

## *Tehnična implementacija zalednih procesov za avtomatsko upravljanje s prožnostjo v Elektro Celju*

Leon Maruša<sup>1</sup>, David Kramer<sup>1</sup>, Anton Kos<sup>1</sup>, Nermin Suljanović<sup>2</sup>, Boštjan Čegovnik<sup>2</sup>, Igor Podbelsšek<sup>2</sup>

<sup>1</sup> – Elektro Celje, Vrunčeva ul. 2a, 3000 Celje

[leon.marus@elektro-celje.si](mailto:leon.marus@elektro-celje.si), [anton.kos@elektro-celje.si](mailto:anton.kos@elektro-celje.si), [david.kramer@elektro-celje.si](mailto:david.kramer@elektro-celje.si)

<sup>2</sup> – EIMV, d.o.o., Hajdrihova 2, 1000 Ljubljana

[nermin.suljanovic@eimv.si](mailto:nermin.suljanovic@eimv.si), [boštjan.cegovnik@eimv.si](mailto:boštjan.cegovnik@eimv.si), [igor.podbelsek@eimv.si](mailto:igor.podbelsek@eimv.si)

**Povzetek** – Upravljanje s prožnostjo predstavlja novo možnost zagotavljanja stabilnega obratovanja distribucijskega omrežja v trenutkih, ko se na omrežju pojavijo raznovrstne motnje. V primerih koriščenja prožnosti se trenutno v distribucijskih omrežjih najpogosteje pojavlja potreba po sledečih sistemskih storitvah: razbremenjevanje SN/NN transformatorjev in napetostna regulacija vzdolž NN vodov. Platforma za upravljanje s prožnostjo ponuja elektrodistribucijskim podjetjem uporabo vseh poslovnih in tehničnih procesov, ki so vezani na upravljanje s prožnostjo: objava razpisov, oddaja ponudb, izbor ponudb, predkvalifikacija virov, izračun napovedi in baseline-a, napoved aktivacij, izvedba aktivacij, obračun aktivacij, itd.

Ker omenjene sistemske storitve po večini potekajo v NN omrežju, se pričakuje, da bo veliko število aktivacij v omrežjih različnih transformatorskih postaj težko upravljati ročno. V ta namen smo v Elektro Celju v okviru evropskega projekta EV4EU implementirali platformo za upravljanje s prožnostjo imenovano FlexIS, ki ima možnost izvajanja avtomatskih poslovnih in tehničnih procesov za proženje aktivacij v distribucijskih omrežjih glede na napovedane obratovalne razmere. V članku je predstavljena ciljna arhitektura povezljivosti in funkcionalnosti vseh informacijskih sistemov v verigi upravljanja s prožnostjo, podrobneje pa je opisana implementacija zalednih procesov za avtomatsko upravljanje s prožnostjo, ki so bili implementirani v sklopu orodij na platformi za upravljanje velepodatkov LAMBDA.

**Ključne besede:** prožnost, sistemske integracije, velepodatki, platforma za upravljanje s prožnostjo.

## *Technical implementation of backend processes for automatic flexibility management at Elektro Celje*

**Summary** – Flexibility management represents a new opportunity for ensuring stable operation of the distribution network during times when various disturbances occur. In cases where flexibility is utilized, the most common system services required in distribution networks currently include: relieving MV/LV transformers and voltage regulation along LV lines. A flexibility management platform offers distribution system operators access to all business and technical processes related to flexibility management: tender publication, bid submission, bid selection, prequalification of resources, forecast and baseline calculation, activation forecasting, activation execution, activation settlement, and more.

Since these system services are predominantly carried out in LV networks, it is expected that a large number of activations across networks of various transformer stations will be difficult to manage manually. For this purpose, Elektro Celje has implemented a flexibility management platform called **FlexIS** as part of the European project **EV4EU**. This platform enables the execution of automated business and technical processes for triggering activations in distribution networks based on forecasted operational conditions. The article presents the target architecture for connectivity and functionality of all information systems involved in the flexibility management chain, with a detailed description of the backend process implementation for automatic flexibility management, developed as part of the tools on the **LAMBDA** big data management platform.

**Keywords:** flexibility, system integration, big data, flexibility management platform.

## 1 UVOD

Prožnost kot sistemska storitev za elektrodistribucijska podjetja (EDP) se fizično izvaja v distribucijskem omrežju s strani aktivnih uporabnikov omrežja. Pri tem so storitve prožnosti pogosto lokacijsko omejene, saj želimo z njimi stabilizirati specifičen del omrežja v katerem se pojavi obratovalna nestabilnost. Takšen primer lokacijsko odvisnih sistemskih storitev sta razbremenjevanje ali regulacija napetosti v NN omrežju. Ti dve sistemski storitvi imata potencial, da pripomoreta h reševanju trenutno najpogostejših problemov obratovanja NN omrežij. Porast vključevanja sončnih elektrarn v NN omrežja je v zadnjih letih imel velik vpliv na pojav previsokih napetosti, še posebej v poletnem času, ko je proizvodnja sončnih elektrarn velika hkrati pa v teh omrežjih ni lokalnega odjema energije. Posledično energija mnogokrat odteka v SN omrežje, včasih tudi v takšnih količinah, da njena trenutna moč presega nazivno moč SN/NN transformatorjev, zaradi česar se pojavijo preobremenitve. Po drugi strani smo v zadnjih letih priča porastu prehoda na ogrevanje s sistemi toplotnih črpalk, ki predstavljajo največji porabnik v večernih zimskih dnevih. Toplotne črpalke tako največji del električne energije porabijo ravno v urah, ko sončne elektrarne v istih omrežjih proizvedejo manj energije, v tem kontekstu se ravno tako pojavljajo preobremenitve SN/NN transformatorjev, kjer energija teče iz SN v NN omrežje.

Zaradi teh razlogov se slovenski EDP na področju prožnosti trenutno najbolj osredotočajo v izrabo sistemskih storitev razbremenjevanja in napetostne regulacije NN omrežij. Kljub temu, da je dobava teh sistemskih storitev fizična je za njihovo učinkovito izvajanje nujno potrebno zagotoviti ustrezno informacijsko podporo, ki zajema poslovne in tehnične procese ter izmenjavo in obdelavo relevantnih podatkov. Prav tako je potrebno sisteme za upravljanje s prožnostjo načrtovati na način, da bo v bližnji prihodnosti z njimi mogoče upravljati tudi nove sistemske storitve, ki še niso popolnoma definirane, a bodo pomembne za EDP (npr. otočno obratovanje NN omrežja), koriščene s strani agregatorjev (npr. optimizacija energetskega portfelja) ali prenosnega operaterja (regulacija frekvence z agregiranimi viri). To pomeni, da mora biti informacijski sistem dovolj fleksibilen za nadgradnje in upoštevati hramonizacijske standarde za enovito definicijo storitev ter izmenjavo podatkov.

V nadaljevanju tega referata predstavljamo postopek uspešne uvedbe informacijskega sistema (platforme) za upravljanje prožnosti imenovanega FlexIS, integracijo sistema z izvori in ponori podatkov, podrobno tehnično implementacijo zalednih procesov za avtomatsko upravljanje s prožnostjo ter način in vsebino izmenjave podatkov, ki so bili potrebni za uspešno izvedbo takšnega projekta. Sistem, ki bo predstavljen je bil implementiran v okviru evropskega projekta EV4EU.

