

DOCKETED	
Docket Number:	21-OIR-03
Project Title:	2022 Load Management Rulemaking
TN #:	246597
Document Title:	Market Informed Demand Automation Server (MIDAS) Documentation Version 1_2 - Staff Instructions
Description:	The Market Informed Demand Automation Server (MIDAS) Documentation Version 1.2 is an update the existing instructions for accessing and uploading information to the MIDAS database.
Filer:	Morgan Shepherd
Organization:	California Energy Commission
Submitter Role:	Commission Staff
Submission Date:	10/17/2022 2:16:03 PM
Docketed Date:	10/17/2022



**CALIFORNIA
ENERGY COMMISSION**



**CALIFORNIA
NATURAL
RESOURCES
AGENCY**

California Energy Commission

STAFF INSTRUCTIONS

Market Informed Demand Automation Server (MIDAS) Documentation Version 1.2

**Connecting to and Interacting with the
MIDAS Database and Application
Programming Interface**

October 2022 | CEC-400-2021-009-ED2

California Energy Commission

Morgan Shepherd

David Cuffee

Stefanie Wayland

Primary Authors

Jennifer Nelson

Branch Manager

EXISTING BUILDINGS OFFICE

Michael J. Sokol

Director

EFFICIENCY DIVISION

Drew Bohan

Executive Director

DISCLAIMER

Staff members of the California Energy Commission prepared this report. As such, it does not necessarily represent the views of the Energy Commission, its employees, or the State of California. The Energy Commission, the State of California, its employees, contractors and subcontractors make no warrant, express or implied, and assume no legal liability for the information in this report; nor does any party represent that the uses of this information will not infringe upon privately owned rights. This report has not been approved or disapproved by the Energy Commission nor has the Commission passed upon the accuracy or adequacy of the information in this report.

ACKNOWLEDGEMENTS

The Market Informed Demand Automation Server (MIDAS) was designed by Karen Herter, PhD and implemented by David Cuffee in 2021. MIDAS continues to be enhanced and maintained by CEC staff through collaboration with utilities, community choice aggregators (CCAs), researchers, and technology companies. This document was developed by Morgan Shepherd with support from David Cuffee and Stefanie Wayland.

For more information and questions about MIDAS, please contact the MIDAS team for assistance at midas@energy.ca.gov between the hours of 8 a.m. and 5 p.m. on weekdays and non-holidays. Emails will be reviewed during this timeframe.

Please use the following citation for this report:

Shepherd, Morgan, David Cuffee, and Stefanie Wayland. 2022. *Market Informed Demand Automation Server (MIDAS) Documentation: Connecting to and Interacting with the MIDAS Database and Application Programming Interface*. California Energy Commission. Publication Number: CEC-400-2021-009-ED2.

TABLE OF CONTENTS

	Page
Market Informed Demand Automation Server (MIDAS) Documentation Version 1.2	i
Connecting to and Interacting with the MIDAS Database and Application Programming Interface	i
Acknowledgements	1
Table of Contents	3
List of Figures	4
1.Introduction	5
2.Database Structure.....	5
3.Using the MIDAS.....	7
Register	7
Get a Token	8
Get RIN List	8
Get Values	8
Get XML Schema	8
Post Values	8
Post Holiday Values	9
Get Lookup Table	9
Get Holiday Table	9
Get Historical RIN List.....	9
Get RIN History Data	9
4.Example Code.....	11
Python.....	11
R.....	11
C-Sharp (C#)	11
6.Acronyms	12
7.Glossary	13
APPENDIX A: MIDAS API Service Architecture	14
APPENDIX B: MIDAS Lookup Tables.....	15
APPENDIX C: Example XML Upload.....	16

LIST OF FIGURES

Page

Figure 1: Rate Identification Number 6
Figure 2: MIDAS API Service Architecture 14

1.Introduction

The California Energy Commission's (CEC) Market Informed Demand Automation Server (MIDAS) is a database of current, future, and historic time-varying rates, greenhouse gas (GHG) emissions associated with electrical generation, and California Flex Alert Signals. The database is populated by electric load serving entities (LSEs), WattTime's Self-Generation Incentive Program (SGIP) application programming interface (API), the California Independent System Operator (California ISO), and other entities that are registered with the MIDAS system.

