



Evropská unie
Evropský sociální fond
Operační program Zaměstnanost



MINISTERSTVO VNITRA
ČESKÉ REPUBLIKY

C3V8 - Návrh zajištění podpory dokumentace datových sad generované z ontologických konceptuálních datových modelů agend v NKOD

Vytvořeno v rámci projektu

**Rozvoj datových politik v oblasti zlepšování kvality
a interoperability dat veřejné správy**
CZ.03.4.74/0.0/0.0/15_025/0013983

Klíčová aktivita: 04: Návrhy a realizace opatření pro zvyšování povědomí
o otevřených datech

Indikátor: 8 05 00 Počet napsaných a zveřejněných analytických a strategických
dokumentů (vč. evaluačních)

Verze výstupu: 01



Návrh zajištění podpory dokumentace datových sad generované z ontologických konceptuálních datových modelů agend v NKOD

V tomto dokumentu popisujeme návrh zajištění podpory dokumentace datových sad v NKOD generované z ontologických konceptuálních datových modelů agend, kterému zde říkáme sémantický slovník pojmů.

Návrh úprav NKOD

Z Portálu otevřených dat data.gov.cz ze sekce pro poskytovatele dat může vést odkaz na instanci vyvinutého prototypu, která poběží nezávisle na NKOD na serverech MV.

Uživatelské rozhraní prototypu

V této sekci představujeme uživatelské rozhraní navrženého a implementovaného prototypu. Má tři hlavní část - manažer specifikací, detailní pohled na specifikaci a editor schématu.

Manažer specifikací

První částí je manažer specifikací. Každá datová sada nebo prepoužitelná část datové sady se zde nazývá *specifikace*, a může být samostatně editovatelná. Manažer specifikací tedy umožňuje pracovat na více datových sadách (jejich specifikacích) naráz. Níže vidíme screenshot seznamu specifikací dostupných v dané instanci prototypu.

Ustanovení právních předpisů RPP	DETAIL	DELETE
Zařazení subjektu do kategorie RPP	DETAIL	DELETE
Reference na adresní místo RPP	DETAIL	DELETE
Informační systémy veřejné správy RPP	DETAIL	DELETE
Politická strana	DETAIL	DELETE

Konkrétní specifikace

Pro konkrétní specifikaci prototyp nabízí detailní přehled. Ten obsahuje seznam datových struktur, které budou dokumentovány, a seznam specifikací, které jsou aktuální specifikací prepoužívané (reused). Pokud specifikace bude obsahovat více datových struktur, budou zde vypsané. Tato obrazovka také obsahuje tlačítko "Generate .zip file", pomocí kterého se generují veškeré artefakty (dokumentace, schémata).



Data specification:

Informační systémy veřejné správy

MODIFY

RPP

Data structures

+ CREATE NEW

Name	Resources	Actions
Informační systém veřejné správy	19	EDIT DELETE

Reused data specifications

SET REUSED DATA SPECIFICATIONS

Name	Actions
Ustanovení právních předpisů	DETAIL

Generate artifacts

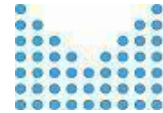
GENERATE .ZIP FILE

Report a bug on [GitHub](#). | Version: main@fba9be8 2022-04-19 22:49:07

Editor datové struktury

Největším detailem je pak editor konkrétní datové struktury. Zde má uživatel možnost vybrat podmnožinu entit sémantického slovníku pojmů a uspořádat ji do hierarchie, ze které bude generována lidsky čitelná dokumentace a JSON schéma.

V prázdné specifikaci uživatel nejprve vybere kořen - entitu ze sémantického slovníku pojmů. Následně přidává atributy a ostatní entity za asistence prototypu dle jejich propojení v sémantickém slovníku pojmů, specifikuje datové typy, technické názvy a kardinality.



Add interpreted surroundings

Ancestors

- Informační systém veřejné správy ⓘ
legislativní 365/2000 Sb.
- Information System ⓘ
veřejný-sektor

Attributes

- má datum poslední změny ISVS** agendový 104 ⓘ
Datum poslední změny ISVS.
- Má identifikátor ISVS** agendový 104 ⓘ
Jednoznačná identifikace ISVS v rámci RPP.
- má verzi ISVS** agendový 104 ⓘ
Určení provozované verze ISVS.
- charakteristika ISVS** legislativní 329/2020 Sb. ⓘ
Stručný popis ISVS, charakteristika účelu ISVS spolu s charakteristikou zpracovávaných dat a použitá architektura...
- datum likvidace ISVS** legislativní 329/2020 Sb. ⓘ
Datum trvalého ukončení poskytování služeb ISVS.
- datum vytvoření ISVS** legislativní 329/2020 Sb. ⓘ
Datum způsobilosti ISVS poskytovat služby uživateli ISVS v produkčním provozu.
- komentář k ISVS** legislativní 329/2020 Sb. ⓘ
Další informace, které správce ISVS považuje za nutné uvést o ISVS. V případě, že došlo k trvalému ukončení pos...
- název ISVS** legislativní 329/2020 Sb. ⓘ
Název ISVS, který používá správce ISVS.
- Provozní dokumentace** legislativní 365/2000 Sb. ⓘ