## 2 FUNKCIONALNE ZAHTEVE IN KONCEPTUALNA ZASNOVA

### 2.1. Platforma za upravljanje prožnosti in Enotna Vstopna Točka

V grobem je platforma FlexIS namenjena upravljalcem distribucijskega omrežja in je vmesni člen med funkcionalnostmi za oceno stanja v omrežju, ki kot rezultat podajo lokacijsko in količinsko potrebo po prožnosti ter trgom prožnosti, ki to prožnost ponujajo kot rešitev (agregatorji ali posameznimi viri prožnosti). Sistem je novost v slovenskih EDP, tako iz stališča uvedbe novih poslovnih procesov, kot tudi iz stališča uvedbe novih tehničnih funkcionalnosti, predvsem tistih, ki se navezujejo na točno oceno trenutnega in prihodnjega obratovalnega stanja NN omrežij in integracijo z že obstoječimi sistemi za nudenje sistemskih storitev, ki so na voljo pri nekaterih slovenskih agregatorjih.

Pri definiciji funkcionalnih zahtev in kasnejšem pregledu trga za že razvitimi standardiziranimi informacijskimi rešitvami za FlexIS smo ugotovili, da trenutno na trgu ni na voljo sistema, ki bi pokril vse nujne zahteve in bi hkrati bil ekonomsko učinkovit, zaradi česar smo se v Elektro Celju odločili za specifičen razvoj sistema FlexIS v sodelovanju z Elektroištitutom Milan Vidmar (EIMV). Pri načrtovanju sistema in pripravi funkcionalnih specifikacij smo v največji možni meri izhajali iz Študije z navodili o storitvah prožnosti [1], katero je leta 2022 izdelal EIMV. Študija obravnava celoten koncept vzpostavitve trga s prožnostjo v Sloveniji in definira poslovne procese za upravljanje s prožnostjo ter harmonizirane produkte prožnosti, slednji so podprti s podatkovnimi modeli, katerih podatki se med informacijskimi sistemi izmenjujejo po standardiziranih formatih za izmenjavo podatkov, slednji so določeni s CIM standardi družine IEC 62325 [2]. Študija prav tako upošteva izsledke preteklih evropskih projektov, kot so SmartNet, CoordiNet, INTERFACE, EUniversal, EU-SysFlex, InteGrid, Flex4Grid in OneNet ki so med leti 2015 in 2023 na ravni EU definirali dobre prakse za upravljanje s prožnostjo.

Pri definiciji funkcionalnih zahtev smo upoštevali, da bo FlexIS umeščena v informacijsko okolje, kjer so določene funkcionalnosti za poslovne procese prožnosti že na voljo na Enotni Vstopni Točki Nacionalnega Podatkovnega Vozlišča (EVT-NPV), slednjega za slovenske EDP-je razvija podjetje Informatika d.o.o. Funkcionalnosti, ki so bile razvite v okviru EU projekta OneNet obsegajo: objavo razpisa za lokacijsko potrebo po prožnosti, pridobivanje ponudb za lokacijsko nudenje prožnosti, sklepanje pogodb o dobavi prožnosti, posredovanje aktivacij storitev prožnosti in registracijo ponudnikov prožnosti.

Funkcionalnosti implementirane na EVT so na voljo preko komunikacijskih vmesnikov namenjenih za M2M (machine-to-machine) komunikacijo. Ti vmesniki bodo nujno potrebni v prihodnosti, ko bo ekosistem prožnosti tehnološko zrel do tega nivoja, da bodo opravila za izvajanje prožnosti potekala popolnoma avtomatsko. Ker so poslovni procesi za prožnost trenutno še v začetnih fazah vsakodnevnega poslovanja EDP-jev, omenjeni vmesniki na EVT pa nimajo podpore za grafični (GUI) pregled nad podatki oz. izvajanjem procesov, smo morali izmenjane podatke preko vmesnikov grafično prikazati na FlexIS.

Druge zahtevane funkcionalnosti, ki so del upravljanja s prožnostjo EVT nima razvitih. Te funkcionalnosti so bile razvite in umeščene v FlexIS platformo. Funkcionalnosti FlexIS platforme in produkte prožnosti, ki jih platforma trenutno podpira so podrobneje opisane v [3].

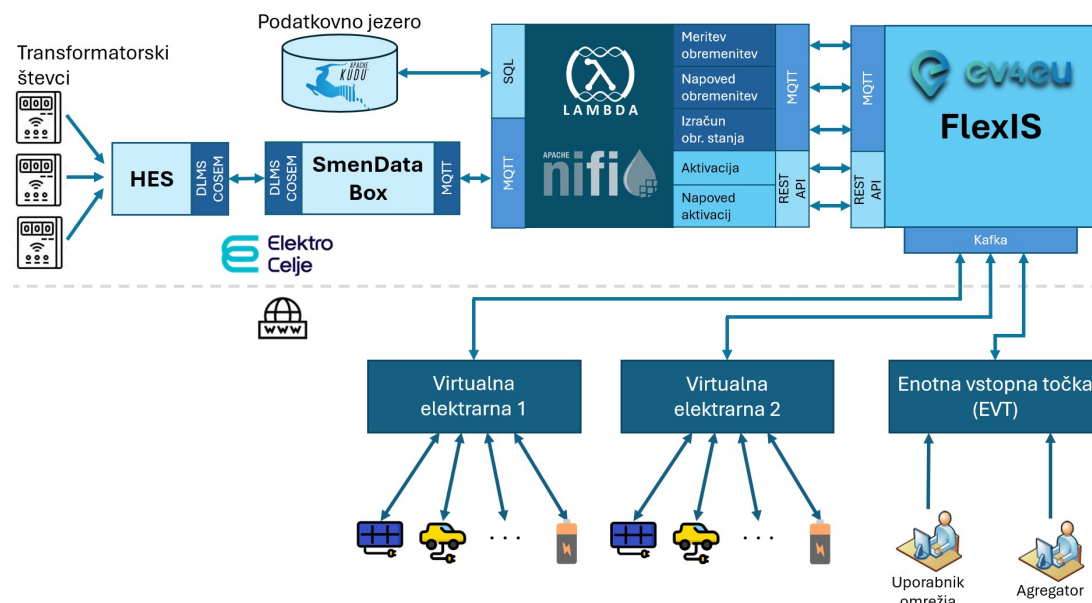
## 2.2. Ostali ključni informacijski sistemi in celotna arhitektura

S stališča izmenjave podatkov in izvajanja funkcionalnosti FlexIS in EVT predstavljata pretežno poslovni del funkcionalnosti za upravljanje s prožnostjo. Za uspešno izvajanje prožnosti je potrebno vzpostaviti tudi tehnične funkcionalnosti, kot so: pridobivanje meritev obremenjenosti NN omrežij v realnem času, izdelavo napovedi preobremenjenosti NN omrežij, izdelavo napovedi aktivacij v NN omrežjih, izračun obratovalnega stanja NN omrežij, izvedba proženja aktivacij glede na njihove napovedi, pridobivanje meritev aktivacij v NN omrežjih.

Poleg FlexIS in EVT so sistemi za delovanje prožnosti tudi:

- virtualne elektrarne (VPP), ki predstavljajo platforme za izmenjavo tehničnih podatkov agregiranih virov prožnosti posameznega agregatorja z FlexIS in posredovanje aktivacij do posameznih virov prožnosti.
- Sistem LAMBDA (Laboratorij Analitskih Multifunkcionalnih Big Data Aplikacij), ki predstavlja velepodatkovno platformo Elektra Celje in implementira zaledne tehnične funkcionalnosti opisane v prejšnjem odstavku.

Končna ciljna arhitektura medsebojno povezuje sisteme: FlexIS, EVT, LAMBDA in VPP različnih agregatorjev. Med temi sistemi se morajo preko integracij izmenjevati standardizirana sporočila na relacijah, ki so prikazane na sliki 1. Podrobnejša implementacija integracij je opisana v poglavju 3.