MIDAS is accessible through a public API at <https://midasapi.energy.ca.gov> in two standard machine-readable formats: extensible markup language (XML), and JavaScript Object Notation (JSON). MIDAS querying is public and accessible to all registered users. CEC strongly encourages LSE users have programming skills and software to effectively populate and maintain rate information stored in the database. Non-LSE users however should be able to retrieve information stored in MIDAS without requiring extensive programming skills. Retrieving MIDAS-hosted data can be easily done through the code examples provided or through a user's own code. *For instructions on accessing the MIDAS database, see section 3.*

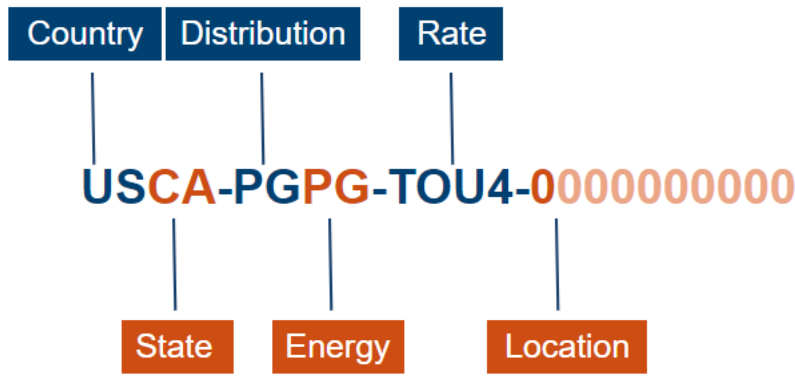
2.Database Structure

The MIDAS database supports retrieval of electric utility price schedules, California Flex Alert signals, and marginal GHG emissions from electrical generation. Flex Alerts and GHG emissions values – both forecasted and real-time - are continually retrieved from the California ISO Flex Alert site and WattTime's SGIP API respectively. GHG realtime and forecasted values are cached, or temporarily stored, within MIDAS until new values are available while Flex Alerts are passed through MIDAS from the California ISO website, as a user queries MIDAS. Previous real-time values are moved to the ValueHistory table automatically. A record of previously active Flex Alerts and historic GHG emissions are stored in the ValueHistory table.

Pursuant to the Load Management Standards, the state's largest utilities and CCAs are responsible for populating the MIDAS database Holiday Table, RateInfo Table, and the Value Table with all time-varying rate information and values offered to customers. For an example upload, please see Appendix C. To retrieve the dynamic XML schema, please see section 3.

The primary lookup identification (ID) for the MIDAS database is a compound key comprised of six individual fields that make up a standardized rate identification number (RIN) as shown in Figure 1. RINs are assigned at the time rate information is uploaded by the LSE through the MIDAS API. If an LSE is uploading to an existing RIN, the correct RIN must be used at the time of upload. Figure 1 illustrates the six identifiers that comprise a RIN: Country, State, Distribution, Energy, Rate, and Location. The location portion of the RIN may consist of 1 to 10 characters depending on the specified location's requirements.

Figure 1: Rate Identification Number (RIN)



Source: California Energy Commission

WattTime monitors GHG emissions in 11 regions across the state of California providing real-time and forecasted values for each. MIDAS includes a total of 33 different GHG RINS - real-time, forecasted, and historic. RINS with the first 12 characters USCA-SGIP-SGRT-XXXX, USCA-SGIP-SGFC-XXXX, and USCA-SGIP-SGHT-XXXX allow access to the information retrieved from queries performed on the WattTime.Org SGIP API in MIDAS. The portion of the RIN specified as "SGRT" stands for "SGIP real-time", "SGFC" stands for "SGIP forecast", and "SGHT" stands for "SGIP historic". The SGRT RIN will return only one data point that specifies the current CO2 level for the specified region. The SGFC RIN will return the forecasted CO2 levels, at 5-minute intervals, available from the WattTime API. SGHT RINS will return the historic information for each SGIP GHG emissions region. Once a real-time emissions value is replaced with a new current value, the information is moved to the ValueHistory Table where it can be retrieved through the SGHT RIN. For a list of the regions and region abbreviations please see WattTime's SGIP webpage at: <https://sgipsignal.com/grid-regions>.

