

Teknisk införandedokumentation för Eltariff-API

Introduktion

Eltariff-API är en specifikation som ger elnätsbolag (DSO:er) möjlighet att publicera sina tariffer i ett standardiserat format. Detta dokument beskriver hur ni som nätbolag implementerar och använder API:et för att tillgängliggöra era nättariffer.

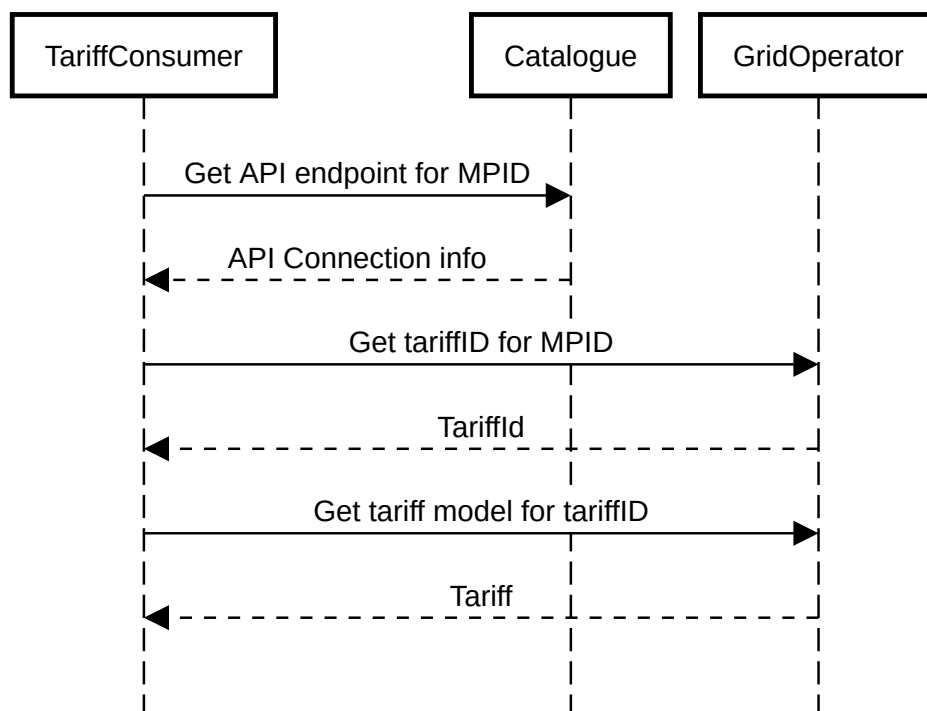
Observera: Tillhandahållandet av data i API:t är i ett utvecklingsskede. Det är sannolikt inte bindande, används på egen risk och innebär således inte några skyldigheter för elnätsbolagen om inte annat meddelas. För mer information kontakta respektive nätbolag.

Översikt av systemet

Eltariff-API bygger på följande komponenter:

1. **Grid Tariff API** - Ett standardiserat API som varje elnätsbolag implementerar för att publicera sina tariffer
2. **Katalog-API** - En central katalogtjänst där elnätsbolag registrerar information om sina tillgängliga API:er
3. **Klienter** - System som konsumerar tariff-information genom att först anropa katalog-API:et och sedan det specifika nätbolagets Grid Tariff API

Eltariff request



Implementationsguide

1. Skapa ditt Grid Tariff API

Som nätbolag behöver du implementera Grid Tariff API-specifikationen enligt [definitionerna i OpenAPI 3.1.0](#). API:et består av följande huvuddelar:

Endpoints

Endpoint	Beskrivning
<code>/info</code>	Ger grundläggande information om API-implementationen
<code>/tariffs</code>	Returnerar en samling av offentligt tillgängliga tariffer
<code>/tariffs/{id}</code>	Returnerar en specifik tariff baserat på ID
<code>/tariffs/search</code>	Söker efter tariffer baserat på sökparametrar
<code>/prices/{componentId}</code>	Returnerar priser för en priskomponent för en given tidsperiod

Datamodeller

Tarifferna struktureras enligt följande schema:

- 1. **Tariff** - Huvudobjektet som innehåller all information om en nättariff
 - Fasta komponenter (FixedPrice)
 - Energibaserade komponenter (EnergyPrice)
 - Effektbaserade komponenter (PowerPrice)
- 2. **Tidsbaserade strukturer**
 - Tidsintervall (DateTimeInterval)
 - Varaktighet (Duration)
 - Återkommande perioder (RecurringPeriod)
 - Kalendermönster (CalendarPattern)
- 3. **Priskomponenter**
 - Fast pris (FixedPriceComponent)
 - Enerpris (EnergyPriceComponent)
 - Effektpris (PowerPriceComponent)

2. Definiera dina tariffer

När du skapar dina tariffmodeller måste du följa datastrukturen som definieras i scheman. Här beskrivs de viktigaste delarna:

Tariffstruktur

En komplett tariff innehåller:

```
{
  "id": "<unikt-uuid>",
  "name": "Tariffnamn",
  "description": "Detaljerad beskrivning av tariffen",
  "product": "Produktnamn för intern användning",
  "companyName": "Ditt elnätbolag",
  "companyOrgNo": "Organisationsnummer",
  "direction": "consumption",
  "timeZone": "Europe/Stockholm",
  "validPeriod": {
    "fromIncluding": "2024-01-01T00:00:00+01:00",
    "toExcluding": "2025-01-01T00:00:00+01:00"
  },
  "billingPeriod": "P1M",
  "fixedPrice": {...},
  "energyPrice": {...},
  "powerPrice": {...}
}
```

