

Domagoj Peharda  
Končar-KET  
[domagoj.peharda@koncar-ket.hr](mailto:domagoj.peharda@koncar-ket.hr)

Ivan Krajnović  
Končar-KET  
[ivan.krajnovic@koncar-ket.hr](mailto:ivan.krajnovic@koncar-ket.hr)

## POTREBE NAPREDNIH SUSTAVA U UPRAVLJAČKIM CENTRIMA ZA VOĐENJE ELEKTROENERGETSKOG SUSTAVA U SVRHU RAZMJENE MODELA

### SAŽETAK

U članku je predstavljeno praćenje verzija podatkovnog modela energetskog sustava koje omogućava ispravak vrijednosti u prošlosti, ali i pogled u prošlost kakva bi bila bez tog ispravka. Predstavljeno praćenje verzija omogućava i pogled u budućnost u obliku planiranih izmjena na modelu te testnih izmjena. Tako bi se stvorenim verzijama modela dobilo više pogleda na model, dispečerski pogled koji omogućava pogled u model kakav je bio u prošlosti, analitičarski pogled koji omogućava pogled u prošlost no s ispravljenim parametrima elemenata ukoliko su pogrešno bili upisani, projektantski pogled koji omogućava pogled na model za koji je planirano da bude u budućnosti, testni pogled koji omogućava pregled utjecaja izmjena koje se unose u model no nisu završene. Spremanjem svih verzija modela omogućeno je i jednostavnije kreiranje CGMES izvoza kojeg EU zahtijeva za razmjenu modela energetskog sustava između TSO i DSO.

**Ključne riječi:** IEC61970, CIM, CGMES, modeliranje sustava, verzije

## NEEDS OF ADVANCED POWER SYSTEM CONTROL CENTERS FOR PURPOSE OF MODEL EXCHANGE

### SUMMARY

Paper shows the versioning of data model of electrical power system which allows for correction of data in the past, but also a view into the past without said correction. Versioning enables a view into the future in the form of planned changes on the model and also applying test changes. Versioning thus enables multiple views on the model, dispatcher view where model is the model how it was in the past, analytical view where the model is corrected model in which parameters are corrected, and project view where the model is in the future with applied planned changes, and test view where model is augmented by changes that are not applied to the model. Versioning allows for simple creation of the CGMES export which the EU is requiring for exchange between TSOs and DSOs.

**Key words:** IEC61970, CIM, CGMES, system modelling, versioning

## 1. UVOD

Elektroenergetski sustavi su izuzetno povezani, pogotovo je jak utjecaj između operatera distribucijskog sustava i operatera prijenosnog sustava. Također se elektroenergetski sustavi kontinuirano mijenjaju, te zbog međusobne povezanosti sustava promjene u jednom sustavu jako utječu na ostale sustave. Dodatno, zbog razvoja računalne snage, modeli elemenata postali puno detaljniji i nadzirana vanjska mreža počinje biti sve šira, te zahvaća velikim dijelom i mrežu distribucijskog sustava kod operatera prijenosnih sustava i mrežu prijenosnog sustava kod operatera distribucijskih sustava. Kod operatera distribucijskih sustava modeli postaju višestruko veći i složeniji zbog upravljanja „po dubini mreže“ i distribuiranih izvora). Potreba za detaljnim modeliranjem unutarnje i vanjske mreže uvidjela se velika potreba za standardiziranjem razmjene podataka o modelu elektroenergetskog sustava. Također sve je više vanjskih programskih sustava unutar samih organizacija koje se moraju povezati ili koriste model elektroenergetskog sustava ili njegove dijelove. Zbog utjecaja tržišta veliki broj programskih sustava koji moraju komunicirati sa sustavima za vođenje elektroenergetskih mreža su financijski sustavi stoga je potrebno prilagoditi model tako da se može koristiti za financijske sustave. U financijskom sustavu je već dugo vremena poznata potreba za praćenjem dvaju vremena, vrijeme kada je promjena nastupila (postala validna) i nasuprot vremenu kada je korisnik sustava saznao za tu promjenu (kada je ta promjena upisana u sustav). Oba vremena su nužna za financijske sustave stoga je potrebno oba vremena pratiti i u modeliranju elektroenergetskih sustava. Praćenje vremena unosa i vremena važenja omogućava i različite poglede na model elektroenergetskog sustava, a i omogućuje lakše manipuliranje verzijama. Korisnici također žele sposobnost da u modelu isprave pogrešan unos parametara ili topologije u prošlosti na način da se sačuva model koji je postojao u prošlosti. U tom slučaju je moguće birati dva pogleda na model u prošlosti: onakav kakav je dispečer vidio u prošlosti prije nego što su se primijenile ispravke modela i/ili mjerenja; i drugi pogled na ispravljeni model i/ili mjerenja. Dok se izvedba baze podataka za praćenje verzija modela mreže može izvesti tradicionalnim relacijskim bazama, za spremanje svih podataka u onom obliku koji je potreban za pogled na neispravljena i ispravljena mjerenja potrebna je posebna baza podataka često zvana povijesna baza iliti *eng. historian*. Također je važno prepoznati da isti element može imati veliki broj različitih reprezentacija zbog podrške u raznim drugim sustavima, stoga je potrebno omogućiti višestruka imena.