MIDAS includes 3 different Flex Alert RINS - real-time, forecasted, and historic values with dates and times of previously active Flex Alerts. The California ISO offers information as to whether there is an active Flex Alert and whether one is planned. RINS of USCA-FLEX-FXRT-0000 and USCA-FLEX-FXFC-0000 check the California ISO Flex Alert web page for Flex Alerts at the moment a query is initiated through the MIDAS API. The data is passed through to the querying user directly and instantaneously. The "FXRT" portion of the first RIN stands for "Flex Alert real-time" and "FXFC" in the second RIN stands for "Flex Alert forecasted". In the RIN USCA-FLEX-FXHT-0000, "FXHT" stands for "Flex Alert historical". All previously active Flex Alerts will be reflected in the information retrieved with this RIN.

For more information on the XML schema and uploads, see section 3 and appendix C respectively.

3.Using the MIDAS

The MIDAS Representational State Transfer Application Programming Interface (RESTful API) is accessible using any programming language able to create instances of a Hypertext Transfer Protocol (HTTP) Client, HTTP Request and HTTP Response classes. Users may develop their own in-house software to connect with the MIDAS RESTful API. The requests (calls) and responses should be executed asynchronously.

The MIDAS API is comprised of five endpoints:

- Registration. Uses POST, a method to send data to create or update a resource, to create a new account.
- Token. Uses GET, a method to request data from a specific resource, to retrieve a temporary token for interacting with the MIDAS. Tokens must be passed to every API call except Registration and Token.
- Holiday. Uses POST to populate the Holiday table (LSE accounts only).
- ValueData. Uses GET to retrieve RateInfo and Value data, the XML schema, lookup tables, and RIN lists. Uses POST to upload data to the RateInfo and Value tables (LSE accounts only).
- ValueHistory. Uses GET to retrieve ValueHistory data that has been moved to the ValueHistory table after a new upload.

PUT and DELETE operations are not used in the MIDAS API.

The MIDAS REST API and Database are protected by the CEC firewall and data throttling to prevent distributed denial of service (DDoS) attacks (see Appendix A). If an error occurs after an API call, the program will send a notification for CEC IT staff to fix the issue.

The following are instructions for registering a MIDAS account and a description of the basic functions that can be used to upload and download MIDAS data. Code examples for these functions can be found in section 4.

Register

There are two types of accounts that can be used to interface with MIDAS: LSE accounts and user accounts. For security and accuracy of the system, only CEC-verified LSE accounts can upload (POST) data. User accounts can only query (GET) data from the MIDAS.

- a. User Accounts. Those wishing to retrieve data stored in the MIDAS database and not upload data should register under a user account. User account registrants may include researchers, automation service providers, technology manufacturers, etc. Registration is done through the API by making a one-time call to the MIDAS registration endpoint with the required parameters. If no errors occur, this process will send an email to the email address specified as a parameter to the call. The user must then respond to the email before they can request a token. User account holders cannot post data.
- b. LSE Accounts. New accounts for utilities and community-choice aggregators will follow the same registration process as User accounts. Following a successful registration, accounts must be verified by a CEC staff member on the MIDAS support team to allow

LSE-level access. An LSE account may only be registered under one distribution or energy company, allowing upload capabilities under only one company.

A successful registration for both User and LSE accounts will return: "User account for [your username] was successfully created. A verification email has been sent to [your email]. Please click the link in the email to start using the API."

For forgotten passwords or usernames please follow the links below:

Password: <https://midasweb.energy.ca.gov/Pages/AccountMaint/ForgotPassword>

Username: <https://midasweb.energy.ca.gov/Pages/AccountMaint/ForgotUsername>

GET a Token

After registering, make a call to the token endpoint with username and password credentials to receive a token string. This token will expire after 10 minutes. During those 10 minutes, the token can be used to call all other endpoint actions as many times as desired. In this case, a token is passed with each call made to the system to allow access to the MIDAS database.

GET RIN List

To receive a full list of all RINs in the MIDAS database, users may query the API using the GET RIN List call. This call is part of the ValueData endpoint with a parameter that identifies the signal type of the RINs being returned. The signal type parameter will return all the RINs of the requested type:

0. All
1. Electricity rates
2. Greenhouse gas emissions
3. California Independent System Operator Flex Alert

GET Values

This call allows all accounts to receive values and other information on a specified RIN. Pass a RIN with parameter RealTime to return the current value, or AllData to the ValueData endpoint to return the full schedule in either XML or JSON, as indicated in the header.

GET XML Schema

This call will allow all accounts to receive the full XML schema for the MIDAS database. This call is part of the ValueData endpoint and uses the GET verb with no parameters. It returns a string with the XML Schema Definition that the system uses to validate incoming XML Upload data.