Associations

- patří do kategorie ISVS** agendový 104 → Kategorie ISVS ⓘ
- je spravován na základě ustanovení** legislativní 329/2020 Sb. → Ustanovení právního předpisu ⓘ
Právní předpis, který upravuje správu, popřípadě provozování a náležitosti ISVS, nebo agendu, pro jejíž výkon je ISVS vytvořen
- má aplikační prvek** legislativní 329/2020 Sb. → Aplikační prvek ⓘ
Přidružuje ISVS jeho aplikační prvek. ISVS může mít více aplikačních prvků.

Výsledkem může být například struktura pro popis Informačního systému veřejné správy, jako na obrázku níže.

slovník.gov.cz

ARTIFACTS ▼

SET ROOT ELEMENT

Informační systém veřejné správy ✎

Data structure for Informační systém veřejné správy. Informačním systémem veřejné správy funkční celek nebo jeho část zabezpečující cílevědomou a systematickou informační činnost pro účely výkonu veřejné správy nebo plnění jiných funkcí státu nebo dalších veřejnoprávních korporací. Každý informační systém veřejné správy zahrnuje data, která jsou uspořádána tak, aby bylo možné jejich zpracování a zpřístupnění, provozní údaje a dále technické a programové prostředky, případně jiné nástroje umožňující výkon informačních činností

Informační systém veřejné správy (informační_systém_veřejné_správy) + ✎

- **Má identifikátor ISVS** (identifikátor): String [1..1]
- **název ISVS** (název): Text [1..1]
- **datum vytvoření ISVS** (vytvoření): Date [0..1]
- **datum likvidace ISVS** (likvidace): Date [0..1]
- **má datum poslední změny ISVS** (poslední-změna): Date [0..1]
- **má verzi ISVS** (verze): Date [0..1]
- 🔍 **má kontakt pro informace o ISVS: Contact** (kontakt) [0..1]
- **charakteristika ISVS** (charakteristika): Text [0..1]
- **komentář k ISVS** (komentář): Text [0..1]
- 🔍 **je spravován na základě ustanovení**: [refers to] **Ustanovení právního předpisu** (je_spravován_na_základě_ustanovení) [0..*]
- 🔍 **poskytuje služby pro výkon agendy: Agenda** (agendy) [0..*]
- ☰ **má typ etapy životního cyklu ISVS: Typ etapy životního cyklu ISVS** (aktuální-typ-etapy) [0..1]
- 🔍 **má etapu životního cyklu ISVS: Etapa životního cyklu ISVS** (má_etapu_životního_cyklu_isvs) [0..*]



Výsledná generovaná dokumentace datové sady

V této části prezentujeme ukázkou výstupních artefaktů generovaných prototypem.

Lidsky čitelná dokumentace

Dokumentace pro čtení uživateli dat, vývojáři aplikací a dodavateli poskytovatele dat. V úvodu obsahuje grafický přehled dokumentované datové sady generovaný pomocí nástroje PlantUML.

TABLE OF CONTENTS	
1	Konceptuální model
1.1	Informační systém veřejné správy
1.1.1	Má identifikátor ISVS
1.1.2	název ISVS
1.1.3	datum vytvoření ISVS
1.1.4	datum likvidace ISVS
1.1.5	má datum poslední změny ISVS
1.1.6	má verzi ISVS
1.1.7	charakteristika ISVS
1.1.8	komentář k ISVS
1.1.9	Vztah: má kontakt pro informace o ISVS
1.1.10	Vztah: je spravován na základě ustanovení
1.1.11	Vztah: poskytuje služby pro výkon agendy
1.1.12	Vztah: má typ etapy životního cyklu ISVS
1.1.13	Vztah: má etapu životního cyklu ISVS
1.1.14	Vztah: má etapu životního cyklu ISVS
1.2	Kontakt
1.2.1	Vztah: má kontakt pro informace o ISVS
1.3	Ustanovení právního předpisu
1.3.1	Vztah: je spravován na základě ustanovení
1.4	Agenda
1.4.1	Vztah: poskytuje služby pro výkon agendy
1.5	Typ etapy životního cyklu ISVS
1.5.1	Vztah: má typ etapy životního cyklu ISVS
1.6	Etapa životního cyklu ISVS
1.6.1	Vztah: má etapu životního cyklu ISVS
1.6.2	Vztah: má etapu životního cyklu ISVS
2	Informační systém veřejné správy