### 1.1. CIM

Common Information Model (CIM) je ime za skup otvorenih standarda korištenih za razmjenu modela EES-a. Razvija se unutar radnih grupa tehničkog odbora IEC TC57. CIM ima tri osnovne upotrebe: za razmjenu podataka o modelu EES-a između organizacija, za razmjenu podataka između aplikacija unutar organizacija, te za razmjenu tržišnih podataka između organizacija [1].

CIM je podijeljen u tri dijela. Prvi dio je ukupni UML model sadržan u standardu IEC 91970-301. Drugi dio čini skupina standarda IEC 91970-4XX, zvani profili, koji isijecaju dijelove modela za pojedine svrhe (npr. Equipment profile IEC 91970-452, Study State Hypotesis profil IEC 91979-456, itd.). Treći dio čini skup standarada IEC 91970-5XX za pohranjivanje (serijalizaciju) u datoteke. Trenutno se za razmjenu podataka o modelu koristi standard IEC 91970-552 koji opisuje način serijalizacije CIM modela nazvan CIMXML. CIMXML je baziran na RDF standardu i omogućava razmjenu punog modela i diferencijalnog modela (samo promjene). Postoje i drugi standardi serijalizacije poput IEC 91968-11 koji omogućava razmjenu XML poruka prema XSD specifikaciji koji se više koristi u distribuciji u svrhu razmjene kraćih poruka.

CIM model je prihvatila udruga europskih operatera prijenosnih sustava ENTSO-E za novi format za razmjenu podataka koji bi zamijenio trenutni UCTE-def format. CIM model je proširen sa ENTSO-E ekstenzijama i nazvan CGMES format.

### 1.2. Praćenje verzija u CIM-u

Trenutno CIM model ne podržava praćenje verzija no podržava razmjenu promjena na modelu što se može primijeniti kao praćenje verzija promjena u prošlosti, te podržava razmjenu projekata koji podržavaju razmjenu potencijalnih promjena (planiranih promjena ili promjena u svrhu provedbe raznih analiza).

Model projekta u CIM-u se još uvijek razvija, te u CIM verziji 16, na kojoj je razvijen CGMES 2.4.15, nije upotrebljiv za razmjenu potencijalnih izmjena. Zbog toga je razmjena projekata zamišljena za sljedeću verziju CIM-a i sljedeću verziju CGMES formata.

Razmjena projekata je preduvjet za potpunu implementaciju CGMES formata u TYNDP i MLTOP. TYNDP – Ten Year Network Development Plan je ukupni desetgodišnji plan za razvoj europske prijenosne mreže gdje prijenosni operatori prilažu svoje planove razvoja prijenosne mreže. MLTOP –

Medium and Long Term Outage Planning Tool je alat koji bi trebao olakšati i automatizirati razmjenu podataka o planiranim radovima i mogućim ispadima u europskoj prijenosnoj mreži.

## 2. MODEL ELEKTROENERGETSKOG SUSTAVA S PRAĆENJEM VERZIJA

### 2.1. Hijerarhijsko imenovanje

Moderni model elektroenergetskog sustava mora biti sposoban razmjenjivati podatke sa vanjskim sustavima zbog čega je potrebno omogućiti više imena za svaki element. Također je važno da elementi mogu preuzeti dio imena od nadređenog elementa, tj. potrebno je hijerarhijsko imenovanje gdje svaki element može imati samo jednog roditelja, no može biti član više hijerarhija (tj. imati više imena). Svaka hijerarhija omogućuje imenovanja zbog različitih potreba. Ako broj hijerarhijskih imenovanja nije ograničen moguće je iskoristiti imenovanja za različite potrebe, prvenstveno za šifre istih elemenata u drugim sustavima što omogućava jednostavniju razmjenu podataka. Važna karakteristika hijerarhijskog imenovanja jest da unutar hijerarhije nisu moguća dva ista imena.

Općenito, u hijerarhijskom imenovanju nema ograničenja na broj nivoa, broj znakova u imenu, te svaki element može biti nadređeni ili podređeni nekom drugom elementu. No prilikom korištenja sustava svako hijerarhijsko imenovanje može biti ograničeno prema različitim postavkama, gdje su te postavke ograničenja u inženjeringu podataka i projektu za svaki sustav, a ne u modelu:

- tipovi elemenata u svakom nivou
- broj znakova
- za svaki element mogući tipovi nadređenog elementa
- za svaki element mogući tipovi podređenih elemenata
- korištena kodna stranica za znakove
- znak(ovi) korišten(i) za razlikovanje nivoa hijerarhije

U sljedećoj tablici su dani primjeri imena za neke elemente sustava:

Tablica 1 Primjer hijerarhijskih imenovanja

<b>hijerarhija</b>	<i>tip elementa transformator</i>	<i>tip elementa generator</i>
<b>puno ime</b>	Elektra Zagreb.Podused.TR1	Hrvatska.PrP Split.HE Zakućac.Gen1
<b>skraćeno ime</b>	PS.TR1	HR.ST.HEZAK.GEN1
<b>CGMES ime</b>		HHEZAK GEN1
<b>mrid</b>	63a53a17-413c-4c20-938f-5c4f3eda12b0	9c6b5b8a-ed8-4346-8187-5fd479b3c28a
<b>lokalno ime</b>	04T17.301	
<b>inventarski broj</b>	1023003	
<b>serijski broj opreme</b>	Končar.122039	Končar-GIM.223332
<b>Tržišno ime</b>		HEP Proizvodnja.Jug.HE Zakućac.Gen1