Slika 1: Podrobna končna ciljna arhitektura sistemov v verigi upravljanja s prožnostjo.

### 3 IMPLEMENTACIJA TEHNIČNIH ZALEDNIH PROCESOV

Kot že omenjeno je nekaj poslovnih funkcionalnosti za upravljanje s prožnostjo že implementiranih v EVT, nove poslovne funkcionalnosti, ki smo jih potrebovali smo implementirali v FlexIS in so podrobneje opisane v [3]. Največji izziv za Elektro Celje pri vzpostavitvi celotnega upravljanja s prožnostjo je bila vzpostavitev zalednih tehničnih procesov, ki se nanašajo na realno časovne meritve in algoritme za napovedovanje preobremenjenosti omrežja. Implementacijo teh smo v celoti izvedli v velepodatkovnem okolju LAMBDA. Za takšen pristop smo se odločili ker okolje LAMBDA že vsebuje celoten nabor D-1 števnih meritev, ki jih vsebuje tudi števeno-merilni center Elektra Celje. LAMBDA prav tako iz HES (ang. Head-End System) posredno pridobiva realno časovne meritve transformatorskih števcov v TP. V okolju LAMBDA že imamo na voljo orodja za implementacijo podatkovnih integracij (konkretno Apache NiFi) z uporabo standardiziranih protokolov (npr. MQTT, AMQP, REST, Kafka stream, itd.). Apache NiFi omogoča enostavno izvajanje poljubnih sprememb na integracijah in samostojni razvoj zahtevanih integracij in algoritmov. Glavna prednost pri implementaciji zalednih procesov pa so bile izkušnje kadrov Službe za napredno analitiko podatkov v Elektro Celju. Namreč s trenutnimi tehničnimi znanji vzpostavitve integracij in algoritmov smo bili v celoti sposobni izvesti funkcionalne zahteve kar je projektu omogočalo veliko mero prilagodljivosti in izjemno hiter razvoj.

V praksi smo se pri demonstraciji prožnosti v okviru EV4EU trenutno osredotočali le na storitve za razbremenjevanje NN omrežij. V prihodnosti pa bo predmet testiranj na FlexIS platformi tudi napetostna regulacija. Na transformatorskih števcih pilotnih območij (omrežji TP INKUBATOR VRBINA in TP GIMNAZIJA VELENJE) smo nastavili 15 minutni PUSH meritve. Te meritve se iz števcov pošljejo v HES (ang. Head End System) sistem po DLMS/COSEM protokolu. Meritve iz HES zajame tudi programska komponenta SmenDataBox, ki podatke iz DLMS/COSEM pretvarja v XML format skladen s CIM standardom IEC 61968-9 (Interface Standard for Meter Reading & Control) [4]. Eno takšno XML sporočilo predstavlja eno 10 ali 15 minutno meritev za posamezni števec. Vsako XML sporočilo se v SmenDataBox-u zapiše v MQTT ovojnico in se po MQTT protokolu pošlje na LAMBDO v komponento Apache NiFi [5], kjer se števnici podatki iz XML formata preberejo in se uporabijo nadalje za ostale namene. V ta namen uporabljamo že obstoječe komponente v NiFi orodju. Zajem MQTT podatkov se izvede s komponento *ConsumeMQTT*, ki predstavlja MQTT odjemalca na določeni temi. Zajeta MQTT sporočila označimo s časovno značko prihoda v NiFi in nato z ostalimi NiFi komponentami iz XML formata izluščimo (parsamo) podatke, ki jih potrebujemo, ter jih zapišemo v enostaven JSON format. Podatke zapišemo v JSON, ker je ta format bolj učinkovit za zapis (velikost sporočil se je za realno časovne meritve zmanjšala za faktor 15,3) kar pomeni manj porabe strežniških virov za procesiranje, prav tako pa se JSON običajno uporablja za integracijske storitve, ki temeljijo na REST arhitekturi, ki je ena izmed najbolj uporabljenih in standardiziranih rešitev za spletne integracije.

V nadaljevanju podrobneje opisujemo implementacijo tehničnih zalednih procesov in izvedenih podatkovnih integracij z orodjem NiFi.

#### 3.1. Pridobivanje in posredovanje meritev obremenjenosti NN omrežij v realnem času

Pri določanju preobremenjenosti NN omrežij je potrebno določiti najbolj obremenjeno točko v omrežju. Pri tem se osredotočamo predvsem na preobremenitev SN/NN transformatorja, NN izvodov (izvodnih varovalk) ali posameznih odsekov na NN vodih. Težavo predstavlja dejstvo, da vse točke NN omrežja niso observabilne (pomanjkanje meritev) ali pa je njihova observabilnost preveč zakasnjena (latenca dostave meritev v merilni center). V večini primerov so z meritvami delovnih moči opremljena le merilna mesta in NN transformatorsko polje, pri čemer imajo merilna mesta latenco števnih meritev od nekaj ur do tudi več dni, zato uporaba meritev iz merilnih mest ni praktična za znotraj-dnevno določitev preobremenitev posameznih elementov NN omrežja. Po drugi strani se meritve v NN transformatorskih poljih izvajajo s pametnimi (transformatorskimi) števci novejših generacij, ki imajo možnost t. i. PUSH meritev, slednje pomeni, da lahko števene meritve pridobivamo v realnem času, običajno vsakih 10 ali 15 minut. Zaradi tega razloga je točka observabilnosti za spremljanje obremenjenosti NN omrežja najpogosteje meritev moči SN/NN transformatorja. Transformatorski števec v realnem času pošilja tudi meritve faznih napetosti in tokov in sicer vsakih 10 minut. Meritev napetosti je osnova za izvedbo napetostne regulacije, a le na nivoju celotnega NN omrežja, npr. v primeru, da na NN zbiralkah zaznamo prenizke ali previsoke napetosti.

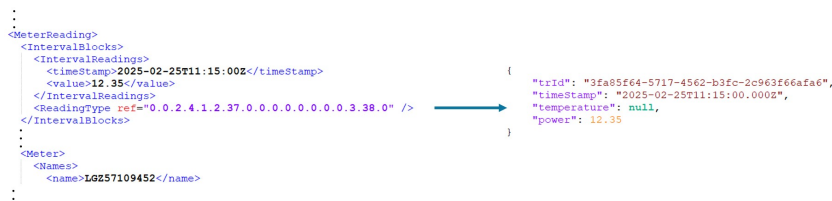
Ena izmed zahtev za implementacijo je bila, da FlexIS platforma pridobi realno časovne meritve transformatorskih števecov po MQTT protokolu v JSON formatu. V praksi je to pomenilo, da iz XML sporočil izluščimo sledeče podatke:

- tovarniško številko števca – v XML sporočilu se podatek nahaja v tagu `Message.Payload.MeterReadings.MeterReading.Meter.Names.name`
- časovno značko meritve – v XML sporočilu se podatek nahaja v tagu `Message.Payload.MeterReadings.MeterReading.IntervalBlocks.IntervalReadings.timeStamp`
- register števca meritve – v XML sporočilu se podatek nahaja v tagu `Message.Payload.MeterReadings.MeterReading.IntervalBlocks.ReadingType`
- vrednost meritve – v XML sporočilu se podatek nahaja v tagu `Message.Payload.MeterReadings.MeterReading.IntervalBlocks.IntervalReadings.value`