POST Values

Populating the RateInfo and Value tables requires a call to the ValueData endpoint using the XML schema (see section 3). This function is available to LSE accounts only. Acceptable data entries are catalogued in supporting MIDAS lookup tables listed in the Appendix. Each time an LSE account holder uploads to MIDAS, the previously uploaded rates under that Distribution or Energy Company are moved to the ValueHistory table, see Archiving Data to ValueHistory

table paragraph below for more information. Please note, each RIN may only store a total of 50,000 values.

Archiving Data to ValueHistory Table

In an effort to make data management easier for LSE accounts, the MIDAS team has included a mechanism within MIDAS to move previously uploaded data to the ValueHistory table once a new upload is made or when information is no longer current. Each time an LSE account, whether a distribution or energy company, uploads a new XML file, the previously uploaded information will be moved to the ValueHistory table. For example, if an LSE user designated as a distribution company uploads a new file with values for one RIN, all previous information under the entire distribution company will be moved to the ValueHistory table including values associated with other RINs not included in the new upload.

To avoid archiving current information, please include all data to be stored in MIDAS and visible to users, including any data not to be changed from a previous upload. More specifically, if a rate requires periodic updating, i.e., a rate that changes on a daily, hourly or sub-hourly basis, include rates that do not change as frequently during each upload, i.e., TOU rates. The information moved to the ValueHistory table can then be queried through the ValueHistory endpoint.

POST Holiday Values

Populating the Holiday table requires a call to the Holiday endpoint. This function is available to LSE accounts only. Acceptable data entries are catalogued in the supporting MIDAS Holiday table.

GET Lookup Table

To receive the values stored in each lookup table, all users may use the GET Lookup table call. This call is part of the ValueData endpoint with a parameter that identifies the relevant lookup table. Possible lookup tables include Country, Daytype, Distribution, Enduse, Energy, Location, Ratetype, Sector, State, and TimeZone. The returned data will have the upload code and description for the specified lookup table.

GET Holiday Table

All accounts may query the MIDAS API for the information stored in the Holiday table. This call is part of the Holiday endpoint. There is no parameter to be specified, the requested data will be returned in either XML or JSON as specified in the header.

GET Historical RIN List

This call will retrieve all RINs with information stored in the ValueHistory table. This call is part of the ValueHistory endpoint. All users can specify the distributor company, energy company or both that they would like to receive RIN information for. All Flex Alert and GHG Emission Historical RINs are automatically returned when querying any distribution or energy company. If a user specifies a code that does not correspond to a RIN in the ValueHistory table, then only GHG and Flex Alert historic RINs will be returned.

GET RIN History Data

To receive historical information, all accounts may query MIDAS using the GET RIN History Data call. This call is part of the ValueHistory endpoint. Pass a RIN with a parameter that identifies the "startdate" and "enddate" to return historic values for the specified RIN during that timeframe in XML or JSON as specified in the header. Please note, only historic RINs may be used in this call. For example, GHG and Flex Alert RINs have separate historic RINs.

4. Example Code

This section contains code examples in Python, R, and C# to accomplish the calls above.

Python

See the most current [Python Repository](https://github.com/morganmshep/MIDAS-Python-Repository) online at

<https://github.com/morganmshep/MIDAS-Python-Repository>

R

See the most current R Repository online at

<https://github.com/morganmshep/MIDAS-R-Repository>

C-Sharp (C#)

See the most current C# Repository online at

<https://github.com/morganmshep/MIDAS-CSharp-Repository>

5. Acronyms

API: Application Programming Interface

CEC: California Energy Commission

DDoS: Distributed Denial of Service

GHG: Greenhouse Gas

GUID: Global Unique Identifier

ISO: Independent System Operator

LSE: Load Serving Entity

MIDAS: Market Informed Demand Automation Server

RESTful API: Representational State Transfer

RIN: Rate Identification Number

TOU: Time-of-Use

UTC: Coordinated Universal Time

6. Glossary

Application Programming Interface (API): Intermediary that allows two software programs to communicate with each other.

California Energy Commission (CEC): California's primary energy policy and planning agency.

Distributed Denial of Service (DDoS): Cyber-attacks aimed at making a website unavailable to actual users of the product.

Greenhouse Gases (GHG): Any gas that absorbs infrared radiation in the atmosphere.