Svaki element ima interni identifikacijski broj i obavezno imenovanje u glavnoj hijerarhiji za taj tip elementa, dok je sudjelovanje u ostalim hijerarhijama opcionalno osim ako nije drugačije navedeno u projektu. Interni identifikacijski broj je nepromjenjiv, dok se imena mogu dodavati, mijenjati i brisati. Promjena imena nekog elementa uzrokuje promjenu imena svih njegovih podređenih elemenata. Izuzetno je važno svojstvo da je ime elementa jedinstveno unutar jedne hijerarhije i verzije. Znači da je moguće nekom elementu promijeniti ime u ime elementa koje je već postojalo u nekoj prijašnjoj verziji.

Često je potrebno grupirati neke elemente u logičku grupu, koja služi samo za lakšu čitljivost i manipulaciju grupama elemenata. Zbog toga u modelu elektroenergetskog sustava mora postojati općenita logička grupa u svrhu hijerarhijskog imenovanja i grupiranja elemenata. Logičko grupiranje ima svrhu da se pojavljuje u hijerarhijskom modelu u svrhu imenovanja ili u svrhu grupiranja raznovrsnih elemenata. Najčešće se logičke grupe koriste za grupiranje signala o pomoćnim sustavima unutar stanica i za grupiranje na osnovi raznih geografskih ili vlasničkih kriterija. Ukoliko model elementa za logičko

grupiranje ne postoji onda se u svrhu logičkog grupiranja upotrebljavaju elementi koji imaju posebno značenje, npr. za grupiranje signala o besprekidnom napajanju se stvori naponski nivo (koji u stvarnosti ne postoji) u koji se stave ti signali u svrhu imenovanja i grupiranja.

Potrebno je naglasiti da su hijerarhija i imena potrebna samo zbog razmjene podataka, bolje čitljivosti i lakše manipulacije s grupama elemenata. Za proračun topologije i same algoritme ostalih proračuna, grupiranje i imena nisu važna. Za promatranje rezultata proračuna, grupiranja i hijerarhije su važna, npr. prilikom izračuna ukupne potrošnje za neku regiju.

Višestruko imenovanje i hijerarhije sa logičkim grupama omogućavaju jednostavno povezivanje sa TIS – Technical Information System i Asset Management sustavima.

Glavne značajke hijerarhijskog imenovanja:

- element može imati više imena
- element može imati podređene elemente
- ime se gradi od svih elemenata u hijerarhiji, promjena jednog imena znači promjenu za sve podređene elemente
- ime je jedinstveno unutar hijerarhije za pojedinu verziju
- moguće je kreirati elemente koji služe za logičko grupiranje drugih elemenata
- broj i vrste hijerarhija, te njihove karakteristike se definiraju posebno za svaki projekt

### **2.1.1. Vlasnička hijerarhija**

Osim hijerarhije imenovanja kojoj je glavna značajka da element može imati samo jedan nadređeni element postoji potreba i za hijerarhijom gdje element može imati više nadređenih elemenata gdje svaki nadređeni element ima udio u podređenom elementu. Ukupni udio svih nadređenih elemenata mora biti 1. Ova hijerarhija se naziva vlasničkom hijerarhijom jer se pomoću takve hijerarhije može odrediti udio vlasništva po pojedinom elementu.

Klasično korištenje takve hijerarhije jest za izračun gubitaka kompanije gdje se gubici na nekim elementima poput vodova i transformatora dijele između više kompanija. U novom, tržišnom, načinu rada proizvodnja i potrošnja u nekim elementima također može imati više vlasnika stoga je vlasnička hijerarhija važna i za izračune proizvodnje i potrošnje kompanija.

