



UNIVERZITA PARDUBICE

Dopravní fakulta Jana Pernera

Katedra dopravních prostředků a diagnostiky
Oddělení kolejových vozidel

EXPERTNÍ SW PRO INTELIGENTNÍ DIAGNOSTIKU POJEZDU KOLEJOVÝCH VOZIDEL

Technická dokumentace výsledku typu SOFTWARE (R)

Číslo zprávy: **CK02000218-V2**

Projekt: **Wayside diagnostika pojezdu kolejových vozidel**
CK 02000218
Technologická agentura České republiky

Řešitel za UPCE: **Ing. Jakub Vágner, Ph.D.**

Odpovědná osoba za STARMON: **Ing. Ondřej Kovář**

Zpracovali: Vágner J.

Za období řešení: leden 2021 až červen 2024
Datum vydání: 25. 6. 2024
Počet stran textu: 27
Počet stran příloh: 0



Program **Doprava 2020+**

Obsah

Identifikace výsledku.....	3
Verze dokumentu.....	4
Seznam symbolů a zkratk	5
1 Popis SW DIPO / DIMET	6
1.1 Modifikace SW	6
1.2 Návaznost na HW	6
2 Účel a funkce SW	7
2.1 Technické předpoklady pro instalaci SW	8
3 Popis funkce jednotlivých modulů.....	9
3.1 Topologie systému	9
3.2 Struktura DB tabulek	9
4 Vstupní data	11
4.1 Sběr dat v koleji.....	11
4.2 Formát výstupních dat z měření	11
4.3 SOL_SPC, SOL_NAP, SOL_STAT	13
4.3.1 Vstupy a výstupy SOL_SPC.....	16
4.3.2 Vstupy a výstupy SOL_NAP	18
4.3.3 Vstupy a výstupy SOL_STAT.....	19
4.4 SOL_NST	19
4.5 SOL_NSD.....	22
4.6 WEB-CONN.....	22
Literatura.....	26

Identifikace výsledku

Řešeno v projektu	TAČR CK 02000218
Název projektu	Wayside diagnostika pojezdu kolejových vozidel
Název výsledku	Expertní SW pro inteligentní diagnostiku pojezdu kolejových vozidel
Veřejná soutěž	2.VS DOPRAVA 2020+ (STA02021CK020)
Období řešení	01. 01. 2021 – 31. 6. 2024
Hlavní příjemce	Univerzita Pardubice
Další příjemce	STARMON s.r.o.
Číslo výsledku	CK02000218-V2
Typ výsledku	R - software
Datum dosažení výsledku	05/2024
Původci výsledku	80 % - Univerzita Pardubice 20 % - STARMON s.r.o.
Možnosti využití	K využití je potřeba vždy získat licenci, kontaktní osoba: Ing. Jakub Vágner, Ph.D., Univerzita Pardubice, Studentská 95, 532 10 Pardubice, jakub.vagner@upce.cz

Podle zapojení na tvorbě výsledku je podíl na výsledku 80 % Univerzita Pardubice a 20 % Starmon dle následujícího rozdělení:

CK02000218-V2	podíl části na celku	podíl na části		podíl na celku	
		UPCE	STARMON	UPCE	STARMON
SW - SOL-SPC	15%	80%	20%	12.00%	3.00%
SW - SOL-NAP	15%	80%	20%	12.00%	3.00%
SW - SOL-STAT	20%	80%	20%	16.00%	4.00%
SW - SOL-NST	20%	100%	0%	20.00%	0.00%
SW - SOL-NSD	10%	100%	0%	10.00%	0.00%
SW - SOL-DAQ	20%	50%	50%	10.00%	10.00%
suma	100%			80.00%	20.00%

Verze dokumentu

Tato dokumentace se váže k následujícím verzím SW modulů. Verze SW se zobrazuje v logu při spuštění a pokud má SW GUI, pak i tam. Při dalším vývoji se budou funkcionality měnit, při vydání nové verze bude manuál korigován příp. doplněn.

Verze tohoto dokumentu	Verze 1
SOL_SPC, SOL_NAP, SOL_STAT	1.6.2
SOL_NST	1.2.0
SOL_NSD	1.2.0
WEB-CONN	1.0.1
Společné knihovny	9.0.0

Seznam zkratk

BACK-END	SW nástroje pro automatické zpracování dat bez zásahu uživatele
DAQ	Digital Acquisition – digitální záznam dat
DB	Databáze (v tomto případě typu MY SQL)
DFJP	Dopravní fakulta Jana Pernera
DiPo	system „Diagnostika Pojezdu“ pro obecné použití
DiMet	system „Diagnostika Metra“ – modifikace systému DiPo pro metro
FRONT-END	Programy určené pro komunikaci s uživatelem pomocí GUI
GUI	Graphic User Interface – grafické prostředí programu.
HW	Hardware
ID	Identifikace měření (číslo měření)
INI	konfigurační soubor v textovém formátu ASCII
IPC	Industrial PC – průmyslový počítač
JSON	JavaScript Object Notation – typ souboru
MYSQL	dotazovací jazyk pro databáze, resp. typ relační databáze
NS	Neuronová síť
OS	Operační systém
POST	Dotazovací metoda HTTP protokolu
RFID	Radio Frequency Identification – bezdrátová identifikace
STARMON	STARMON s.r.o.
SW	Software
TAČR	Technologická agentura České republiky
UPCE	Univerzita Pardubice
VPN	Virtuální privátní síť

1 Popis SW DIPO / DIMET

1.1 Návaznost na měřící HW

SW byl vytvořen zejména pro použití s HW, který vznikl v souvislosti s výsledkem č. CK02000218-V1 projektu TAČR CK02000218, ale za splnění potřebných parametrů lze použít i jiný HW pro sběr dat. Podmínkou je splnění požadavků na vstupní data, parametrů měření a požadavků na konfiguraci snímačů.

1.2 Modifikace SW

Název SW DIPO vychází ze slov **D**iagnostika **P**ojezdu – v tomto případě se jedná o variantu umožňující diagnostikovat pouze výstrahy při poruše projíždějícího vozidla, a to zejména při poruše na jízdni ploše kol.

Název SW DIMET vychází ze slov **D**iagnostika **M**ETra – a jedná se o rozšířenou verzi SW DIPO, která navíc umožňuje diagnostiku definované flotily vozidel – v tomto případě flotily vozidel metra.

Použití systému DIMET je podmíněno možností identifikace vozidel (nejlépe přímo na trati), kdežto systém DIPO lze provozovat i bez nutnosti identifikace vozidel. Poruchy jsou pak reportovány časem průjezdu a počtem nalezených náprav.

2 Účel a funkce SW

SW je určen pro diagnostiku pojezdu kolejových vozidel, a to pomocí analýzy vibrací měřených na kolejnicích při průjezdu vozidla.

