *Implementation Strategy Industrie 4.0: Report on the Results of the Industrie 4.0 Platform*. 2016. Dostupné online: `https://www.bitkom.org/Bitkom/Publikationen/`

[6] Pribiš, R. *Komponent Industry 4.0 a jeho digitálna reprezentácia pomocou Asset Administration Shell*. Učebný text, FEI STU v Bratislave, 2021. Dostupné online: `https://elearning.mechatronika.cool/blog/`