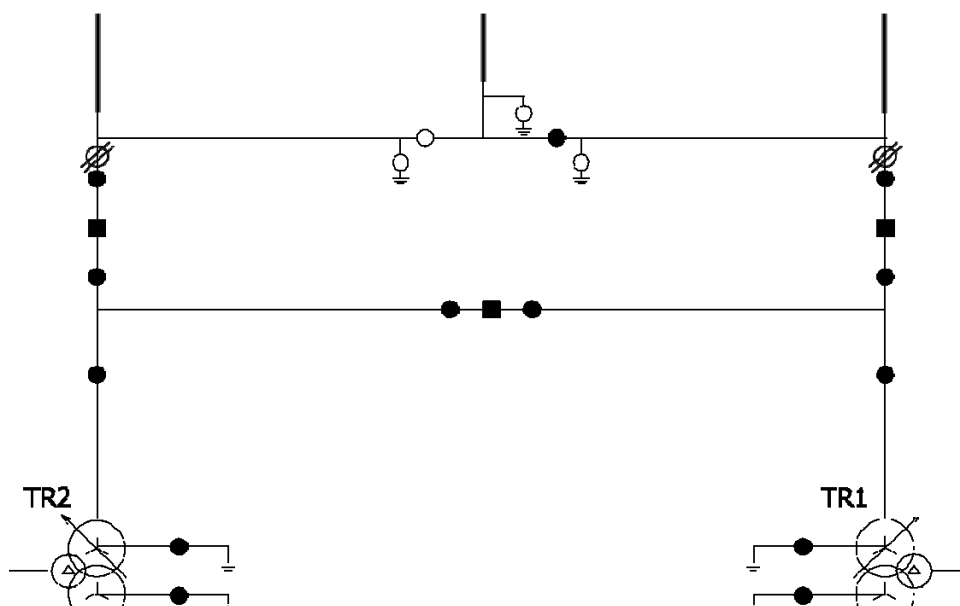
### **2.1.2. Dozvole**

Hijerarhija omogućava jednostavno dodjeljivanje dozvola u obliku uloga→dozvola→element u hijerarhiji. Podrazumijeva se da se dozvola odnosi i na sve podređene elemente tom elementu u navedenoj hijerarhiji. Ukoliko je potrebno napraviti dozvole za posebne elemente moguće je upotrijebiti posebnu hijerarhiju za pojedinačno dodjeljivanje dozvola. Naravno, dodjeljivanje dozvola je također potrebno pratiti u obliku verzija.

## **2.2. Mjerni uređaji kao posebni topološki elementi**

Mjerenja napona, struje i snage se dohvaćaju iz mjernih transformatora. Mjerni transformatori su najčešće smješteni zasebno od ostale opreme, te nije uvijek moguće unaprijed pripisati mjerenje nekom elementu u mreži već je to ovisno o topološkom stanju mreže. Zbog tih razloga mjerenja treba pripisivati mjernim uređajima koji mogu u mrežu biti spojeni u seriju (tj. dvije priključnice) ili paralelno (jedna priključnica). Za potrebe proračuna procjene stanja topološki procesor treba pripisivati mjerenje elementima mreže koji ulaze u proračun procjene stanja.

Problem topološkog smještaja mjernih uređaja je uočen i od strane radne grupe WG13 te je u CIM modelu predložen novi element FlowSensor sa dvije priključnice na koje je moguće spojiti mjerenja koja imaju dvije priključnice.



Slika 1. Primjer topološkog smještaja mjernih uređaja (krug sa kosim crtama) gdje pripisivanje mjerenja elementu (vodu ili transformatoru) ovisi o statusu prekidača (kvadrati) i rastavljača (krugovi)

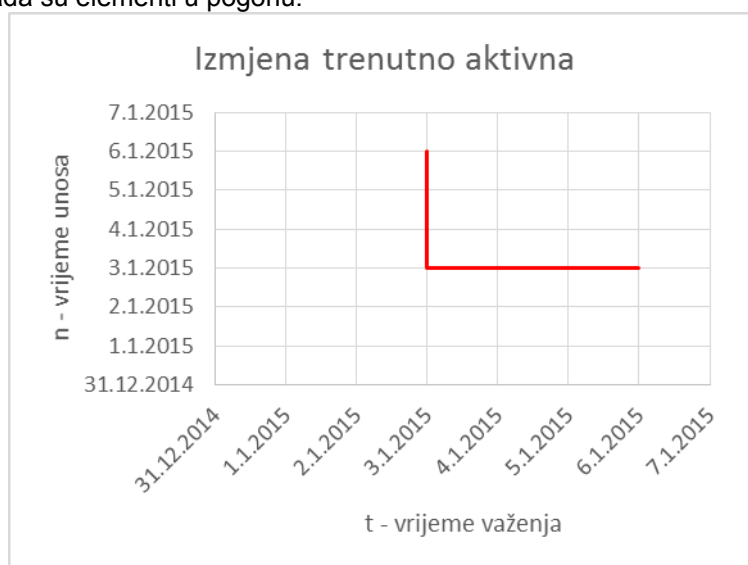
Dodatno, kreiranje mjernih uređaja kao posebnih topoloških elemenata, omogućava stvaranje posebnog identifikacijskog broja i imena koje je moguće koristiti nadalje u TIS-u i upravljanju imovinom.

## 2.3. Najčešći slučajevi modeliranja elektroenergetskog sustava i korištenje u sustavima za vođenje

### 2.3.1. Dodatak ili izmjena elemenata koji postoje no nisu modelirani

Ovo je najčešća akcija nad modelom. Postoji dio mreže koji još nije u modelu, te je potrebno dodati u model te izmjene najčešće prije nego što se taj dio modela stavi u pogon. Skup promjena autor promjena provjerava u alatima za analizu mreže i svim ostalim alatima za simulaciju koji su dostupni.