Celý systém diagnostiky je vytvořen jako distribuovaný, obsahující několik modulů, které mezi sebou komunikují. Moduly jsou dle funkce a místa rozděleny na:

- 1) **DAQ** – měření dat v koleji (více míst) vč. odesílání dat; identifikace vlaků – běží v měřícím PC v místě měření;
- 2) **BACK-END** – příjem dat, archivace v DB, předzpracování výsledků pro diagnostiku, automatické vyhodnocení, zpracování výsledků pro vizualizaci, sběr dat z jiných zdrojů a databází – obecně automatické SW moduly, které zpracovávají data;
- 3) **FRONT-END** – vizualizace výsledků, interakce s uživateli (např. zadávání dat z údržby), logování funkce systému, vzdálený dohled nad DAQ; webové rozhraní (server UPCE) a desktopové aplikace.

Popisované moduly spadají do kategorie BACK-END a v následující tabulce je uveden jejich přehled:

Název modulu	Popis funkce
SOL_SPC	Výpočet spektrogramů a vytvoření podkladů pro webovou prezentaci spektrogramů
SOL_NAP	Výpočet mezivýsledků ve frekvenční oblasti pro jednotlivé nápravy a celé průjezdy
SOL_STAT	Výpočet statistických charakteristik signálů a jejich výřezů v okolí náprav
SOL_NST	Modul pro trénování NS na vybraných datech z údržby nebo z provozu
SOL_NSD	Modul pro detekci příznaků pomocí natrénované NS
WEB-CONN	Serverová aplikace umožňující upload, download dat od uživatelů a solverů; lze přenášet také log zprávy a další metadata; umožňuje také konfiguraci DAQ pro administrátory

2.1 Technické předpoklady pro instalaci SW

Pro chod všech modulů je potřeba:

- 1) Databázový server MaraDB MySQL verze 10.4.28 nebo vyšší [4].
- 2) Webový server Apache verze 2.4.56 (WIN64) nebo vyšší s PHP v 8.2.4 nebo vyšší [3].
- 3) Sdílený souborový systém pro uložení měřených dat.
- 4) Virtuální PC s OS Win 10 64bit nebo vyšší.

Výše uvedené 4 body lze realizovat jako jeden server nebo rozdělit do několika jednotlivých serverů.

Prostor v databázi je nutné dimenzovat na potřebný objem dat. Ten lze odhadnout takto:

$$2 \times CH \times NAP \times 250B + VLAK \times 300B$$

Kde: CH – počet měřených kanálů
NAP – počet projetých náprav za dané období
VLAK – počet průjezdů vlaků za dané období

Např. pro metro za jeden den: CH = 2, NAP = 250x20 = 5000 / den, VLAK = 250, tedy odhad objemu je:

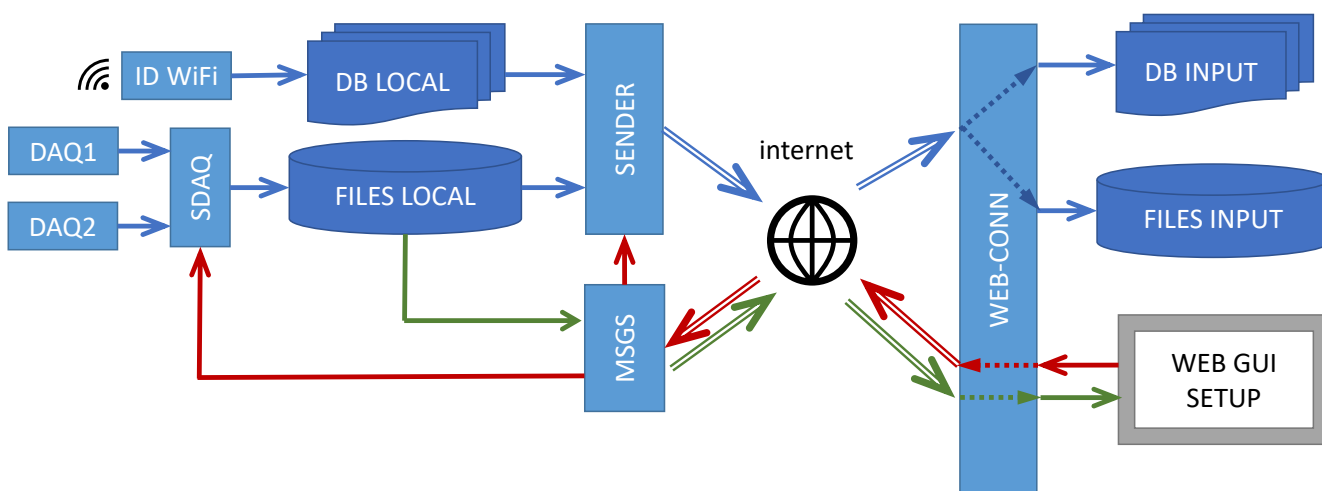
$$\begin{aligned} 2 \times 2 \times 5000 \times 250B + 250 \times 300B &= 5\,000 \text{ kB} + 75 \text{ kB} \\ &= \text{cca } 5 \text{ MB za den} \\ &= \text{cca } 2 \text{ GB za rok} \end{aligned}$$

Objem dat v souborovém systému se špatně odhaduje, protože nesouvisí pouze s množstvím vlaků (a náprav), ale také s jejich rychlostí (pomalejší průjezd delšího vlaku = větší objem dat). U metra se jedná o cca 1 GB/den při 250 vlacích, u konvenční trati se jedná o 0,5 GB/den při 150 vlacích za den. Přesnější odhad lze stanovit při znalosti hustoty provozu. **Jedná se o zálohu primárních naměřených dat, které lze po vyhodnocení postupně odmazávat.**

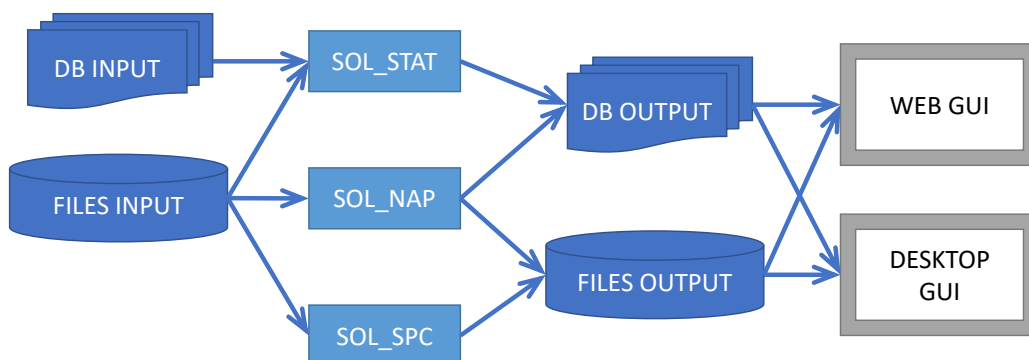
3 Popis funkce jednotlivých modulů

3.1 Topologie systému

Jednotlivé moduly si předávají data přímo (např. SENDER -> WEB-CONN) nebo pomocí DB. Topologie modulů pro měření, přenos a archivaci dat je zobrazena na Obr. 1, na Obr. 2 je topologie modulů pro vyhodnocení a vizualizaci dat. Obě části jsou tedy propojeny pomocí DB INUT a FILES INPUT.