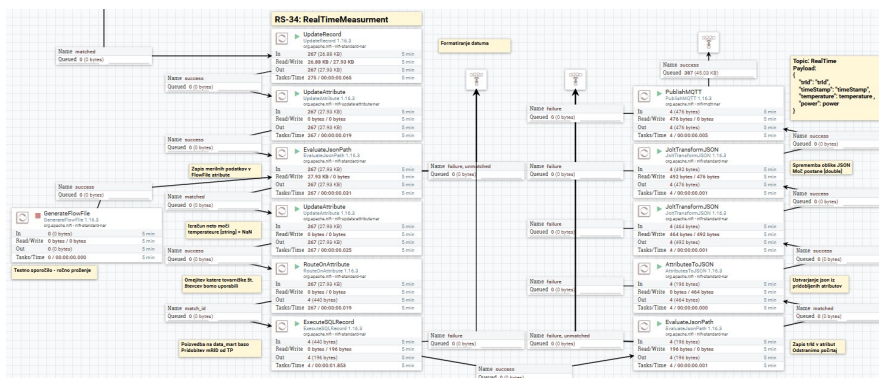
Pri tem je potrebno upoštevati, da ima vsak specifičen register števca svojo časovno značko in svojo vrednost meritve. Posamezni registri so v tagu `ReadingType` označeni z `ReadingType` CIM kodami, ki so definirane v standardu IEC 61968-9 – Aneks C [4]. Iz števecov pridobivamo meritve za 4 različne registre števca, od tega za prožnost uporabljamo meritve le 2 registra, ki predstavljata:

- delovno energijo prejeto iz omrežja (A+) v kW – `ReadingType = 0.0.2.4.1.2.37.0.0.0.0.0.0.0.0.0.3.38.0`
- delovno energijo oddano v omrežje (A-) v kW – `ReadingType = 0.0.2.4.19.2.37.0.0.0.0.0.0.0.0.0.3.38.0`

Pridobljeni meritvi netiramo tako, da v NiFi komponenti od prejete delovne energije odštejemo oddano delovno energijo in dobimo eno netirano meritev, ki je pozitivna, ko energija teče skozi transformator v smeri SN → NN in negativna, ko energija teče v smeri NN → SN. Meritev pretvorimo iz energije v moč in formatiramo v JSON format z NiFi komponentami, vendar je meritev še zmeraj zapisana za posamezni števec, saj je identifikator izvora meritve še zmeraj tovarniška številka števca, kar ne identificira SN/NN transformatorja kateremu ta števec pripada. Ker v LAMBDO shranjujemo tudi podatke o zgodovini merilnih mest (obračunskih in obratovalnih), prav tako pa shranjujemo zgodovino sprememb na omrežju (snapshoti GIS podatkov) smo številko števca uparili z naborem obratovalnih merilnih mest in dobili tisto merilno mesto kateremu pripada dotični transformatorski števec. Pridobljeno merilno mesto smo nadalje uparili s seznamom TP-jev iz podatkov o omrežju, s čimer pridobimo mRID identifikator TP-ja v katerem je to obratovalno mesto nameščeno. Celoten proces je izveden z SQL proceduro, ki tako iz tovarniške številke števca pridobiva mRID identifikator TP-ja v katerem se števec nahaja. SQL procedura se proži v NiFi-ju in poizveduje po Kudu podatkovnem jezeru v LAMBDI. mRID identifikator TP-ja se zapiše v JSON sporočilo, ki sedaj predstavlja meritev moči za posamezni TP. Pretvorbo iz originalne XML oblike v JSON obliko kaže slika 2. Pretvorjena JSON sporočila (imenovana *RealTimeMeasurement.json*) nato z NiFi komponento *PublishMQTT* zapišemo v MQTT ovojnico in jih pošljemo na FlexIS platformo. Celoten tok procesiranja podatkov z vsemi pripadajočimi NiFi komponentami za obdelavo realno časovnih meritev transformatorskih števecov kaže slika 3.



Slika 2: Meritev transformatorskega števca pretvorjena v *RealTimeMeasurement* JSON sporočilo za FlexIS.



Slika 3: NiFi komponente za procesiranje realno časovnih meritev transformatorskih števecov.

### 3.2. Izdelava napovedi obremenjenosti NN omrežij

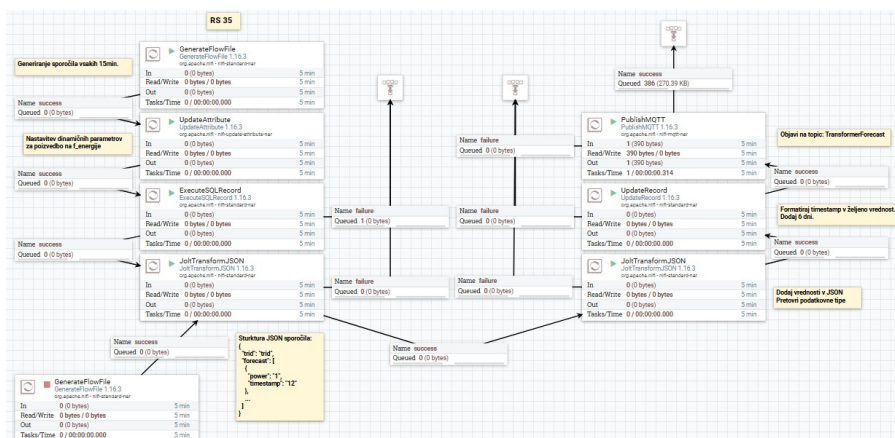
Da pridobimo informacijo o tem kdaj v prihodnosti bo SN/NN transformator preobremenjen moramo izdelati napoved obremenitve transformatorja. V teoriji to sicer ne bi bilo potrebno, saj lahko dejansko preobremenjenost pridobimo iz realno časovnih meritev in aktivacijo prožimo za naslednjo 15 minutno časovno periodo. V tem primeru, bi transformator bil preobremenjen le 15 minut in še dodatni čas, ki je potreben za dovolj veliko aktivacijo virov prožnosti. V praksi aktivacije virov prožnosti zahtevajo pripravo in določene procedure, kar lahko podaljša čas pričetka aktivacije in s tem tudi čas preobremenitve transformatorja, zato je ustrežnejše, da aktivacije vnaprej napovemo virom prožnosti (oz. agregatorjem) kar je podrobneje opisano v poglavju 3.3.

V konkretnem primeru smo za izdelavo napovedi uporabili zelo enostaven algoritem soležnih dni. Za konkretni transformatorski števec v LAMBDA bazi števnih meritev z SQL proceduro v NiFi komponenti *ExecuteSQLRecord* pridobimo meritve za časovne značke večje od trenutnega časa v dnevu D-6. Te meritve predstavljajo napovedi za enake časovne značke v dnevu D+1, saj je D-6 soležni dan od D+1 (razlika je točno 1 teden, kar pomeni, da gre za isti dan v tednu). Poenostavljeno zapisano, če je trenutni čas 11:15 je napoved za jutri ob 11:15 predstavljena kot dejanska meritev 6 dni v preteklosti ob 11:15. Napoved se sicer izdelava vsakih 15 minut (proženje z NiFi komponento *GenerateFlowFile*) in pošlje na FlexIS platformo za časovni horizont T+24h. Potrebno je poudariti, da algoritem soležnih dni ni dejanski napovedni model v smislu modela strojnega učenja, vendar pa predstavlja osnovni »baseline« model napram kateremu se bodo primerjali novi modeli izdelani v prihodnosti. Poleg tega smo se v sklopu EV4EU projekta osredotočili na integracije podatkov za delovanje procesov prožnosti. Načrtujemo, da se bo napovedni model v bližnji prihodnosti v NiFi-ju zamenjal z bolj točnim, ki bo najverjetneje izdelan v Python jeziku s knjižnicami strojnega učenja kot je *scikit-learn*. Python skripte napovednih modelov je iz NiFi-ja možno pognati z uporabo komponente *ExecuteStreamCommand*.