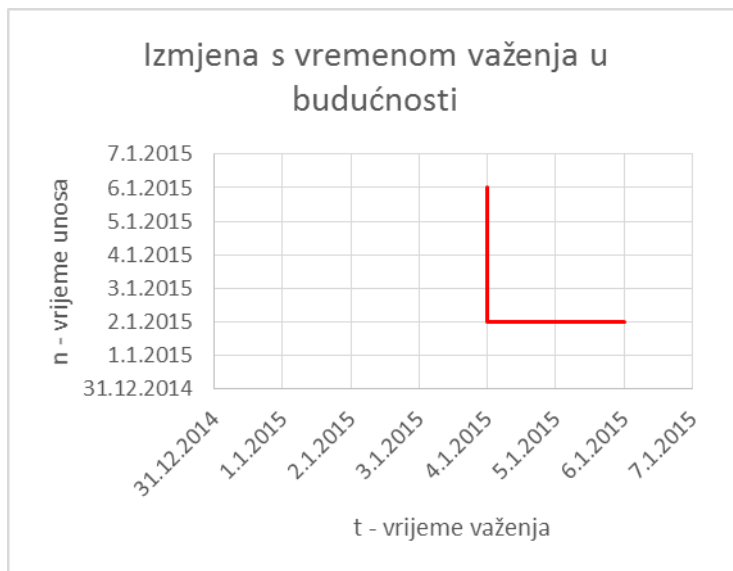
Nakon provjere koje je učinio sam autor promjena, prije nego što se promjene učine aktivnima, korisnik zadužen za aktiviranje promjena testira promjene u testnom okruženju, te ukoliko je sve korektno modelirano aktivira promjene i podiže verziju u sustavu. Elementi koji su izmijenjeni u tom skupu promjena za vrijeme izmjene dobivaju vrijeme aktivacije skupa promjena ili, ako je poznato, može se upisati vrijeme od kada su elementi u pogonu.



Slika 2. Izmjena koja je unesena 3.1.2015 i važi od 3.1.2015

### 2.3.2. Dodatak ili izmjena elemenata za koji je poznat trenutak ulaska u pogon u budućnosti

Poznat je trenutak kada će neka izmjena postati aktivna, te je potrebno tu izmjenu unaprijed unijeti. To može biti izmjena svojstva nekog elementa koji će biti zamijenjen u budućnosti kada neće biti moguće aktivirati tu promjenu. Npr. zamjena transformatora koja se događa u tijeku noći kada nije prisutan korisnik koji aktivira promjene.



Slika 3. Izmjena koja je unesena 2.1.2015 i ne počinje važiti tek od 4.1.2015

### 2.3.3. Ispravak modela u prošlosti

Kreira se skup promjena sa ispravkom parametara ili čak dijelova mreže koji su krivo modelirani zbog različitih razloga (pogrešne ili nepoznate informacije, pogreške u modeliranju). Taj skup promjena mora imati definirano vrijeme u prošlosti od kada će skup promjena vrijediti za buduće analize (to vrijeme može biti i od stvaranja ukupnog modela).

Postupak aktivacije skupa promjena radi korisnik zadužen za aktiviranje promjena. Promjene se testiraju za vrijeme u prošlosti definirano za taj skup promjena, te za proizvoljno odabrana vremena od strane korisnika. Prilikom aktivacije elementi koji su izmijenjeni u tom skupu promjena za vrijeme izmjene dobivaju vrijeme definirano u skupu promjena.



Slika 4. Prvi unos elementa se dogodio 1.1.2015 od kada je i vrijeme važenja, ispravak pogrešnog unosa se dogodio 3.1.2015 i taj ispravak počinje važiti od 1.1.2015

#### **2.3.4. Potencijalne izmjene**

Potencijalne izmjene su skup promjena u kojem se nalaze promjene koje su tek u fazi planiranja i koje ne moraju biti u konačnici izvedene, tj. projekt se može u nekom trenutku prekinuti. Taj skup promjena ne smije biti aktiviran no mora se omogućiti razmjena tih potencijalnih promjena (planova) sa susjednim operatorima sustava da bi i oni mogli planirati svoje promjene. Nakon što autor modelira i testira planirane izmjene one postaju dostupne ostalim analitičarima unutar kompanije, te je te promjene potrebno moći razmijeniti sa susjednim operaterima, najčešće u obliku CIM projekta.

Skup promjena s potencijalnim izmjenama mora imati datum kada su navedene izmjene planirane da postanu aktivne. Sve analize koje ostali analitičari unutar kompanije pokreću za vrijeme nakon toga datuma automatski sadržavaju navedeni skup planiranih promjena, te mogu taj skup uzeti u obzir ili odbaciti prilikom analiza u budućnosti.

Nakon što sadašnjost dostigne planirano vrijeme izmjena tada taj skup postaje običan skup promjena bez datuma.

#### **2.3.5. Brisanje starih verzija**

Model mora omogućavati brisanje starih verzija modela, gdje se sve verzije prije određene verzije brišu i stapaju u novo početno stanje. Npr. prije puštanja novog sustava u pogon potrebno je izbrisati sve akcije koje su rađene u svrhu kreiranja modela ili se žele odbaciti stari podaci radi uštede prostora.

#### **2.3.6. Skup razlika između verzija**

Ponekad je potrebno znati koje su se izmjene dogodile između neke dvije verzije modela. Često se događa da se pojavi neki problem u modelu koji nije vidljiv prilikom aktivacije već se pojavljuje nakon nekog vremena. Da bi se utvrdio uzrok problema korisno je znati koje su se izmjene dogodile između neke dvije verzije modela.

Modeli elektroenergetskih sustava postaju sve veći zbog čega u IEC 61970-552 CIMXML standardu serijalizacije CIM modela je moguće razmjenjivati samo promjene što omogućava značajnu uštedu u količini razmijenjenih podataka. Za razmjenu samo promjena mora biti moguće dohvatiti izmjene modela između neke dvije verzije modela gdje se izmijenjeni elementi razmjenjuju sa svim parametrima.