Obr. 1 Topologie systému DiPo – část pro měření, přenos a archivaci dat z měřícího/měřících IPC.



Obr. 2 Topologie systému DiPo – část vyhodnocení a vizualizaci dat.

3.2 Struktura DB tabulek

Strukturu tabulek je nutné dodržet a SW si tabulky sám nevytvoří. Tabulky jsou vždy označeny „doména_tabulka“, kde „doména“ označuje buďto lokalitu nebo organizaci, kde se měření realizuje.

POZOR! Vyhodnocení nelze provádět skrze více domén, ale v každé doméně může být použito více měřících zařízení. Každá doména má také svou databázovou tabulku flotily vozidel pro diagnostiku.

Při dodání SW bude přiložen skript ve formátu MYSQL, který po zadání názvu domény příslušné tabulky vytvoří při instalaci nové domény.

Pro chod SW jsou potřeba tyto tabulky (kde **X** je název domény):

Název a popis	Přehled uložených dat
X_config	Nastavení SW, HW a režimů přenosu dat
X_log	Ukládání událostí v časové ose a s příznakem důležitosti
X_files	tabulka souborů, které byly na server nahrány.
X_pass	tabulka průjezdů v čase vč. identifikace vozidel (pokud se uplatňuje)
X_trains	tabulka vlakových souprav ve sledované flotile
X_trains_health	tabulka výsledků pro jednotlivá vozidla ve formě hodnotícího parametru (jedna tabulka pro různé typy poruch), přiřazen čas vložení
X_trains_km *	Tabulka nájezdu kilometrů pro sledovaná vozidla/soupravy – pouze pro informaci
X_trains_revize *	Tabulka provedených revizí pro sledovaná vozidla/soupravy – pouze pro informaci
X_trains_history *	Tabulka událostí pro sledovaná vozidla/soupravy – pouze pro informaci
X_stat_z1, X_stat_z2, X_stat_???	Tabulka statistik pro jednotlivé snímače a nápravy. Záznamem je náprava přiřazená ID průjezdů a matice hodnot.

* - tabulky nepovinné sloužící pouze jako podklad pro vizualizaci

4 Vstupní data

4.1 Sběr dat v koleji

Pro sběr dat v koleji je potřeba rozmístit snímače zrychlení, a to nejméně 2 kusy. Vyšší počet lze použít pro zpřesnění výsledků nebo pro zálohu v případě poruchy. Dále je potřeba umístit detektor náprav, který detekuje nápravy projíždějících vozidel. Pro případ SW DIMET je nutné dále přidat HW modul umožňující identifikaci vozidel (např. RFID).

Zařízení musí umět realizovat funkci „pretrigger“, tedy spuštění měření ještě před příjezdem první nápravy k prvním snímači. Pretrigger musí mít dostatečný kruhový buffer – obvykle je potřeba uchovat cca 1 až 2 sekundy záznamu.

4.2 Formát výstupních dat z měření

Časové záznamy signálů musí být v binární podobě a to jako 32-bit INTEGER nebo 64-bit FLOAT hodnoty. Jejich vzorkování a absolutní počátek měření v čase je uložen v hlavičkových souborech metadat a to v podobě textových souborů INI (výstup programu DiMet-M) nebo JSON (výstup programu SDAQ). Rozvoj souborů INI se při dalším vývoji SW již dále nepředpokládá.

Výstupem měření (resp. vstupem pro vyhodnocení) musí být data v následujícím formátu:

- JSON – hlavičkový soubor pro daný průjezd
- I32 – datový soubor signálů
- IP – soubor identifikace průjezdu (volitelně)

Soubory musí být organizovány ve vstupním adresáři v této struktuře:

```
Adresář: ..\YYYY\YYYYMMDD\YYYYMMDD-hhnnss\..  
Soubor: ID_ YYYYMMDD-hhnnss_DAQ_CH
```

```
YYYY - rok; MM - měsíc; DD - den; hh - hodina; nn - minuta; ss - sekunda; ID -  
identifikátor HW; DAQ - identifikátor DAQ
```

Příklad názvů souborů a adresáře pro jeden průjezd dne 30. 5. 2024 v 06:18:13 je:

```
..\DATA\2024\20240530\20240530-061813\OPOS001_20240530-061813.json  
..\DATA\2024\20240530\20240530-061813\OPOS001_20240530-061813_DAQ1-ch1.I32  
..\DATA\2024\20240530\20240530-061813\OPOS001_20240530-061813_DAQ1-ch2.I32  
..\DATA\2024\20240530\20240530-061813\OPOS001_20240530-061813_DAQ2-ch1.I32  
..\DATA\2024\20240530\20240530-061813\OPOS001_20240530-061813_DAQ2-ch2.I32
```

Struktura hlavičkového souboru měření JSON je následující:

```
{
  "name": "DIPO",
  "Version_SW": "1.02",
  "Version_HW": "1.00",
  "DIPO_ID": "OPOS001",
  "date": "20240530-061813",
  "format": ".I32",
  "last sync timestamp": 1717041694.722046,
  "Vt": 146.16,
  "Trigger source DAQ": 2,
  "convert to V": 6.097777804879e-07,
  "Temperature": 9,
  "DAQ1": {
    "fsampl": 25600,
    "first ADC sampling time": 1717042682.435046,
    "internal time in DAQ [ms]": 989733,
    "channels": [
      {
        "name": "Z1Lp",
        "file": "OPOS001_20240530-061813_DAQ1-ch1.I32"
      },
      {
        "name": "Z2Lm",
        "file": "OPOS001_20240530-061813_DAQ1-ch2.I32"
      }
    ],
    "Axles": []
  },
  "DAQ2": {
    "fsampl": 25600,
    "first ADC sampling time": 1717042682.436046,
    "internal time in DAQ [ms]": 989731,
    "channels": [
      {
        "name": "Z1Pp",
        "file": "OPOS001_20240530-061813_DAQ2-ch1.I32"
      },
      {
        "name": "Z2Pm",
        "file": "OPOS001_20240530-061813_DAQ2-ch2.I32"
      }
    ],
    "Axles": [
      {
        "t": 1717042684.4400458,
        "V": 146.16
      },
      {
        "t": 1717042684.5220459,
        "V": 141.48
      }
    ]
  }
}
```

Struktura může mít následující odchylky:

- 1) Sekce „Axles“ je uvedena u té DAQ jednotky, která měří signál detekce náprav, resp. u té, která je uvedena v hodnotě „Trigger source DAQ“.
- 2) Pokud sekce „Axles“ není zahrnutá u žádné jednotky je průjezd vyhodnocen jako vydný.
- 3) DIPO_ID je identifikátor měřicího místa, dle tohoto názvu se pak data třídí
- 4) Sekce „Channels“ může obsahovat neurčený počet kanálů, předpokládá se 1 – 8 na jednu DAQ.