Pridobljene napovedi prav tako označimo z mRID TP postaje za katero napoved izdelujemo. Takšne podatke nadalje pretvorimo v JSON format, katerega zna brati FlexIS platforma. Pretvorba v JSON format poteka z uporabo *JoltTransform* komponente, ki glede na zapisana JOLT pravila formatira JSON v zahtevano obliko. Sporočila imenovana *TrForecast.json* z napovedanimi vrednostmi z uporabo NiFi komponente *PublishMQTT* zapišemo v MQTT ovojnico in jih pošljemo na FlexIS platformo po MQTT protokolu. Slika 4 kaže končno obliko JSON sporočila, slika 5 pa kaže celoten tok procesiranja podatkov in izdelave napovedi z vsemi pripadajočimi NiFi komponentami.

```
{
  "trId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "forecast": [
    {
      "timeStamp": "2025-02-26T11:15:00.000Z",
      "power": 156.45
    },
    ...
  ]
}
```

Slika 4: Primer napovedi obremenitve transformatorja pretvorjena v *TrForecast* JSON sporočilu za FlexIS.

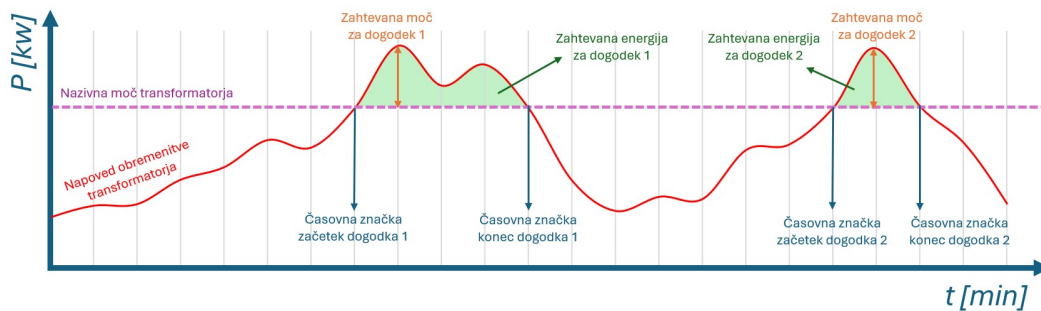


Slika 5: NiFi komponente za izdelavo in pošiljanje napovedi obremenitev transformatorja.

### 3.3. Izdelava napovedi aktivacij v NN omrežjih

Pri prožnosti je ključnega pomena informacija, kdaj bomo prožnost aktivirali. Kot že omenjeno mora ponudnik prožnosti svoje vire pripraviti in jih obvestiti o aktivacijah, zato mora naročnik prožnosti aktivacijo napovedati vnaprej (idealno vsaj nekaj ur vnaprej). Algoritem, ki napoveduje obremenjenost transformatorjev smo uporabili tudi za napoved aktivacij prožnosti. Če poznamo napoved obremenjenosti in nazivno moč transformatorja lahko s primerjavo teh dveh vrednosti sklepamo, kdaj bo transformator obremenjen nad nekim procentom svoje nazivne moči. Za tiste časovne trenutke, ko bo transformator preobremenjen napovemo aktivacijo prožnosti.

Po algoritmu soležnih dni opisanem v poglavju 3.2 izdelamo napoved obremenjenosti transformatorja za celoten naslednji dan in te napovedi primerjamo z nazivno močjo. Pri tem dobimo 15 minutne časovne značke, ko bo transformator obremenjen nad 100 % nazivne moči. Za takšne zaporedne 15 minutne časovne značke določimo začetek in konec zaporedja, ta dva časa predstavljata začetek in konec strnjene obdobja preobremenitve, kar v praksi predstavlja tudi začetek in konec aktivacije, ki jo bomo prožili naslednji dan. Lahko se tudi pojavi situacija, ko bo za naslednji dan napovedanih več preobremenitev znotraj dneva. To pomeni, da bomo imeli več napovedi aktivacij, vsako seveda z različnim začetnim in končnim časom. Ko imamo definirane čase aktivacij izračunamo tudi moč in energijo, ki ju potrebujemo za aktivacijo. Pri dogodku preobremenitve izračunamo razliko med posamezno 15 minutno napovedano preobremenitvijo in nazivno močjo transformatorja, največja izračunana razlika je tudi največja moč aktivacije, ki jo bomo za ta dogodek potrebovali (t. i. kapaciteta moči prožnosti). Pri izračunu potrebne energije izračunamo integral razlike napovedane preobremenitve in nazivne moči transformatorja. Grafični proces izračuna teh količin kaže slika 6.



Slika 6: Proces izračuna časovnih značk začetka in konca napovedanih aktivacij in zahtevane moči ter zahtevane energije.

Izračun količin prikazan na sliki 6 je narejen za primer, ko se preobremenitev zgodi zaradi prevelikega odjema v NN omrežju. Preobremenitev se lahko zgodi tudi v primeru prevelike proizvodnje v NN omrežju, kjer energija teče skozi transformator proti SN omrežju, pri čemer je trenutna izmerjena obremenjenost večja od nazivne moči transformatorja. To je pogosto v NN omrežjih, kjer priključena večja količina RVE (npr. sončnih elektrarn). V tem primeru moramo v NN omrežju aktivirati porabnike električne energije, da se lokalna proizvodnja porabi znotraj NN omrežja ali zmanjšati proizvodnjo lokalnih RVE, s čimer razbremenimo transformator (t. i. negativna aktivacija). Postopek priprave sporočila za napoved negativne aktivacije je s stališča procesa v NiFi-ju popolnoma enak, razlika je le v vrednosti parametra za smer aktivacije.

V praksi 1x-dnevno (npr. ob 20:00) z NiFi komponento *GenerateFlowFile* pošljemo proces izdelave in pošiljanja napovedi aktivacij za naslednji dan. V Kudu podatkovnem jezeru LAMBDE imamo definirano tabelo *d\_napoved\_dogodkov* v katero zapisujemo izdelane napovedi aktivacij. V to tabelo zapišemo sledeče parametre:

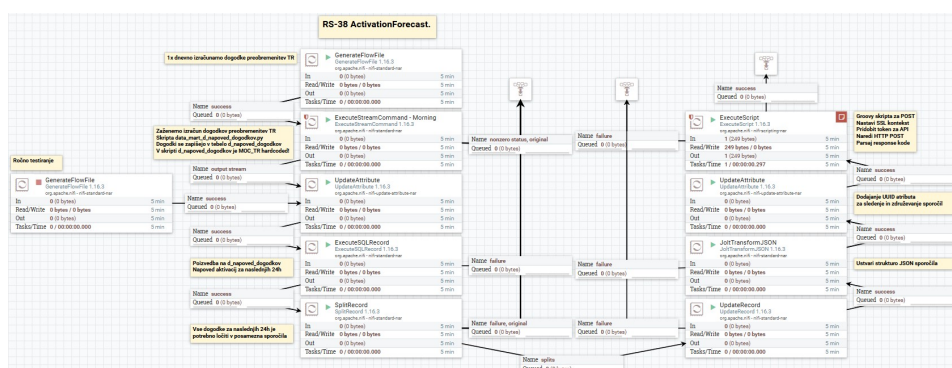
- *datumcas\_zacetek* – časovna značka začetka aktivacije
- *datumcas\_konec* – časovna značka konca aktivacije
- *trajanje\_dogodka* – trajanje aktivacije v minutah
- *zahtevana\_moc* – maksimalna moč, ki jo bomo tekom aktivacije zahtevali
- *rezerva\_moci* – rezerva (procent od zahtevane moči) v primeru izpada dela virov prožnosti
- *dejansko\_zahtevana\_moc* – zahtevana moč seštet z rezervo moči
- *zahtevana\_energija* – integral moči, ki bo aktivirana vsako 15 minutno periodo aktivacije
- *rezerva\_energije* – rezerva (procent od zahtevane energije) v primeru izpada dela virov prožnosti
- *dejansko\_zahtevana\_energija* – zahtevana energija seštet z rezervo energije
- *mrid\_rtp* – mRID šifra RTP, ki je nadrejena TP-ju v katerem bo prožena aktivacija