#### **2.3.7. Analiza događaja u prošlosti**

Postoji potreba za analizom događaja u prošlosti sa ispravljenim modelom i ispravljenim mjeranim podacima.

#### **2.3.8. Pogled u prošlost kako ju je vidio dispečer u tom trenutku**

Nasuprot analizi događaja u prošlosti gdje upotrebljavamo ispravljeni model i ispravljene mjerene podatke ponekad je potrebno utvrditi što je dispečer mogao vidjeti i analizirati u danom trenutku u prošlosti. Tada je moguće analizirati što je potrebno izmijeniti u sustavu da bi dispečeri imali bolji uvid u stanje mreže.

#### **2.3.9. Usporedba rezultata prognoze zagušenja sa stvarnim tokovima snaga**

Proračun prognoze zagušenja omogućava uvid dispečerima u moguće buduće probleme sa mrežom. Da bi bilo moguće utvrditi kvalitetu proračuna prognoze zagušenja potrebno je uspoređivati stvarne tokove snaga u mreži sa rezultatima proračuna prognoze zagušenja za taj trenutak. U tu svrhu potrebno je imati pogled u prošlost gdje su dostupne samo one informacije koje su stigle do tog trenutka u prošlosti, izračunati prognozu zagušenja, te ju usporediti sa stvarnim tokovima snaga za promatrani trenutak.

### **2.4. Praćenje verzija modela u bazi podataka**

#### **2.4.1. Praćenje verzija**

Klasične baze modela EE sustava zapisivale su jednu (zadnju) sliku sustava čime je bilo nemoguće interpretirati povijesne podatke (npr. mjerenja) u kontekstu promjena na modelu koje su se dešavale.

Potreba za bilježenjem promjena i unošenjem tih promjena unaprijed ili ispravak u prošlosti je u financijskom svijetu riješeno korištenjem tkz. bitemporalnog praćenja verzijaverzioriranja [2]. Tako se za implementaciju baze elektroenergetskog modela predlaže korištenje bitemporalno praćenje verzija kojim

se svakoj izmjeni u bazi pridružuju 2 vremenska trenutka: vrijeme unosa (vrijeme aktivacije promjena u bazi)  $n$  i vrijeme važenja (procesno vrijeme)  $t$ .

Cilj ovakvog modela je zapis potpune povijesti izmjena nad sustavom s mogućnošću dohвата slike sustava u bilo kojem trenutku  $model(n, t)$ , tj. neovisno kretanje kroz procesno vrijeme i verzije unosa parametrizacijom modela sustava sa dva neovisna parametra.

U kontekstu modela EE sustava vrijeme važenja je stvarno procesno vrijeme od kojeg činjenica (izmjena) važi, npr. "Od 20.5. u 12h važe parametri voda A.B.C=...". Vrijeme unosa je vrijeme zapisa (aktivacije) u bazu, npr. "19.5. u 10h unesene su promjene parametra voda A.B.C, koje važe od...".

Vrijeme unosa može prethoditi vremenu važenja (unos promjena koje postaju aktivne u budućnosti), biti jednako vremenu važenja (promjene koje postaju trenutno aktivne) ili kasniti za vremenom važenja (unos retroaktivnih promjena). Pritom je jedino bitno da se unesena promjena dodaje sa stvarnim strogo rastućim vremenom unosa.

U nekom trenutku  $(N, T)$  važeća je izmjena  $X(n, t)$  za koju vrijedi da je promatrani procesni trenutak  $t \leq T$ , promatrani trenutak izmjene  $n \leq N$ ,  $\max n, t$ .

Valja napomenuti da neki modeli ili same SQL baze podataka omogućuju vraćanje u prošlost koristeći samo jedan vremenski parametar (efektivno postavljajući  $T := N$ ) što je i dalje nepotpuno. Ovakav model tjera korisnika da izmjene u bazi unosi točno u trenutku kad se izvode fizičke izmjene na sustavu te onemogućuje retroaktivne izmjene kao i unošenje promjena iz budućnosti.

#### 2.4.2. Modeliranje elemenata

Fizički EE elementi imaju ime, svoje topološko mjesto u mreži i svojstva za taj element. Najčešće se ime i hijerarhijski smještaj unose posebno od unosa topološkog smještaja i svojstava za neki element. Potrebno je odvojiti unos elementa sa svojim imenom i hijerarhijskom povezanošću od njegovih svojstava. Topološki smještaj svih elemenata je potrebno također posebno unositi. Često su svojstva elemenata ista za veliki broj elemenata te je potrebno svojstva grupirati.

Gore navedeni zahtjevi nas navode da je model u bazi potrebno izdvojiti u tri tipa tablica: element, relacija i tip. Element predstavlja jedinstveni identifikator elementa, tip je set fizikalnih parametara koji opisuju neki element ili više njih, a relacija je pridruživanje tipova elementima. Dodatno, topologija i hijerarhije opisane su u posebnim tablicama.

Bitno je napomenuti da sama tablica elementa sadrži isključivo jedinstveni identifikator, tj. prazno mjesto za element bez fizikalnih i drugih karakteristika.