Greenhouse gases include, but are not limited to, water vapor, carbon dioxide (CO₂), methane (CH₄), nitrous oxide (N₂O), hydrochlorofluorocarbons (HCFCs), ozone (O₃), hydrofluorocarbons (HFCs), perfluorocarbons (PFCs), and sulfur hexafluoride (SF₆). If the atmospheric concentrations of these greenhouse gases rise, the average temperature of the lower atmosphere will gradually increase.

Global Unique Identifier (GUID): Unique set of characters to identify records in the MIDAS database.

Independent System Operator (ISO): oversees the operation of California's electric power system, transmission lines, and electricity market generated and transmitted by its member utilities.

Load Serving Entity (LSE): Company that sells or provides electricity to end users in California or generates electricity at one site and consumes electricity at another site that is located in California and is owned or controlled by the company.

Market Informed Demand Automation Server (MIDAS): California's centralized, publicly accessible, rate, greenhouse gas emissions, and Flex Alert database.

Representational State Transfer (RESTful): API that follows the guidelines of REST architectural style.

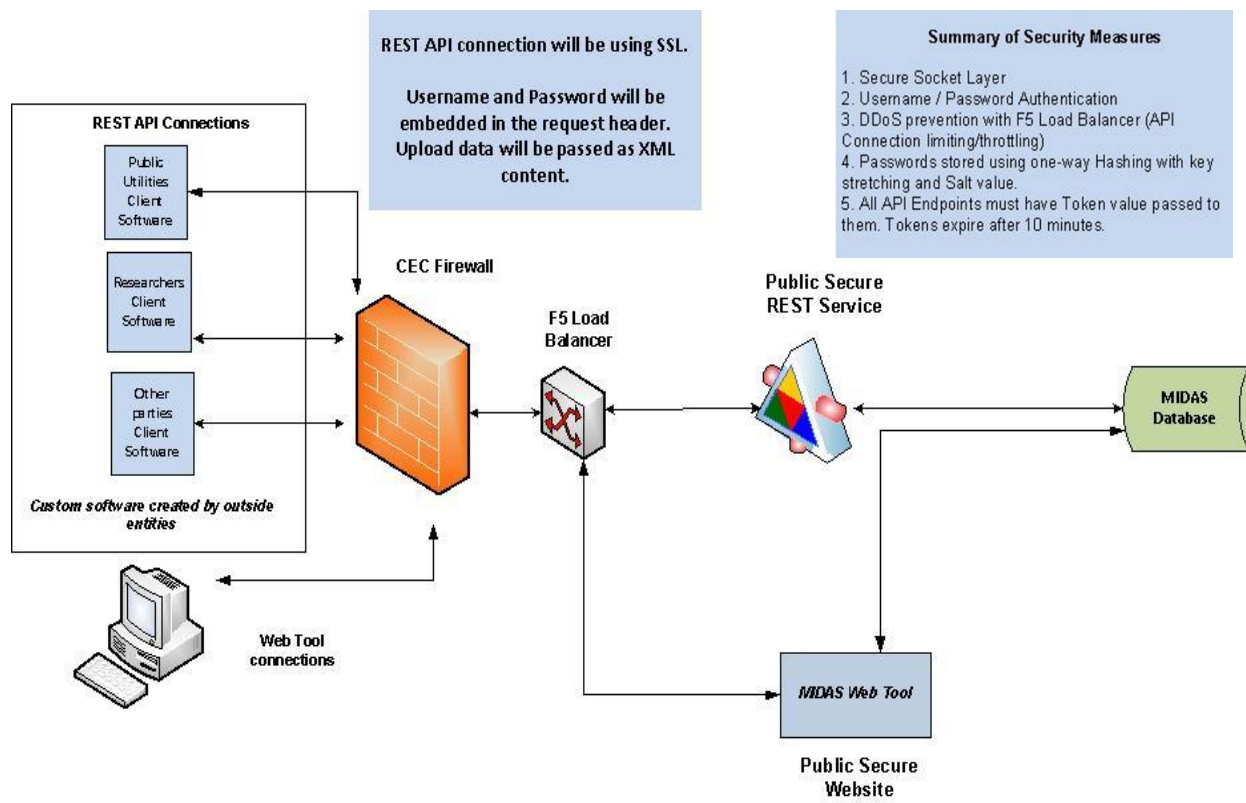
Rate Identification Number (RIN): P primary lookup ID for the MIDAS system.