Struktura souborů I32 je binární. Jedná se o posloupnost 32-bit integer hodnot, které je nutno přepočíst na výsledný signál přepočtovou konstantou a offsetem. Každý kanál je uložen v jednom souboru. Soubory NEOBSAHUJÍ žádnou hlavičku, jedná se o RAW data DAQ převodníku.

Pro měření lze použít také výstup v podobě **dat v souborech R64**, pak se jedná o 64-bit float hodnoty. Jeden kanál v jednom souboru. Jedná se pak přímo o reálné hodnoty zrychlení v jednotkách m/s² již vypočtené v měřicím zařízení. Soubory R64 však mají 2x větší velikost a v případě 24-bit DAQ převodníku nepřinášejí vyšší přesnost.

4.3 SOL_SPC, SOL_NAP, SOL_STAT

Technicky se jedná o jeden spustitelný soubor, který v sobě ale obsahuje tři moduly. Při spuštění se zadá parametr, který funguje jako „přepínač“ určující, který modul pro vyhodnocení se má provést:

- SPC = výpočet spektrogramů včetně metadat pro zobrazení na webu,
- NAP = výpočet FFT spekter pro jednotlivé nápravy a další vyhodnocení ve frekvenční oblasti,
- STAT = výpočet statistických charakteristik pro jednotlivé nápravy,
- CHECK = kontrola integrity DB a souborového úložiště.

Dalším parametrem je označení kanálu, který se má vyhodnotit, tedy např. „Z1“ nebo se uvede znak „*“, který značí vyhodnocení všech kanálů. Avšak program se řídí také konfiguračním souborem, kde lze jednotlivé kanály zapnout, a navíc nastavit pro každý jinou konfiguraci výpočtu. Lze tak operativně vyhodnotit jen určité kanály a průjezdy.

Dále se uvádí, kolik průjezdů má program vyhodnotit a po dokončení se automaticky ukončí. To umožňuje výpočty spouštět v určitých dávkách a časových intervalech. Program automaticky najde, který průjezd je v pořadí k vyhodnocení dle jeho příznaku v tabulce X_pass. Mohou nastat stavy: vyhodnoceno/nehodnoceno/vyhodnocuje se/chyba vyhodnocení.

Spuštění solveru

Spuštění se provádí spuštěním EXE souboru s parametry. Pro automatické zpracování lze toto spuštění naplánovat v operačním systému. Vhodný interval je 15 min a NENÍ nutné nastavit „nespouštět novou instanci“. I když se zastaví chod programu nebo program nestihne v 15 minutách data vyhodnotit, lze spustit další instanci, protože díky příznakům v DB nemůže vzniknout kolize (vyhodnocení jednoho průjezdu paralelně).

Parametry spuštění jsou následující:

```
SOLVER.EXE AUTO SOL ID CH CNT

AUTO - automatické ukončení: 0 = ne / 1 = ano,
SOL - požadovaný solver: STAT/SPCNAP/FFTNAP,
ID - ID průjezdu pro vyhodnocení, při „*“ zvolí se následující nevyhodnocený,
CH - volba kanálu k vyhodnocení, při „*“ všechny zapnuté v INI konfiguraci
CNT - počet výpočtů, po kterém se SOLVER ukončí, při „0“ vypočte vše dostupné
```