Informační systémy veřejné správy

Living Standard, 20 April 2022

Editor:

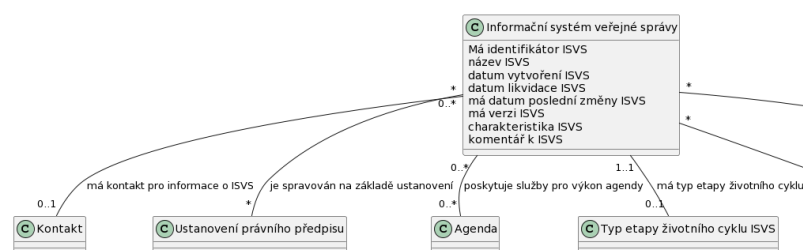
Model-Driven Generator

Abstract

Tento dokument je otevřenou formální normou ve smyslu § 3 odst. 9 zákona č. 106/1999 Sb., o svobodném přístupu k informacím, pro zveřejňování číselníků. Norma popisuje konceptuální model číselníků a stanovuje podobu jejich reprezentace ve strojově čitelné podobě ve formátech JSON-LD [json-ld11], a tedy i JSON [ECMA-404], a CSV [rfc4180] v denormalizované i normalizované podobě. Jednotlivé způsoby reprezentace číselníků také demonstruje na příkladech.

§ 1. Konceptuální model

V této sekci je definován konceptuální model.



§ 1.1. Informační systém veřejné správy

Pro každou položku datové sady pak obsahuje detailní specifikaci - jméno, popis, povinnost, kardinalitu, význam a odkaz do sémantického slovníku pojmů.



§ 1.1.3. datum vytvoření ISVS

Jméno

datum vytvoření ISVS

Popis

Datum způsobilosti ISVS poskytovat služby uživatelům ISVS v produkčním provozu.

Povinnost

Nepovinná

Kardinalita

0 - 1

Význam

Vlastnost datum vytvoření ISVS je definována v sémantickém slovníku pojmů jako [datum vytvoření ISVS](#).

JSON Schema

V této sekci uvádíme úryvek generovaného schématu v jazyce JSON Schema odpovídajícímu části dokumentované datové sady. Toto schéma je možné použít pro validaci dat jak uživatelů, tak poskytovatelů, například v nástroji [JSON Schema Validator](#), [Jim Blackler's JSON tools](#) nebo [Altova XMLSpy](#).

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "title": "Informační systém veřejné správy",
  "description": "Informačním systémem veřejné správy funkční celek nebo jeho část zabezpečující cílevědomou a systematickou informační činnost pro účely výkonu veřejné správy nebo plnění jiných funkcí státu anebo dalších veřejnoprávních korporací. Každý informační systém veřejné správy zahrnuje data, která jsou uspořádána tak, aby bylo možné jejich zpracování a zpřístupnění, provozní údaje a dále technické a programové prostředky, případně jiné nástroje umožňující výkon informačních činností",
  "type": "object",
  "required": [
    "identifikátor",
    "název"
  ],
  "properties": {
    "identifikátor": {
      "title": "Má identifikátor ISVS",
      "type": "string"
    },
    "název": {
      "title": "název ISVS",
```