Time-of-Use (TOU): Electricity prices or rates that are based on the times at which they are demanded and consumed.

Coordinated Universal Time (UTC): Primary universal standard for regulating time.

APPENDIX A: MIDAS API Service Architecture

Figure 2: MIDAS API Service Architecture



Source: California Energy Commission

APPENDIX B:

MIDAS Lookup Tables

Country Table: Contains the two-letter code for each country with the name of each country describing each code.

Day Type Table: Contains a single number Day Type Code choice with the name of each day corresponding to a number 1 through 8, where 1=Monday through 8=Holiday.

Distribution Table: Contains the two-letter code for each distribution company in California with the name of each distribution company describing each code.

Energy Table: Contains the two-letter code for each energy company in California with the name of each energy company describing each code.

Location Table: Contains the 1 to 10-character code relevant to each location with a description of each accompanying each code.

Rate Type Table: Contains the Rate Type Code, up to 10 characters, options applicable to each company's rates with a description of each rate.

Sector Table: Contains the three-letter code for the sector type relevant to each value with a description for each sector.

State/Province Table: Contains the two-letter code for each US state with the name of each included in the description.

Time Zone Table: Contains the three- to four-letter code relevant to each time zone with an accompanying description.

End Use Table: Contains the code relevant to each end use, up to 10 letters with a description describing each end use.

Unit Table: Contains the unit code relevant to each rate, emissions, or event value, up to 50 characters with a description describing each.

APPENDIX C:

Example XML Upload

Please see the example upload provided by Lawrence Berkeley National Laboratory below:

XML Input Formats for specific tags:

Tags	Format
<DateStart>,<DateEnd>	YYYY-MM-DD
<TimeStart>,<TimeEnd>	24 hour time, HH:MM or HH:MM:ss
	Where: HH = 24 Hour, 00-23
	MM = Minutes, 00-59
	ss = Seconds, 00-59

Please note: Not all, optional, fields are populated in this example. Use the current XML schema as reference for all fields.

```
<DemandData>
  <RateInformation>
    <RateID>USCA-CFCF-SHDP-0000</RateID>
    <RateName>Summer Test HDP</RateName>
    <RateType>RTP</RateType>
    <TimeZone>PST</TimeZone>
    <ValueInformation>
      <ValueData>
        <DateStart>2022-09-21</DateStart>
        <DateEnd>2022-09-21</DateEnd>
        <TimeStart>06:00:00</TimeStart>
        <TimeEnd>06:59:59</TimeEnd>
        <DayStart>1</DayStart>
        <DayEnd>8</DayEnd>
        <Value>0.149</Value>
        <ValueName>hour0</ValueName>
        <Unit>$/kWh</Unit>
      </ValueData>
```

```
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>07:00:00</TimeStart>
  <TimeEnd>07:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.086</Value>
  <ValueName>hour1</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
```

```
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>08:00:00</TimeStart>
  <TimeEnd>08:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.059</Value>
  <ValueName>hour2</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
```

```
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>09:00:00</TimeStart>
  <TimeEnd>09:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.038</Value>
  <ValueName>hour3</ValueName>
```

```
<Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>10:00:00</TimeStart>
  <TimeEnd>10:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.025</Value>
  <ValueName>hour4</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>11:00:00</TimeStart>
  <TimeEnd>11:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.016</Value>
  <ValueName>hour5</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>12:00:00</TimeStart>
  <TimeEnd>12:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
```

<Value>0.022</Value>
<ValueName>hour6</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>13:00:00</TimeStart>
<TimeEnd>13:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.036</Value>
<ValueName>hour7</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>14:00:00</TimeStart>
<TimeEnd>14:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.021</Value>
<ValueName>hour8</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>15:00:00</TimeStart>
<TimeEnd>15:59:59</TimeEnd>

```
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.011</Value>
<ValueName>hour9</ValueName>
<Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>16:00:00</TimeStart>
  <TimeEnd>16:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.019</Value>
  <ValueName>hour10</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
  <TimeStart>17:00:00</TimeStart>
  <TimeEnd>17:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.031</Value>
  <ValueName>hour11</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-21</DateStart>
  <DateEnd>2022-09-21</DateEnd>
```