- *tp\_mrid* – mRID šifra TP-ja v katerem bo prožena aktivacija
- *tr\_mrid* – mRID šifra SN/NN transformatorja v TP-ju za katerega bo prožena aktivacija
- *smer\_aktivacije* – A01 = pozitivna aktivacija, A02 = negativna aktivacija
- *datum\_etl* – časovna značka izdelave napovedi aktivacije

Dogodke, ki so zapisani v tabeli *d\_napoved\_dogodkov* in napovedujejo aktivacije za naslednji dan v NiFi-ju zajamemo s komponento *ExecuteSQL*, nato zapise ločimo enega po enega in za vsakega izmed teh zapisov izdelamo JSON sporočilo imenovano *ActivationForecast*. Na ta način bo eno *ActivationForecast* sporočilo predstavljalo eno napoved (dogodek) neprekinjene aktivacije za naslednji dan. Izdelanim JSON sporočilom dodamo še časovno značko kdaj se je sporočilo ustvarilo in jim dodamo UUID identifikator, da jim lahko sledimo v FlexIS sistemu. *ActivationForecast* sporočila objavimo na REST API storitev FlexIS platforme z uporabo POST http metode. Izdelana JSON sporočila so skladna s specifikacijo integracije z FlexIS platformo, vendar ne vsebujejo vseh parametrov iz tabele *d\_napoved\_dogodkov*, sporočilo *ActivationForecast* je seveda mogoče razširiti z dodatnimi parametri iz *d\_napoved\_dogodkov* v prihodnosti. Slika 7 kaže primer *ActivationForecast* sporočila. Celoten tok procesiranja podatkov z vsemi pripadajočimi NiFi komponentami za izdelavo in pošiljanje napovedi aktivacij kaže slika 8.

```
{
  "activationSignal" : "forecast",
  "areaId" : {
    "rtp" : "83b72764-7db8-4fbf-be8a-58bb643603f7",
    "tp" : "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "mm" : ""
  },
  "activationStart" : "2025-02-26T11:15:00.000Z",
  "activationEnd" : "2025-02-26T13:15:00.000Z",
  "flowDirection": "A01",
  "activatedPower" : 10.58
}
```

Slika 7: Primer napovedi aktivacije transformatorja pretvorjena v končno *ActivationForecast* JSON sporočilo za FlexIS.



Slika 8: NiFi komponente za izdelavo in pošiljanje napovedi aktivacij prožnosti.

### 3.4. Izračun obratovalnega stanja NN omrežij

Aktivacije prožnosti se seveda lahko prožijo izključno le, če je obratovalno stanje NN omrežja stabilno. V praksi to pomeni, da z aktivacijo kateregakoli produkta ne smemo povzročiti dodatnih preobremenitev omrežja ali deviacijo napetosti izven dovoljenih meja, ki morajo biti skladne s standardom SISTE EN 50160. Namreč pričakuje se (oz. ponekod je to že praksa), da bo prožnost v NN omrežju v prihodnje namenjena tudi sistemskim storitvam za prenosnega operaterja, pri čemer imamo v mislih predvsem rRPF storitev oz. t. i. terciarno regulacijo. V primeru, da agregator uspe na območju Slovenije agregirati dovolj manjših prožnih virov lahko pridobi dovolj kapacitete za izvajanje rRPF, saj ta ni lokacijsko odvisna. V primeru aktivacije teh virov ob neugodnem času, pa lahko le-ti povzročijo nestabilno obratovalno stanje v lokalnih distribucijskih NN omrežjih, običajno s prevelikim dvigom ali spustom lokalne napetosti. V ta namen je za dovoljenje aktivacije potrebno agregatorjem zagotavljati signal, kdaj lahko v nekem omrežju vir prožnosti aktivirajo in kdaj morajo aktivacijo v tem omrežju zaustaviti. To problematiko rešuje t. i. semafor oz. TLS (ang. Traffic Light System). Naloga TLS algoritma je, da v realnem času zazna prevelika odstopanja napetosti in to informacijo pošlje agregatorjem.

V praksi smo TLS algoritem razvili v NiFi-ju in njegove rezultate poslali FlexIS sistemu, ta pa jih posreduje naprej posameznemu agregatorju oz. VPP. Kot vhod TLS algoritma potrebuje realno časovne meritve napetosti,



idealno v več kritičnih točkah NN omrežja, v praksi pa smo imeli na voljo le meritve napetosti na SN/NN transformatorju. Tok podatkov se začne s komponento *ConsumeMQTT*, ki je naročena na MQTT sporočila katera vsebujejo meritve faznih napetosti, izmerjene s transformatorskim števcem. Te meritve se pošiljajo na enak način in v enakem CIM-XML formatu, kot je to opisano za meritve moči v poglavju 3.1. Prav tako na enak način izluščimo podatke o tovarniški številki števca, časovnih značkah meritev, registrih števca in vrednostih meritev. Te pretvorimo v JSON format. Razlika je le ta, da se meritve napetosti po MQTT pridobivajo na časovno granulacijo 10 minut, pri tem XML vsebuje ReadingType kode za sledeče 3 registre:

- fazno napetost v fazi L1 v voltih – *ReadingType* = 0.0.1.4.20.2.158.0.0.0.0.0.0.128.0.29.0
- fazno napetost v fazi L2 v voltih – *ReadingType* = 0.0.1.4.20.2.158.0.0.0.0.0.0.64.0.29.0
- fazno napetost v fazi L3 v voltih – *ReadingType* = 0.0.1.4.20.2.158.0.0.0.0.0.0.32.0.29.0

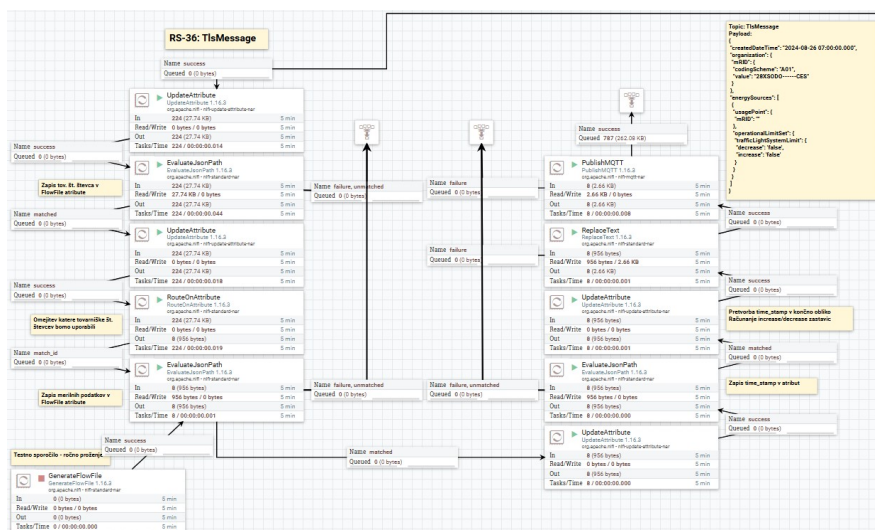
Meritve napetosti pretvorimo iz izvorne 10 minutne časovne granulacije na 15 minutno časovno granulacijo z namenom, da zagotovimo enako preiodiko pošiljanja realno časovnih meritev, napovedi obremenjenosti transformatorja in TLS vrednosti. Meritve napetosti uparimo na mRID šifrant TP-ja, da vemo za kateri TP bo TLS izračunaval vrednosti za zaustavitev ali dovoljenje izvajanja aktivacij. Pri izračunu TLS vrednosti uporabljamo 2 binarni zastavici in sicer *increase* in *decrease*. Logika izračuna teh zastavic je sledeča:

- Če so hkrati vse tri fazne napetosti manjše od 230 V + 10 % (t. j. 253 V) in večje od 230 V - 10 % (t. j. 207 V) nastavi zastavici na *increase* = *true* in *decrease* = *true*, to pomeni, da aktivacija prožnosti lahko poteka v obe smeri.
- Če je katerakoli izmed faznih napetosti manjša ali enaka 230 V - 10 % (t. j. 207 V) nastavi zastavici na *increase* = *false* in *decrease* = *true*, to pomeni, da aktivacija prožnosti lahko poteka le v smeri pozitivne aktivacije (A01) oz. zmanjšanje odjema ali povečanje proizvodnje.
- Če je katerakoli izmed faznih napetosti večja ali enaka 230 V + 10 % (t. j. 253 V) nastavi zastavici na *increase* = *true* in *decrease* = *false*, to pomeni, da aktivacija prožnosti lahko poteka le v smeri negativne aktivacije (A02) oz. povečanje odjema ali zmanjšanje proizvodnje.

Vrednosti zastavic *increase* in *decrease* izračunamo v NiFi komponenti *UpdateAttribute*, za vsako 15 minutno periodo za posamezni TP. Te podatke nadalje pretvorimo v JSON format, katerega zna brati FlexIS platforma. Sporočila imenovana *TlsMessage.json* s TLS vrednostmi zapišemo v MQTT ovojnico in jih pošljemo na FlexIS platformo po MQTT protokolu z uporabo NiFi komponente *PublishMQTT*. Slika 9 kaže končno obliko JSON sporočila, slika 10 pa kaže celoten tok procesiranja podatkov in izdelave TLS vrednosti z vsemi pripadajočimi NiFi komponentami.

```
{
  "createdDateTime": "2025-02-25T11:15:00Z",
  "organization": {
    "mRID": {
      "codingScheme": "A01",
      "value": "28XSODO-----CES"
    }
  },
  "energySources": [
    {
      "usagePoint": {
        "mRID": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
      },
      "operationalLimitSet": {
        "trafficLightSystemLimit": {
          "decrease": "true",
          "increase": "false"
        }
      }
    }
  ]
}
```

Slika 9: Primer TLS vrednosti zapisanih v *TlsMessage* JSON sporočilu za FlexIS.



Slika 10: NiFi komponente za zajem meritev napetosti, izračun TLS in pošiljanje TLS vrednosti.

### 3.5. Proženje aktivacij glede na njihove napovedi

Dejansko proženje aktivacij se izvaja skladno z njihovimi napovedmi, ki so opisane v poglavju 3.3. Sporočila *Activation* JSON pošljamo iz LAMBDE na FlexIS platformo in od tam naprej do VPP. Aktivacije morajo biti prožene minimalno 15 minut pred začetkom dobave prožnosti. V praksi trenutno z NiFi-jem aktivacijska sporočila izdelamo in testno pošljemo proti VPP že prejšnji dan ob 23:00 uri, ker je v sporočilih zapisan dejanski čas aktivacije, ki se bodo prožile šele naslednji dan. S takšnim načinom pošiljanja aktivacijskih sporočil nismo opazili težav, je pa bolj ugodno, da v prihodnosti v NiFi-ju pošiljanje *Activation* JSON sporočil posodobimo tako, da se bodo le-ta poslala proti VPP npr. 1 uro pred samo aktivacijo. To nam to zagotovi dodaten čas, da aktivacijo prekličemo / popravimo, če bi kratkoročna napoved pokazala, da bo transformator v resnici preobremenjen krajši čas ali pa sploh ne bo preobremenjen.

Pri izdelavi *Activation* JSON sporočil so osnova podatki iz tabele *d\_napoved\_dogodkov* v Kudu podatkovnem jezeru na LAMBDE. Iz te tabele zajamemo podatke za posamezno napoved aktivacije in jih s komponento *EvaluateJsonPath* zapišemo v zahtevano JSON strukturo. Pri tem definiramo tudi tip produkta, ki ga želimo prožiti. Aktivacijo označimo tudi z mRID TP (in mRID nadrednega RTP) za katero bomo prožili aktivacijo. Za samo aktivacijo moramo definirati časovno serijo moči, s katero bomo vsakih 15 minut aktivirali vire prožnosti. V ta namen za časovne značke definirane v napovedi aktivacij uporabimo kar časovno serijo napovedi obremenjenosti transformatorja in od nje odštejemo nazivno moč transformatorja. S tem poenostavljeno dobimo časovno serijo preobremenjenosti transformatorja kar je tudi enako količini moči, ki jo moramo aktivirati, da bomo transformator razbremenili. Takšna je trenutna implementacija, ki kot že prej opisano temelji na napovedi z algoritmom soležnih dni. V praksi bi vsekakor bilo ugodneje, da bi napovedni model obremenjenosti transformatorja izdelal napoved tik pred aktivacijo, saj bi ta bila točnejša, iz te napovedi bi nato računali preobremenjenost oz. časovno serijo moči potrebne za aktivacijo.

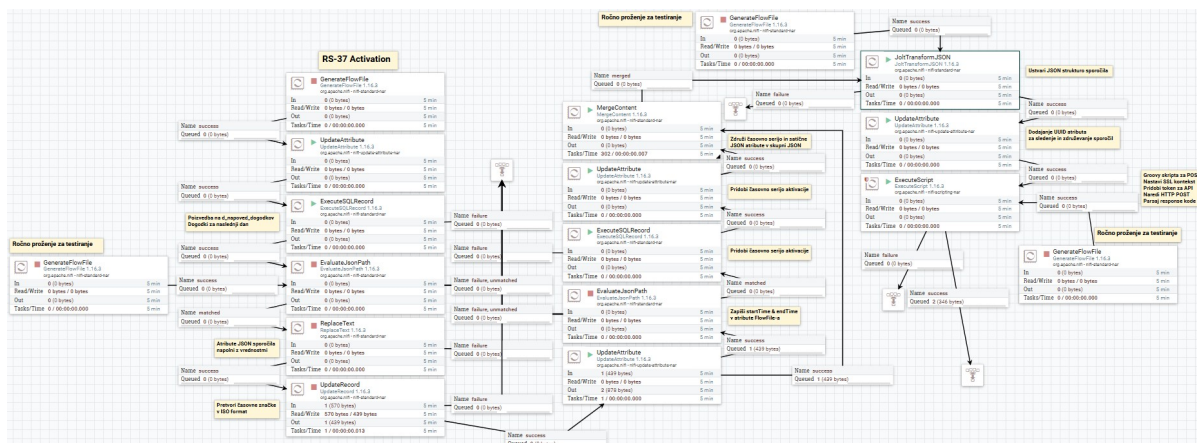
Pridobljeno časovno serijo moči z NiFi komponento *JoltTransform* pretvorimo v ustrezen JSON format in jo združimo z ostalimi JSON parametri, ki smo jih pridobili iz tabele *d\_napoved\_dogodkov*. Sama časovna serija aktivacije je v JSON sporočilu zapisana v *point* atributu s pari podatributov *position* in *quantity*. Izdelanim JSON sporočilom dodamo še časovno značko kdaj se je sporočilo ustvarilo in jim dodamo UUID identifikator, da jim lahko sledimo v FlexIS sistemu. *Activation* sporočila objavimo na REST API storitev FlexIS platforme z uporabo POST http metode. Slika 11 kaže končno obliko JSON sporočila, slika 12 pa kaže celoten tok procesiranja podatkov in izdelave *Activation* JSON sporočila z vsemi pripadajočimi NiFi komponentami.

```

{
  "mRid": "00005c6d-9804-4651-966a-56a866705833",
  "activationSignal": "activation",
  "productType": "PredictiveCongestionManagementCapacity",
  "activatedPower": 10.58,
  "areaId": {
    "rtp": "83b72764-7db8-4fbf-be9a-58bb643603f7",
    "tp": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "mm": ""
  },
  "timePeriod": {
    "end": "2025-02-26T13:15:00.000Z",
    "start": "2025-02-26T11:15:00.000Z"
  },
  "timeSeries": [
    {
      "mRid": "d100122c-dd00-40d1-bc1b-604a0115952c",
      "measurementUnitName": "KAW",
      "flowDirection": "A01",
      "period": [
        {
          "timeInterval": {
            "end": "2025-02-26T13:15:00.000Z",
            "start": "2025-02-26T11:15:00.000Z"
          },
          "resolution": "PT15M",
          "point": [
            {
              "position": "1",
              "quantity": 6.45
            },
            ...
          ]
        }
      ]
    }
  ]
}