Konfigurace solveru

Níže je uveden popis konfiguračního souboru, který obsahuje konfiguraci společnou pro všechny moduly a také nastavení jednotlivých modulů. Pokud se u modulu objevuje parametr zapnutí/vypnutí kanálu, pak i když je požadavek na vyhodnocení, tak se vyhodnocení neprovede. Modul tak lze spouštět s parametrem „*“ – vyhodnotit vše a v INI souboru nastavit požadované kanály. Naopak lze v INI nastavit všechny kanály jako zapnuté a spouštět solver pouze pro jednotlivé kanály. První varianta je výhodná v tom, že není potřeba řešit spuštění mnoha úloh po sobě a vyhodnocení je rychlejší.

```
[setup] # společné nastavení pro všechny moduly
logdir=LOG\ # adresář pro logování
logfile=%s_tmpSolver.log # šablona názvu souboru (%s čas.značka)
solve=DEF # odkaz na konfiguraci definic

[DEF] # konfigurace definic
maindir=x:\DiPo\Data\ # adresář pro zdrojová data
datadir=DATA # podadresář dat
resdir=RES # adresář pro výsledky
timsep=_ # oddělovač ID a času
chsep=- # oddělovač času
ext=R64 # přípona souborů pro vyhodnocení

[STAT] # podadresář výsledků modulu STAT
dir=SPCSTAT

[STAT_DB] # přístupové údaje k databázi
name=Lokální DB DiPo
host=127.0.0.1
user=auto
password=*****
database=dipo
timeout=5000

[STAT_CH] # přepínače kanálů
count=8 # počet
CH_name_1=Z1 # jméno 1.kanálu
CH_enab_1=1 # zap/vyp 1.kanál
. . .
CH_name_8=M3 # jméno 8.kanálu
CH_enab_8=0 # zap/vyp 8.kanál

[SPC]
```

```

title=Spektrogram          # název v hlavičce spektrogramu
dir=SPCNAP                  # adresář výsledků
L=2048                      # délka okna
K=256                       # krok okna
fmin=0.0                   # minimální frekvence
fmax=5000.0                # maximální frekvence
max=1                       # nastavení max. hodnoty škály
hann=1                     # použít hanning okno?
cb=1                        # použít 1=barevnou škálu/ 2=černobílou škálu

[SPC_DB]                    # přístupové údaje k databázi
name=Lokální DB DiPo
host=127.0.0.1
user=auto
password=*****
database=dipo
timeout=5000

[SPC_CH]                    # přepínač kanálů
count=8                     # počet

CH_name_1=Z1               # název 1.kanálu
CH_enab_1=1                # vyp/zap 1.kanálu
CH_maxi_1=1.0              # změna škály 1.kanálu
. . .
CH_name_8=M3               # název 8.kanálu
CH_enab_8=0                # vyp/zap 8.kanálu
CH_maxi_8=1.0              # změna škály 8.kanálu

[FTTNAP]                   # nastavení modulu NAP
title=fftnap
dir=fftnap
L=2048                      # délka okna
fmin=0.0                   # minimální frekvence
fmax=5000.0                # maximální frekvence
max=1                       # nastavení max. hodnoty škály
hann=1                     # použít hanning okno?

[FFTNAP_CH]                # přepínače kanálů 0=vyp/1=zap
count=8                     # počet
CH_enab_1=1                # vyp/zap 1.kanálu
. . .
CH_enab_8=0                # vyp/zap 8.kanálu

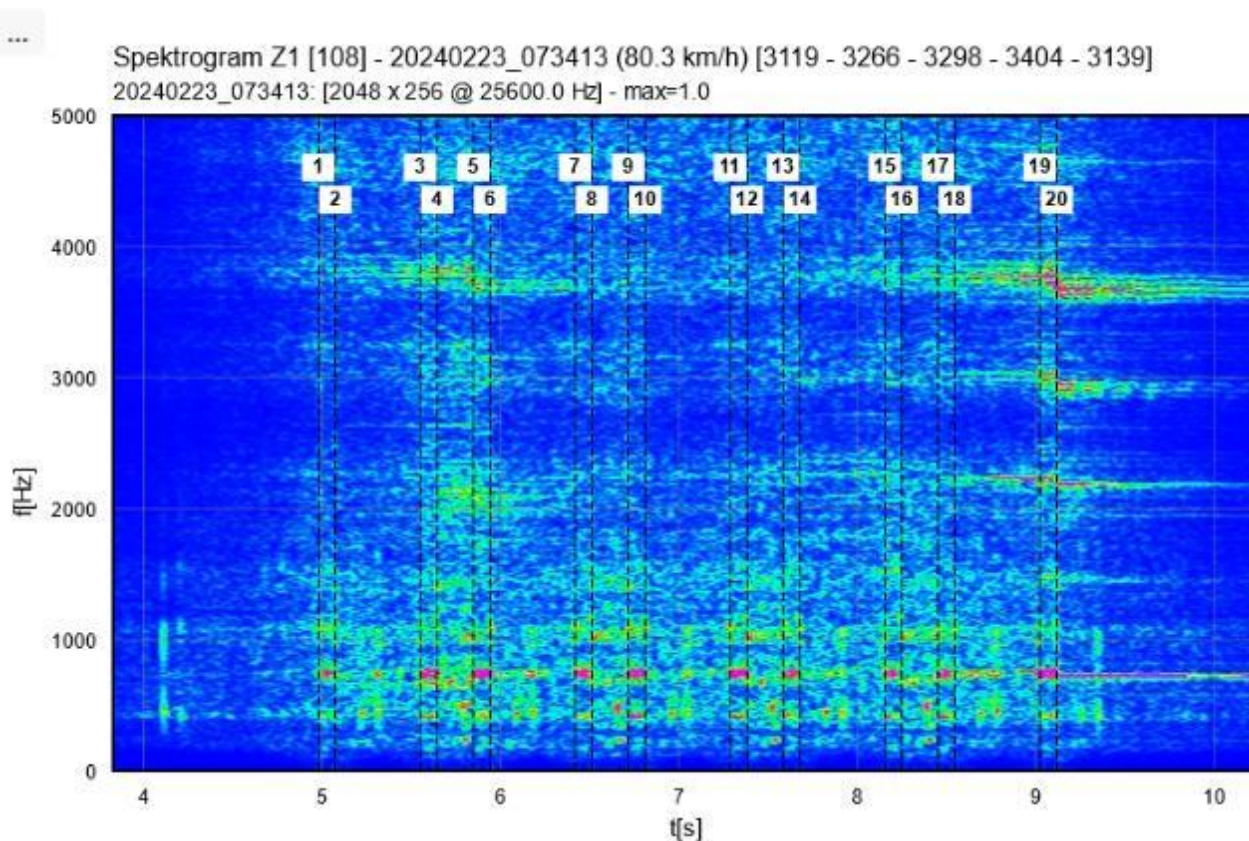
```

4.3.1 Vstupy a výstupy SOL_SPC

Modul SOL_SPC provádí vyhodnocení v časově-frekvenční oblasti. Jedná se zejména o tvorbu spektrogramů průjezdů. Spektrogramy slouží pro vizuální kontrolu průjezdů za účelem verifikace detekce a pro hledání zatím neznámé odezvy pro zatím nedetekované typy poruch.

Vstup:	1) Binární datové soubory R32 nebo R64 2) Hlavičkové soubory JSON nebo INI	Sdílený souborový systém IN-DIR Sdílený souborový systém IN-DIR
Parametry	3) Kanál a ID pro vyhodnocení 4) N, K, f_0 , f_1 , CB, ... 5) Výstupní soubor	Parametr „ID“ INI setup Automaticky generováno
Výstup:	1) Soubor spektrogramu JPG nebo PNG 2) Soubor metadat JSON	Sdílený souborový systém OUT-DIR Sdílený souborový systém OUT-DIR

Na Obr. 3 je vidět výstup spektrogramu generovaný ve webovém prohlížeči. Jedná se o prvek „canvas“, tedy jej lze následně uložit jako PNG soubor (včetně popisků).



Obr. 3 Ukázka generovaného spektrogramu.