<TimeStart>18:00:00</TimeStart>
<TimeEnd>18:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.061</Value>
<ValueName>hour12</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>19:00:00</TimeStart>
<TimeEnd>19:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.092</Value>
<ValueName>hour13</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>20:00:00</TimeStart>
<TimeEnd>20:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.124</Value>
<ValueName>hour14</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>

<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>21:00:00</TimeStart>
<TimeEnd>21:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.167</Value>
<ValueName>hour15</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>22:00:00</TimeStart>
<TimeEnd>22:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.215</Value>
<ValueName>hour16</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-21</DateStart>
<DateEnd>2022-09-21</DateEnd>
<TimeStart>23:00:00</TimeStart>
<TimeEnd>23:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.356</Value>
<ValueName>hour17</ValueName>
<Unit>\$/kWh</Unit>

```
</ValueData>
<ValueData>
  <DateStart>2022-09-22</DateStart>
  <DateEnd>2022-09-22</DateEnd>
  <TimeStart>00:00:00</TimeStart>
  <TimeEnd>00:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.489</Value>
  <ValueName>hour18</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-22</DateStart>
  <DateEnd>2022-09-22</DateEnd>
  <TimeStart>01:00:00</TimeStart>
  <TimeEnd>01:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.71</Value>
  <ValueName>hour19</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-09-22</DateStart>
  <DateEnd>2022-09-22</DateEnd>
  <TimeStart>02:00:00</TimeStart>
  <TimeEnd>02:59:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.788</Value>
```

<ValueName>hour20</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-22</DateStart>
<DateEnd>2022-09-22</DateEnd>
<TimeStart>03:00:00</TimeStart>
<TimeEnd>03:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.559</Value>
<ValueName>hour21</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-22</DateStart>
<DateEnd>2022-09-22</DateEnd>
<TimeStart>04:00:00</TimeStart>
<TimeEnd>04:59:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.333</Value>
<ValueName>hour22</ValueName>
<Unit>\$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-09-22</DateStart>
<DateEnd>2022-09-22</DateEnd>
<TimeStart>05:00:00</TimeStart>
<TimeEnd>05:59:59</TimeEnd>
<DayStart>1</DayStart>

```
<DayEnd>8</DayEnd>
<Value>0.215</Value>
<ValueName>hour23</ValueName>
<Unit>$/kWh</Unit>
</ValueData>
</ValueInformation>
</RateInformation>
<RateInformation>
<RateID>USCA-CFCF-STOU-0000</RateID>
<RateName>Summer Test TOU</RateName>
<RateType>TOU</RateType>
<TimeZone>PST</TimeZone>
<ValueInformation>
<ValueData>
<DateStart>2022-05-26</DateStart>
<DateEnd>2022-09-30</DateEnd>
<TimeStart>00:00</TimeStart>
<TimeEnd>03:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
<Value>0.6982</Value>
<ValueName>Peak-1</ValueName>
<Unit>$/kWh</Unit>
</ValueData>
<ValueData>
<DateStart>2022-05-26</DateStart>
<DateEnd>2022-09-30</DateEnd>
<TimeStart>04:00</TimeStart>
<TimeEnd>05:59</TimeEnd>
<DayStart>1</DayStart>
<DayEnd>8</DayEnd>
```

```
<Value>0.2637</Value>
<ValueName>Mid-Peak-2</ValueName>
<Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-05-26</DateStart>
  <DateEnd>2022-09-30</DateEnd>
  <TimeStart>06:00</TimeStart>
  <TimeEnd>20:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.031</Value>
  <ValueName>Off-Peak</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-05-26</DateStart>
  <DateEnd>2022-09-30</DateEnd>
  <TimeStart>21:00</TimeStart>
  <TimeEnd>22:59</TimeEnd>
  <DayStart>1</DayStart>
  <DayEnd>8</DayEnd>
  <Value>0.2637</Value>
  <ValueName>Mid-Peak-1</ValueName>
  <Unit>$/kWh</Unit>
</ValueData>
<ValueData>
  <DateStart>2022-05-26</DateStart>
  <DateEnd>2022-09-30</DateEnd>
  <TimeStart>23:00</TimeStart>
  <TimeEnd>23:59</TimeEnd>
```

```
<DayStart>1</DayStart>  
<DayEnd>8</DayEnd>  
<Value>0.6982</Value>  
<ValueName>Peak-2</ValueName>  
<Unit>$/kWh</Unit>  
</ValueData>  
</ValueInformation>  
</RateInformation>  
</DemandData>
```