```

Slika 11: Primer vrednosti zapisanih v *Activation* JSON sporočilu za FlexIS



Slika 12: NiFi komponente za izdelavo in pošiljanje *Activation* JSON sporočil.

#### 4 KIBERNETSKA VARNOST SISTEMOV V VERIGI UPRAVLJANJA S PROŽNOSTJO

Kljub temu, da se podatki na relacijah HES, SmenDataBox, LAMBDA / NiFi in FlexIS izmenjujejo znotraj poslovnega omrežja Elektra Celje, smo h implementaciji pristopili z ustrežno mero zagotavljanja kibernetske varnosti. MQTT povezave med SmenDataBox in NiFi-jem ter MQTT in HTTP povezave NiFi-jem in FlexIS so zaščitene s šifriranjem na TLS (ang. Transport Layer Security) nivoju, katerega implementirajo nekateri MQTT brokerji ter HTTP strežniki. Avtentikacija na MQTT nivoju je zagotovljena s poverilnicami, medtem ko avtentikacija na REST spletne storitve poteka z uporabo izdaje avtentikacijskih žetonov. FlexIS implementira avtentikacijski strežnik na katerega se mora NiFi pred vsakim klicem REST API-ja avtentificirati s poverilnicami in pridobiti avtentikacijski žeton. Žeton je veljaven samo enkrat in samo za specifično spletno storitev (ang. endpoint) API-ja. S tem žetonom se NiFi avtentificira na določeno REST spletno storitev na kateri izvaja izmenjavo podatkov. Takšna funkcionalnost REST izmenjave v osnovi ni podprta v NiFi-ju, zato smo jo v celoti izvedli s komponento *ExecuteScript*, ki za vsak POST klic požene skripto napisano v programskem jeziku Groovy. Povezave med FlexIS in EVT potekajo preko Kafka stream vmesnikov, ki zaščitijo omrežni promet s šifriranjem na TLS nivoju.

## 5 ZAKLJUČEK

V referatu smo predstavili vse sisteme in njihove medsebojne povezave ter izmenjavo podatkov, ki je potrebna v verigi upravljanja s prožnostjo. Podrobneje smo opisali implementacijo zalednih procesov za avtomatsko upravljanje prožnosti, ki smo jih izvedli z orodjem Apache NiFi, ki je del LAMBDA velepodatkovne platforme v Elektru Celje. Storitve, ki tečejo na predstavljeni informacijski infrastrukturi so prestale testiranja posameznih storitev, trenutno pa smo v fazi testiranja v realnem svetu, kjer pričakujemo, da bomo v aprilu 2025 uspešno zaključili prve end-to-end teste celotne verige. Pred poletjem 2025 želimo izvesti dejanske aktivacije prožnosti v pilotnih TP postajah.

Tekom razvoja smo pri implementaciji FlexIS in zalednih procesih na NiFi prepoznali tudi določene pomanjkljivosti. Ostaja nam še nekaj procesov, ki trenutno niso implementirani vendar je njihova implementacija predvidena v prihodnosti, kot na primer predkvalifikacija virov prožnosti. Trenutno lahko prožimo le nekvalificirane vire. Za izvajanje predkvalifikacijskih postopkov bo potrebno nadgraditi tako procese v NiFi, kot tudi v FlexIS. Dodatno bomo za izvajanje regulacije napetosti

Algoritem za napovedovanje bomo v prihodnosti zamenjali z algoritmi strojnega učenja, kjer je poleg napovedi obremenitev potrebno implementirati tudi napovedni model meritev napetosti na transformatorju, saj bo to osnova za napovedovanje aktivacij produktov napetostne regulacije. Dodatno je smiselno pri preobremenitvah upoštevati modele za DTR (ang. Dynamic Thermal Rating) transformatorjev, saj le ti dopuščajo preobremenitve, s čimer bi zmanjšali število aktivacij in povečali prihranek stroškov za izvajanje prožnosti.

Vsekakor bi bilo bolj pravilno, da meritev transformatorskega števca vežemo na posamezni SN/NN transformator in ne na TP, kot je trenutno implementirano. En TP lahko vsebuje več transformatorjev, vendar v začetnih fazah planiranja nismo predvideli, da bo prožnost potrebno prožiti tudi na dvo-transformatorskih TP-jih. Ta koncept je stvar nadgradnje FlexIS platforme, ki bo izveden v prihodnosti.

Ocena obratovalnega stanja NN omrežij (TLS algoritem) je trenutno izvedena v LAMBDA. Najpogosteje računanje obratovalnih stanja omrežij izvajajo funkcionalnosti ADMS (ang. Advanced Distribution Management System) sistemov, ki so implementirane tudi v Elektru Celju. Praktični problem pri implementaciji bolj točne ocene obratovalnega stanja NN omrežij z ADMS pa predstavlja sam način njihovega ocenjevanja. Namreč ADMS v modelu obremenitev končnih točk NN omrežja uporablja nadomestne krivulje, ki so le grob približek realnega stanja in se običajno posodablja 1x do 2x letno. Z nadomestnimi krivuljami ni mogoče točno napovedati obratovalnega stanja in s tem TLS vrednosti. Če bo ADMS sistem v prihodnosti podprl uvoz realno časovnih meritev iz transformatorskih števecov je smiselno oceniti ali se lahko TLS algoritem izvaja v sklopu standardnih ADMS funkcij.

## 6 ZAHVALA

Projekt EV4EU financira program Horizon Europe v okviru pogodbe 101056765 / Funded by the European Union under grant agreement no. 101056765.

## 7 LITERATURA – LEON UREDI!

- [1] Študija z navodili za prožnost – SODO, št. 2528
- [2] <https://webstore.iec.ch/en/iec-search/result?q=IEC%2062325>
- [3] Članek D3-223
- [4] IEC 61968-9 <https://webstore.iec.ch/en/publication/75041>
- [5] Apache NiFi