Kód JSON pro spektrogram je v následujícím formátu:

```
{
  "souprava": "108",
  "CH": "Z1",
  "lokalita": "DE-BO_km15.002",
  "mac": "****",
  "ip": "****",
  "Title": "Spektrogram Z1 [108] - 20240223_073413 (80.3 km/h) [3119 - 3266 - 3298 - 3404 - 3139]",
  "subTitle": "20240223_073413: [2048 x 256 @ 25600.0 Hz] - max=1.0",
  "axisX": {
    "Title": "t[s]",
    "minimum": 3.8295311927795410E+000,
    "maximum": 1.0249530792236328E+001,
    "step": 1.0000000000000000E+000
  },
  "axisY": {
    "Title": "f[Hz]",
    "minimum": 0.0000000000000000E+000,
    "maximum": 5.0000000000000000E+003,
    "step": 1000
  },
  "wheels": [
    {
      "i": "1",
      "Vr": 8.0276351928710938E+001,
      "t": 4.9789843749999996E+000
    },
    {
      "i": "2",
      "Vr": 8.0278892517089844E+001,
      "t": 5.0737109374999996E+000
    },
    . . . další nápravy . . .
    {
      "i": "20",
      "Vr": 8.0387451171875000E+001,
      "t": 9.1166796874999996E+000,
      "h400": 0.0000000000000000E+000,
      "min": 0.0000000000000000E+000,
      "max": 0.0000000000000000E+000,
      "absmax": 0.0000000000000000E+000,
      "rms": 0.0000000000000000E+000,
      "mean": 0.0000000000000000E+000,
      "energy": 0.0000000000000000E+000,
      "sdv": 0.0000000000000000E+000,
      "kurtosis": 0.0000000000000000E+000,
      "skewness": 0.0000000000000000E+000,
      "crestf": 0.0000000000000000E+000
    }
  ],
  "spc": {
    "spc_file": "20240223_073413-Z1.jpg",
    "spc_tmin": 3.8295311927795410E+000,
    "spc_tmax": 1.0249530792236328E+001,
    "spc_min": 0.0000000000000000E+000,
    "spc_max": 1.0000000000000000E+000,
  }
}
```

```

"spc_L": 7.9999998211860657E-002,
"spc_K": 9.9999997764825821E-003,
"spc_fvz": 2.5600000000000000E+004,
"spc_height": 400,
"spc_width": 643,
"fft_file": "",
"fft_L": 4096,
"fft_f0": 0.0000000000000000E+000,
"fft_f1": 5.0000000000000000E+003,
"Lstat": 256,
"i0_400": 64,
"i1_400": 80
},
"temps": {
  "T_PC": 2.6500000000000000E+001,
  "T_backUP": 1.7250000000000000E+001,
  "T_backDOWN": 1.8250000000000000E+001,
  "T_frontUP": 1.5750000000000000E+001,
  "T_out": 1.4000000000000000E+001,
  "T_CPU1": 3.8000000000000000E+001,
  "T_CPU2": 3.7000000000000000E+001
}
}

```

Kde v hlavní sekci JSON jsou informace o soupravě a průjezdu, v sekcích axisX a axisY nastavení os, sekce „wheels“ obsahuje polohy dvojkolí včetně dalších informací, sekce SPC obsahuje informaci o nastavení spektrogramu (za jakých podmínek byl vypočten) a sekce temps nese doplňkovou informaci o teplotách v místě měření.

Soubor JSON tedy neobsahuje pouze informace nutné pro vizualizaci spektrogramu, ale také další doplňkové informace k měření a průjezdu. Toho lze do budoucna využít k dalším typům vizualizací. Hlavním důvodem je ale to, že soubory lze dále distribuovat včetně informací, o které by uživatel přišel, protože jsou uloženy jen v databázi. Jedná se tak o „dokumentaci“ průjezdu, kterou lze k posouzení zaslat emailem apod.

4.3.2 Vstupy a výstupy SOL_NAP

Modul vypočte frekvenční spektra z N hodnot pro každou nápravu a tyto uloží do sdíleného souborového systému. Zároveň ve spektrech detekuje příznaky a uloží je do tabulky výsledků.

Vstup:	1) Binární datové soubory R32 nebo R64 2) Hlavičkové soubory JSON nebo INI	Sdílený souborový systém IN-DIR Sdílený souborový systém IN-DIR
Parametry	1) Kanál a ID pro vyhodnocení 2) N, f0, f1 3) Výstupní soubor	Parametr „ID“ INI setup Automaticky generováno
Výstup:	1) Soubor spekter náprav FFTNAP 2) Záznam v DB v tabulce X_trains_health	Sdílený souborový systém OUT-DIR Databáze

Soubory FFTNAP se ukládají do výstupního adresáře výsledků s automaticky generovaným názvem. Adresář může být stejný jako výstup SPCNAP, nebo jej lze směřovat jinam. Výstup v databázi pak slouží rovnou pro vizualizaci časových řad.

4.3.3 Vstupy a výstupy SOL_STAT

Modul vypočte statistiku z N hodnot pro každou nápravu a hodnoty uloží do tabulky databáze X_stat_CH. Dále detekuje příznaky, které uloží do X_trains_health – ty lze dále vizualizovat do časové řady.

Vstup:	3) Binární datové soubory R32 nebo R64 4) Hlavičkové soubory JSON nebo INI	Sdílený souborový systém IN-DIR Sdílený souborový systém IN-DIR
Parametry	4) Kanál a ID pro vyhodnocení 5) N	Parametr „ID“ INI setup
Výstup:	3) Záznam v DB v tabulce X_stat_CH 4) Záznam v DB v tabulce X_trains_health	Databáze (CH = název kanálu) Databáze

Výstupem je matice hodnot pro každou nápravu. Tyto data jsou archivována v DB a dále je používají moduly SOL_NST a SOL_NSD.

Vr	rychlost nápravy
T	čas průjezdu
Min	min. signálu
Max	max. signálu
Absmax	max. odchylka od mean
Rms	rms signálu
mean	střední hodnota
energy	energie
sdv	směrodatná odchylka
kurtosis	hodnota kurtosis
skewness	špičatost signálu
crestf	crest faktor

Uvedené statistiky se vypočtou pro okno N vzorků, kdy střed okna je v čase průjezdu nápravy. Velikost okna lze měnit v INI souboru.

4.4 SOL_NST

SOL_NST je modul pro trénování NS, který na základě trénovacích množin „naučí“ NS detekovat příslušnou poruchu.

Vstup:	6) Trénovací množina naměřených dat párovaná s množina příznaků (soubory STAT a FFT_NAP) nebo metadata z databáze 7) Soubor topologie sítě (*.NNW)
Výstup (pro modul SOL_NSD):	5) Soubor vah (natrénovaná data) *.TRD 6) Soubor topologie sítě (*.NNW) 7) Soubor *.dat s průběhem trénování

Definice NS

Před trénováním je potřeba vydefinovat topologii NS. Nedá se přesně říct, jaká topologie je vhodná, pro každý případ to je individuální a je to předmětem zkoumání před zavedením dalšího typu detekce do systému. To se provede editací souboru *.NNW, který má následující strukturu:

```
[info]
name=detekce1           # nazev detekce
note=popis detekce     # zde lze upřesnit informace k detekci
nL=3                   # počet vrstev
nI=325                 # velikost vstupního vektoru
nT=1                   # velikost výstupního vektoru
nN[0]=325              # počet ve vrstvě 0
nN[1]=325              # počet ve vrstvě 1
nN[2]=1                # počet ve vrstvě 2
tf=0.15                # nastavení rychlosti trénování
te=0.0000197637419108639 # hodnota posledního trénování
count=82579            # počet kroků posledního trénování
data=D:\SOL_NST\DATA\detekce1.dat # datový soubor uchovávající průběh učení
```