Ovakvom implementacijom razdvojen je identifikator elementa, fizički parametri, imena i pripadnosti i topologija čime je omogućena velika fleksibilnost te korektno i neovisno modeliranje različitih izmjena koje su se u klasičnim modelima nisu mogle jednostavno izvesti.

Stoga, dodavanje novog elementa kreira samo novi identifikator kojem nije potrebno odmah dodijeliti fizikalne parametre, poziciju u hijerarhiji niti imenovati element.

Fizikalni parametri dodjeljuju se tipu, koji je ovisno o potrebama u 1:1 odnosu sa elementom (svaki element ima svoj set fizikalnih parametara) ili u 1:N odnosu (elementi koji dijele zajedničke parametre), i tada tip mora biti imenovan. Izmjenom parametara tipa mijenjaju se parametri za sve elemente kojima je dodijeljen taj tip.

Izmjenom u tablici hijerarhija mijenjaju se pripadnosti i imena svih podređenih elemenata što omogućuje laku manipulaciju i preimenovanje cijelih dijelova hijerarhije.

U CIM modelu je topologija izvedena pomoću fiktivnih čvorišnih mjesta (ConnectionNode) koje povezuje sa priključnicama elemenata. Time ukupna topologija može biti izvedena u bazi pomoću tablice fiktivnih čvorišta i tablice relacije priključnica elemenata i fiktivnih čvorišta. Posljedica takvog načina unosa jest da promjena topologije utječe samo na tablicu relacija priključnica elemenata i fiktivnih čvorišta.

Sve tablice, a time svi elementi, fizičkog modela se trebaju verzionirati bitemporalno. Pošto se na hijerarhije i pogotovo imenovanja vežu vanjski sustavi uputno je vezati hijerarhiju i imenovanja samo sa verzijom unosa (aktivacije).

Bitemporalno praćenje verzija je neophodno za financijsko praćenje imovine, a ako je model bitemporalan i ima višestruka imenovanja ništa ne priječi povezivanje Asset Management sustava sa modelom sustava. Postojanje bitemporalnog modela omogućava spajanje sa TIS i GIS sustavima za praćenje imovine.



Važno je napomenuti da u bitemporalnoj bazi nema brisanja redova (delete) niti izmjene redova (update) već samo dodavanja redova (insert) u bazi. Ukupna povijest svih promjena nad modelom je sačuvana.

## **2.5. Pogledi na bazu**

### **2.5.1. Dispečerski pogled**

Dispečerski pogled  $model_d(t)=model(t,t)$  predstavlja rekonstrukciju cjelokupnog stanja sustava kakvog ga je dispečer vidio u nekom prošlom trenutku  $t$ , tj. pogled u prošlost bez uključenih retroaktivnih promjena modela i bez ispravaka mjerenja.

To daje zahtjev na povijesnu bazu da također mora prikazivati stanje procesnih vrijednosti u tom trenutku bez uključenih ispravaka koji su nastali nakon tog vremena  $t$  (iako je potrebno imati ispravke mjerenja koje su došle do vremena  $t$ ). Također dodatni je zahtjev na povijesnu bazu da mora dohvaćati prognozirane i planirane vrijednosti koje su došle samo do vremena pogleda  $t$ .

### **2.5.2. Analitičarski pogled**

Analitičarski pogled  $model(n,t)$  predstavlja model sustava u trenutku  $t$  sa uključenim svim retroaktivnim izmjenama do trenutka  $n$ .  $t$  se može nalaziti i u budućnosti čime će se dohvatiti i izmjene koje su unesene, ali im je vrijeme važenja u budućnosti.

Zahtjev na povijesnu bazu jest da mora uključiti ispravke procesnih vrijednosti koje su unesene do trenutka  $n$ , ali i da mora dohvaćati prognozirane i planirane vrijednosti koje su stigle do trenutka  $n$ .

### **2.5.3. Pogled za izmjene**

Pogled za izmjene  $model_e(n,t,izmjene)=model(n,t)+izmjene$  predstavlja kompoziciju analitičarskog pogleda i skupa promjena koje još nisu unesene u bazu modela. Ovaj pogled omogućava korisniku da pripremi i isproba potencijalne izmjene bez da ih aktivira.

Ovaj, vrlo fleksibilan pogled, omogućava korištenje aplikacija za vođenje sustava kao off-line alata za analizu sustava. Primjeri za korištenje su brojni:

- provjera da li su izmjene na mreži korektno unesene (trenutni  $n$ , trenutni  $t$ )
- testiranje planiranih izmjena u mreži koja planiraju planeri ili susjedni operateri nad budućim stanjem mreže (trenutni  $n$ , a  $t$  u budućnosti)
- testiranje ispravaka elemenata ili svojstava u prošlosti (trenutni  $n$ , a  $t$  u prošlosti)
- provjere da li bi neka proširenja mreže mogla spriječiti probleme u mreži da su se napravila u prošlosti ( $n$  i  $t$  u prošlosti)
- testiranje proširenja mreže koje može otkloniti zagušenja u budućnosti (MLTOP, TYNDP)