```
"description": "Název ISVS, který používá správce ISVS.",
"type": "object",
"required": [],
"properties": {
  "cs": {
    "title": "Hodnota v českém jazyce",
    "type": "string"
  },
  "en": {
    "title": "Hodnota v anglickém jazyce",
    "type": "string"
  }
},
"vytvoření": {
  "title": "datum vytvoření ISVS",
  "description": "Datum způsobilosti ISVS poskytovat služby
uživateli ISVS v produkčním provozu.",
  "type": "string",
  "format": "date"
},
"likvidace": {
  "title": "datum likvidace ISVS",
  "description": "Datum trvalého ukončení poskytování služeb
ISVS.",
  "type": "string",
  "format": "date"
},
"poslední-změna": {
  "title": "má datum poslední změny ISVS",
  "description": "Datum poslední změny ISVS.",
  "type": "string",
  "format": "date"
},
}
```

Testování použitelnosti

V této sekci se věnujeme testování použitelnosti prototypu pomocí standardní metodiky [System Usability Scale \(SUS\)](#). Testovací uživatelé nejprve prototyp vyzkoušeli na základě



scénáře, a pak vyplnili dotazník založený na SUS. Pro plné využití potenciálu generování dokumentace je třeba se orientovat v datových formátech a datovém modelování.

Scénář

V tomto scénáři vytvoříme dokumentaci a související artefakty (schémata) pro jednoduchou datovou sadu Sportoviště. Toto téma již je namodelováno v Sémantickém slovníku pojmů, tedy máme vše, co pro specifikaci datové sady a generování její dokumentace budeme potřebovat.

V naší datové sadě bude mít Sportoviště následující atributy:

- Podmínky užívání
- Provozní řád
- Adresu (text)
- Provozovatele - právnickou osobu

Provozovatel má identifikační číslo osoby.

Testovací scénář

1. Pracovat budeme s nástrojem na adrese <https://demo.dataspecer.com/>, který si teď otevřete v novém okně.
2. V seznamu existujících specifikací pravděpodobně již naleznete nějakou specifikaci pro Sportoviště, my si ale vytvoříme novou pomocí tlačítka "Create new". V dialogu si specifikaci pojmenujeme. Po vytvoření se specifikace ukáže v seznamu. Klikneme u ní na "Detail".
3. V detailu specifikace vidíme její název, datové struktury, které specifikuje, specifikace, které přepoužívá a tlačítka "Generate .ZIP file", které na základě specifikace vygeneruje výstupní artefakty, tedy dokumentaci, schémata, atd.
4. Vytvoříme novou datovou strukturu tlačítkem "Create new". Uvidíme prázdnou specifikaci datové struktury. Tu je teď třeba naplnit tím, co chceme, aby v datové struktuře bylo, viz seznam atributů výše.
5. Nejprve je třeba zvolit kořen tlačítkem "Zvolit kořen". Tvoříme strukturu pro Sportoviště, tedy jako kořenem bude Sportoviště z datového slovníku.
6. Pod kořen chceme přidat potřebné atributy. Můžeme tak udělat tlačítkem "+".
7. Podmínky užívání a provozní řád vidíme přímo jako atributy Sportoviště, tedy je zvolíme.
8. Co se týče Adresy, v levé části dialogu vidíme rodičovské třídy Sportoviště, konkrétně Místo. Klikneme na něj. Vidíme, že místo může mít Adresu. Díky tomu, že Sportoviště od Místa dědí, Adresu může mít i Sportoviště. Vybereme jí.
9. Sportoviště také dědí od Veřejného místa, které má Provozovatele. Vybereme ho.
10. U tlačítka vpravo dole vidíme "Přidat (4)" - máme vybrány 4 položky, které kliknutím přidáme pod kořen.



11. Nyní musíme specifikovat, že u Adresy chceme v datové sadě mít pouze text. Tedy u vazby "má adresu: adresa" opět klikneme na "+" a přidáme "text adresy".
12. U Provozovatele chceme říct, že to má být právnická osoba. Když klikneme na "+", žádný obsah k použití nevidíme. Je to proto, že sám provozovatel žádné atributy nemá. Má ale řadu specializací (v dědičnostní hierarchii), a jednou z nich je Provozovatel jako právnická osoba. Pro nahrazení třídy její specializací slouží tlačítko tří teček a volba "Replace along inheritance". Tam vybereme "Provozovatel jako právnická osoba".
13. Nyní již víme, že je náš provozovatel sportoviště je právnickou osobou. A tedy má identifikační číslo osoby (IČO), které přidáme pomocí "+". IČO však není definováno pouze pro provozovatele, ale pro Osobu dle zákona 111/2009 Sb. Tedy klikneme na tu, a přidáme identifikační číslo osoby.
14. Tím je naše specifikace hotová. Přejdeme zpět do manažeru specifikací a vygenerujeme výstupní artefakty pomocí "Generate .ZIP file".
15. Výsledný soubor artifact.zip pak obsahuje vygenerovanou HTML dokumentaci a XML, JSON a CSV schémata.

Výsledky dle metodiky SUS

Počet testovacích uživatelů: **3**

Výsledné skóre a jeho interpretace: **65** - lehce podprůměrné skóre, jednomu z uživatelů prototyp evidentně nesedl.

Jednotlivé výsledky

Už./O.	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	Skóre
1	4	3	4	3	4	3	3	3	3	3	82.5
2	4	3	3	4	3	2	4	3	3	4	82.5
3	1	1	1	1	4	1	1	1	1	0	30

Přílohy

Zdrojové kódy prototypu z GitHubu jsou v příloze [model-driven-data-main.zip](#).