Příprava trénovací sady

Dále je potřeba sestavit trénovací sadu. Jedná se o dvourozměrný vektor obsahující v páru ID průjezdu a příznak „tar“ (hodnota target). Tento vektor se sestaví databázovým dotazem v sekci **A**. Pro zjednodušení se dá v text SQL dotazů využít proměnných, které doplní hodnotu z tabulky C (ta je uložena v souboru „setup.INI“:

```
[train001]
rychlost=0.15
eps=2e-4
TRAINfile=train108a118_pohon_0_1
CH=Z1
souprava=
nap=19
odt=2022-10-01 00:00:01
dot=2023-01-01 00:00:01
dirfft=x:\DiPo\Data\BORISLAVKA\EXPORT1\
sqlcmd=SELECT ID, souprava, (IF(souprava="108",1,0) + IF(souprava="118",1,0)) as tar,
"19" as nap from passdata WHERE souprava > "100" AND souprava < "120" AND
datetime>"%2:s"
```

Jména sekcí v INI popisují jednotlivé přednastavené výpočty. Tedy např. „train001“ pak lze vybrat z rozevíracího seznamu D a načte se příslušné nastavení. Tlačítka NEW a OPEN vytvoří nebo otevrou zvolené nastavení. Případně lze soubor „setup.ini“ editovat ručně. Lze nastavit rychlost učení prahovou hodnotu pro ukončení, výstupní soubor. Parametry CH, souprava, nap, odt, dot jsou nepovinné a dají se použít jako proměnné v SQL dotazu, kdy:

```
%0:s = CH
%1:s = nap
%2:s = odt
%3:s = dot
%4:s = souprava
```

Tlačítkem „read TRAIN data“ se načtou data do trénovací sady.

Trénování NS

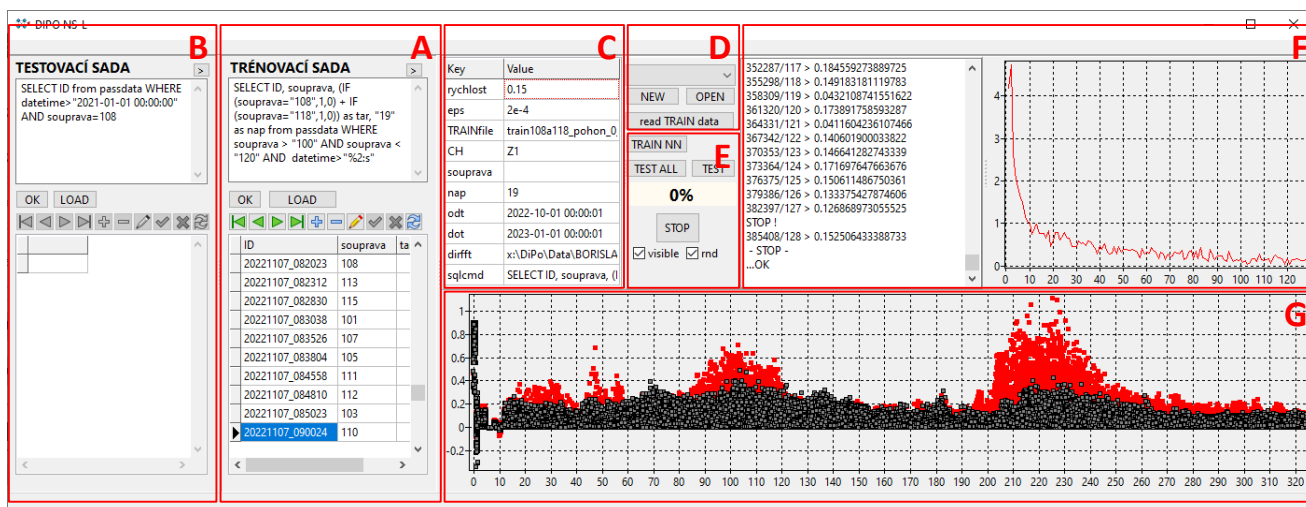
Trénování se spustí tlačítkem „TRAIN NN“ a průběh trénování lze vidět v logu a také v grafu v sekci F, kde se vykresluje chybová funkce v závislosti na počtu epoch učení. Jedná se o učení systémem zpětné propagace. Výsledek učení se průběžně ukládá do souboru *.dat a lze jej použít pro pozdější vyhodnocení. Další funkci však nemá. Trénování je dostatečné, když hodnota chybové funkce klesne pod nastavenou prahovou hodnotu „eps“.

Průběh trénování může být různý – očekává se klesající funkce. Někdy se mohou objevit lokální minima a maxima, což nevadí. Pokud hodnota chybové funkce ke konci učení roste, je NS přeučená. Pak je potřeba změnit rychlost učení a omezit počet epoch učení. Pokud ani to nepomůže je nutné zvážit jinou trénovací množinu nebo jinou topologii sítě.

Testování NS

Výsledkem trénování jsou soubory *.TRD (váhy) a *.NNW (topologie). Tyto je potřeba uchovat, aby mohla být prováděna detekce pomocí modulu SOL_NSD. Test lze ale provést také přímo v SOL_NST. V sekci B se nastaví SQL dotaz pro výběr testovací sady. Lze tak otestovat vektor mnoha průjezdů nebo pouze jeden vybraný průjezd. Filtry CH, souprava, nap, odt a dot lze použít jako v sekci A. Tlačítko TEST provede test aktuálně vybraného průjezdu v sekci B, tlačítko TEST ALL provede test všech průjezdů načtených v seznamu sekce B. Výsledky se zobrazují v textové části sekce F a také do logu, kde je lze kontrolovat. Přímý výstup do databáze zde možný není, protože se jedná pouze o testování. Samotnou detekci pak provádí program SOL_NSD.

Výsledkem detekce je číslo od 0 do 1, jedná se o pravděpodobnost přítomnosti naučeného příznaku. Výsledky se zobrazují v sekci G, kde na X ose jsou čísla souprav a na Y ose jsou hodnoty detekce. Stejným způsobem provádí detekci program SOL_NSD s tím, že se výsledky ukládá do databáze.



Obr. 4 Popis GUI programu SOL_NST.

Nepředpokládá se, že byl tento modul běžnému uživateli dostupný, protože u trénování NS se jedná o odbornou činnost nastavení systému. K trénování je potřeba vhodně vytvořit trénovací množinu z reprezentativních dat a tvorba nové NS pro detekci je činnost výzkumná, nikoli uživatelská. **Proto při komerčním využití SW je tento modul dodáván samostatně, případně bude tato činnost nabídnuta jako služba.**

4.5 SOL_NSD

SOL_NSD je modul pro detekci poruch pomocí natrénované NS. Vstupem je topologie sítě a naučené váhy, výstupem je hodnota pravděpodobnosti detekce příznaku poruchy.