## **2.6. Zahtjevi nad povijesnom bazom**

Povijesna baza mora omogućiti iste zahtjeve nad podacima kao što su i zahtjevi na model. Najvažnije jest da bilježi vrijeme kada je podatak stigao u sustav (vrijeme unosa) i kada je podatak uistinu izmjeren (vrijeme izmjere). To je važno kod pogleda dispečer gdje dispečer u nekom trenutku ima pogled samo na mjerenja koja su stigla u sustav do tog vremena, bez obzira kada je to mjerenje izmjereno (npr. ukoliko mjerenje kasni zbog zagušenih komunikacijskih kanala). Tu se razlikuje od analitičarskog pogleda kojemu su dostupni svi izmjereni podaci, čak i oni koji su naknadno došli. Povijesna baza mora omogućiti ispravke i da te ispravke imaju vrijeme kada su unesene, te da se ne gube originalne vrijednosti, te da se omoguće ispravci ispravaka.

Važna uloga povijesne baze su također i unosi prognoziranih i planiranih vrijednosti. U dispečerskom pogledu dostupni su samo prognozirani podaci koji su stigli do tog vremena, stoga povijesna baza mora spremati sve verzije planiranih i prognoziranih podataka da bi mogla rekonstruirati dispečer pogled.

## **2.7. Performanse povijesne baze**

Povijesna baza mora biti izvedena učinkovito s obzirom na ograničenja tvrdih diskova (HDD) i „flash“ memorijskih diskova (SSD) uređaja za pohranu (blokowska organizacija, seek time, brzina prijenosa).

Klasične povijesne baze implementirane nad SQL bazama (star schema, 1 redak=1 sample) su spore (reda veličine 1K sample/s) i postanu potpuno neupotrebljive u trenutku kad indeksna struktura premaši količinu RAMa. (reda veličine 100M zapisa)

Potrebno je omogućiti i brz dohvat usrednjenih trendova bez dohvata cijelih vremenskih serija i usrednjavanja prilikom svakog upita.

Radi smanjenja zahtjeva za prostorom i povećanjem performansi podatke je potrebno pohranjivati po stupcima (column-oriented) i koristiti kompresiju prilagođenu tipu vremenske serije koja se zapisuje [3].

Postoje dva osnovna zahtjeva za dohvatom podataka iz povijesne baze: dohvat snapshot-a, tj. svih mjerenja za jednu točku u vremenu (bilo dispečerski ili analitičarski pogled), te dohvat trenda pojedinog mjerenja. Za optimizaciju dohvata ta su dva zahtjeva oprečna, tj. nije moguće napraviti zapis u bazi tako da omogućava brzi dohvat i u jednom i u drugom slučaju. Stoga je uputno svaki podatak zapisati na oba načina tako da je moguć brzi dohvat svih podataka u jednom trenutku i dohvat jednog podatka za dugi period.

Baze koje implementiraju ove smjernice nazivaju se historianima ili bazama vremenskih serija i mogu biti nekoliko redova veličine učinkovitiji od jednostavnog rješenja realiziranog u klasičnoj SQL bazi.

### 3. ZAKLJUČAK

Moderni sustavi za vođenje elektroenergetskog sustava više nemaju samo jednu funkciju vođenja kao što su imali sustavi vođenja u prošlosti već moraju razmjenjivati podatke sa velikim brojem raznovidnih sustava unutar vlastite organizacije i sa susjednim operatorima sustava. Iskorištena su znanja iz financijskih sustava koji su se susreli s takvim problemima puno ranije, te je predstavljen bitemporalni model elektroenergetskog sustava, te poboljšanja povijesne baze. Predstavljena metoda za praćenje verzijamodela elektroenergetskog sustava je iskorišten za izvoz modela u CGMES EQ formatu.

Pokazalo se da je spremanje i dohvaćanje verzija modela elektroenergetskog sustava moguće upotrebom klasičnih relacijskih baza kojima je potrebno posebno pripremiti tablice za pohranu. Za spremanje i dohvaćanje svih mjerenja i podataka potrebna je posebna vrsta baze podataka nazvana povijesna baza (historian).

Moderni sustavi za vođenje moraju pokrivati sljedeće zahtjeve:

- višestruka imenovanja sa piramidalnom hijerarhijom
- vlasnička hijerarhija
- dozvole nad hijerarhijskim elementima
- mjerne uređaje kao posebne topološke elemente
- bitemporalni zapis modela u bazi
- pogled u tadašnju prošlost modela i mjerenja
- pogled u ispravljenju prošlost modela i mjerenja
- pogled u prognoziranu/planiranu budućnost modela i mjerenja
- povijesna baza izvedena kao historijan (baze vremenskih serija)
- brz dohvat trendova i snapshot-a
- omogućena razmjena samo razlika modela
- sva povijest promjena na modelu je sačuvana

### 4. LITERATURA

- [1] A. Rüping, "2D History - Versioning in the Presence of Retroactive and Future Changes," in *EuroPLog*, Irsee, Germany, 2002.
- [2] M. Kaufmann, A. A. Manjili and P. M. F. P. Vagenas, "Timeline Index: A Unified Data Structure for Processing Queries on Temporal Data in SAP HANA," in *SIGMOD*, New York, NY, USA, 2013.
- [3] EPRI, Common Information Model Primer: First Edition, Palo Alto, CA: EPRI, 2011.