Fasta priskomponenter

Används för avgifter som inte beror på förbrukning eller effekt:

```
"fixedPrice": {
  "id": "<unikt-uuid>",
  "name": "Fast avgift",
  "description": "Månatlig fast avgift",
  "costFunction": "sum(fixed(main))",
  "components": [
    {
      "id": "<unikt-uuid>",
      "name": "Abonnemangavgift",
      "description": "Fast avgift per månad",
      "type": "public",
      "reference": "main",
      "validPeriod": {
        "fromIncluding": "2024-01-01T00:00:00+01:00",
        "toExcluding": "2025-01-01T00:00:00+01:00"
      },
      "price": {
        "priceExVat": 200.0,
        "priceIncVat": 250.0,
        "currency": "SEK"
      },
      "pricedPeriod": "P1M"
    }
  ]
}
```

Energipriskomponenter

Används för priser relaterade till energiförbrukning:

```
"energyPrice": {
  "id": "<unik-id>",
  "name": "Energiavgift",
  "description": "Pris per kWh",
  "costFunction": "sum(energy(main))",
  "unit": "kWh",
  "components": [
    {
      "id": "<unik-id>",
      "name": "Överföringsavgift",
      "description": "Pris per överförd kWh",
      "type": "fixed",
      "reference": "main",
      "price": {
        "priceExVat": 0.25,
        "priceIncVat": 0.3125,
        "currency": "SEK"
      },
      "validPeriod": {
        "fromIncluding": "2024-01-01T00:00:00+01:00",
        "toExcluding": "2025-01-01T00:00:00+01:00"
      },
      "recurringPeriods": [...]
    }
  ]
}
```

Effektpriskomponenter

Används för priser relaterade till effektuttag:

```
"powerPrice": {
  "id": "<unik-id>",
  "name": "Effektavgift",
  "description": "Pris per kW",
  "costFunction": "sum(power(main))",
  "unit": "kW",
  "components": [
    {
      "id": "<unik-id>",
      "name": "Höglasteffekt",
      "description": "Effektavgift under höglastperiod",
      "type": "peak",
      "reference": "main",
      "price": {
        "priceExVat": 50.0,
        "priceIncVat": 62.5,
        "currency": "SEK"
      }
    }
  ]
}
```

```

    },
    "validPeriod": {
      "fromIncluding": "2024-01-01T00:00:00+01:00",
      "toExcluding": "2025-01-01T00:00:00+01:00"
    },
    "peakIdentificationSettings": {
      "peakFunction": "peak(main)",
      "peakIdentificationPeriod": "P1M",
      "peakDuration": "PT1H",
      "numberOfPeaksForAverageCalculation": 3
    },
    "recurringPeriods": [...]
  }
]
}

```

Tidsperioder

Definiera återkommande tidsperioder för tidsdifferentierade tariffer:

```

"recurringPeriods": [
  {
    "reference": "high",
    "frequency": "P1D",
    "activePeriods": [
      {
        "fromIncluding": "06:00:00",
        "toExcluding": "22:00:00",
        "calendarPatternReferences": {
          "include": ["weekday"],
          "exclude": ["holiday"]
        }
      }
    ]
  }
]
}
]

```

3. Skapa en fungerade implementering

För att implementera API:et behöver du:

1. Skapa en webbservertjänst som följer OpenAPI-specifikationen
2. Exponera de nödvändiga endpoints som beskrivs ovan
3. Implementera autentisering med JWT Bearer-token om det behövs
4. Formatera dina tariffer enligt schemadefinitionerna
5. Tillhandahålla en korrekt felhantering enligt specificerade ProblemDetails-format

Ett exempel på en implementation finns tillgänglig i projektet under `src/ExampleController`.

4. Registrera ditt API i katalogtjänsten

När du har implementerat Grid Tariff API behöver du registrera det i katalog-API:et. Detta gör du genom att anropa följande endpoint:

```
POST /tariffcatalogue/register
```

Du behöver tillhandahålla följande information:

```
{
  "id": "<unik-uuid>",
  "name": "Nätbolagets namn",
  "url": "https://api.dittbolag.se/gridtariff",
  "contactEmail": "kontakt@dittbolag.se",
  "mpidRanges": [
    {
      "mpidFormat": "SE",
      "rangeFrom": "735999000000000001",
      "rangeTo": "735999000000100000"
    }
  ]
}
```

Du kan uppdatera din registrering med PUT och ta bort den med DELETE mot relevanta endpoints. Mer information om katalog-API:et finns i Swagger-filen för katalogen under `specification/catalogueapi.json`.

Felsökning och vanliga problem

Vanliga implementationsproblem:

1. **Felaktiga datumformat** - Säkerställ att du använder ISO 8601-format för alla datum och tider
2. **Ogiltiga UUID** - Kontrollera att alla ID:n följer UUID-formatet
3. **Saknade obligatoriska fält** - Verifiera att alla obligatoriska fält definieras i dina objekt
4. **Felaktiga kostnadsfunktioner** - Granska syntax i kostnadsfunktioner för att säkerställa korrekt format

Problem vid registrering i katalogtjänsten:

1. **Överlappande MPID-intervall** - Kontrollera att dina MPID-intervall inte överlappar med befintliga registreringar
2. **Ogiltigt URL-format** - Säkerställ att API-URL:en är giltig och tillgänglig

Support och ytterligare information

För mer information om implementationen, kontakta supporten via GitHub-projektets issues eller via e-post till eddie.olsson@ri.se eller andra ansvariga för Eltariff-API-specifikationen.

API-specifikationen utvecklas kontinuerligt och du bör regelbundet kontrollera GitHub-repositoryt för uppdateringar.

Referens

- [Projekt: Datastandard nättariffer - förutsättningar för smart fordonsladdning](#)
- [GitHub-repository](#)
- [API-specifikation \(OpenAPI\)](#)
- [Katalogtjänstens API-specifikation](#)