Vstup:	1) Soubor vah (natrénovaná data) *.TRD 2) Soubor topologie sítě *.NNW
Výstup:	1) Záznam v DB v tabulce X_trains_health

Tento modul nemá grafické prostředí a spouští se stejným způsobem jako modul SOL_STAT. Parametry spuštění jsou následující:

```
SOLNSD.EXE AUTO TRD NNW ID CH CNT
```

```
AUTO - automatické ukončení: 0 = ne / 1 = ano,  
TRD - soubor natrénovaných vah,  
NNW - soubor topologie sítě  
ID - ID průjezdu pro vyhodnocení, při „*“ zvolí se následující nevyhodnocený,  
CH - volba kanálu k vyhodnocení, při „*“ všechny zapnuté v INI konfiguraci  
CNT - počet výpočtů, po kterém se SOLVER ukončí, při „0“ vypočte vše dostupné
```

Ve výstupní tabulce je záznam uložen pro dané ID průjezdu, soupravu, nápravu a kanál. Záznamem je číslo od 0 do 1, kde se jedná o pravděpodobnost detekce příznaku poruchy od 0 = bez příznaku, 1 = s příznakem. Nastavení prahových hodnot pro alarm už je na straně FRON-END řešení, kde lze hodnotu zobrazit v barevné škále.

4.6 WEB-CONN

Modul WEB-CONN je realizován PHP skriptem na straně serveru. Tato serverová aplikace umožňuje přenos dat mezi serverem a okolním internetem. Komunikace probíhá vždy zabezpečenou formou skrze https spojení a komunikace může probíhat mezi jinou webovou aplikací, konzolovou aplikací nebo uživatelským programem.

Hlavním úkolem tohoto modulu je data přijímat, validovat, třídít, archivovat a předávat jiným aplikacím. Před doručení dat se může (ale nemusí) odesílající strana dotázat, zda bude soubor přijat a až v případě kladné odezvy soubor odešle. To je výhodné při odesílání větších souborů, aby se zamezilo zbytečným přenosům dat v případě, kdy budou data stejně zamítnuta. Při zasílání metadat

malého objemu se validace předem neprovádí. Druhým důležitým úkolem tohoto modulu je třídění dat podle domény diagnostiky, měřené lokality, měřeného kanálu, druhu dat apod. Po vytřídění jsou soubory nahrány do příslušného adresáře na serveru a následně je do DB uložena informace o přijetí souboru (název, typ, zařazení, velikost apod.).

Parametry pro přenos

Komunikace probíhá pomocí HTTPS POST na portu 80. Při každé komunikaci musí dotaz obsahovat správný API KEY, pokud tomu tak není, pak se dotaz ignoruje. V dotazu lze nastavit parametr „cmd = test“ nebo „cmd = upload“. V prvním případě se testuje, zda lze soubor odeslat (resp. zda ho server přijme), v druhém případě již dochází k odesílání souboru.

Přehled všech parametrů:

key	API key pro povolení přenosu
cmd	příkaz požadované operace
dom	doména diagnostického místa
file	název souboru
size	velikost souboru

Server odešle odpověď v textové podobě:

```
[upload]
name=DIPO_connector-2
ver=0.2
cmd=upload

[testfile]
dom=opos
domexists=1
confexists=1

[meta]
hwid=OPOS001
id=20240202-072119
ch=ch1
daq=DAQ1
ext=res
date=20240202
time=072119
upload1_enable:1
fileexists=0

[upload]
upload_test=0--upload--OK

[save]
dir=./opos_data/down/2024/20240202/20240202-072119/
dbg_dir=./opos_data/down/2024/20240202/20240202-072119/

[upload]
name=DIPO_connector-2
ver=0.2
cmd=upload
```

```
[testfile]
dom=opos
domexists=1
confexists=1

[meta]
hwid=OPOS001
id=20240202-072119
ch=ch1
daq=DAQ1
ext=res
date=20240202
time=072119
upload1_enable:1
fileexists=0

[upload]
upload_test=0--upload--OK
```

Pokud odpověď končí znaky „--OK“, pak byl soubor v pořádku nahrán. Pokud došlo k chybě, pak tato značka na konci textu není a v textu se lze dočíst o tom, jaká chyba vznikla.

Testuje se:

- existence požadované domény na serveru,
- existence adresáře pro upload na serveru,
- velikost a souboru (zda se nahrál celý),
- existence souboru na serveru – aby nevznikly duplicity,
- správná přípona souboru,
- volné místo na disku serveru.

Speciální funkci mají soubory *.JSON. Pokud je nahrán tento soubor, pak je serverem analyzován a dále zpracován. Např. z *.JSON souborů průjezdů se data uloží do databáze průjezdů (ID, rychlost, počet náprav, apod..). Dále lze pomocí JSON souborů měnit nastavení serveru nebo předávat jiná metadata. Jedná se o otevřenou funkcionalitu připravenou pro další rozvoj. V aktuální verzi se extrahují pouze soubory průjezdů. V případě přenosu dat (cmd = upload) jsou k dotazu POST přidána také binární data. U hlavičkových souborů je MIME TYPE = application/json, u souborů binárních (měření) je MIME TYPE = application/octet-stream.

Nastavení serveru

V tabulce databáze X_config jsou uloženy parametry, které ovlivňují chování serveru:

upload_only_json	Posílat pouze hlavičkové soubory JSON	0 = NE / 1 = ANO
upload_extes	Povolené přípony souborů	r32,json,r64,ini
upload_enable	Povolit upload?	0 = NE / 1 = ANO
upload_interval	Interval v jakém HW kontroluje server	číslo v sekundách
upload_delmode	Potvrzené soubory: delete = mazat / move = přesunout	

na základě výše uvedených hodnot server rozhodne, zda bude soubor nahrán.

Soubory lze odesílat z aplikace SENDER nebo realizovat jinou aplikací. Možné je také vytvořit WEB formulář pro ruční nahrávání souboru.

Literatura

- [1] VÁGNER, J. et al. *Wayside diagnostika pojezdu kolejových vozidel. Průběžná zpráva z řešení projektu*. Č. zprávy DIPO-2022-JV-01. Pardubice: Univerzita Pardubice, 2023.
- [2] VÁGNER, J. et al. *Wayside diagnostika pojezdu kolejových vozidel. Průběžná zpráva z řešení projektu*. Č. zprávy DIPO-2021-JV-01. Pardubice: Univerzita Pardubice, 2022.
- [3] *Apache HTTP Server Project* [online]. The Apache Software Foundation, 2023 [cit. 2023-01-10]. Dostupné z: <https://httpd.apache.org/>
- [4] *MariaDB Foundation* [online]. [cit. 2023-01-05]. Dostupné z: <https://mariadb.org/about/>
- [5] *Grafana Labs* [online]. [cit. 2023-01-24]. Dostupné z: <https://grafana.com/oss/grafana/>
- [6] *Lazarus: The professional Free Pascal RAD IDE* [online]. [cit. 2023-01-24]. Dostupné z: <https://www.lazarus-ide.org/>