Oracle DataRaker Standard Edition

Overview

Release 3.8.0.1.0

E70420-01

January 2016

ORACLE®

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remain at the sole discretion of Oracle.

Table of Contents

Introduction

Oracle DataRaker Standard Edition is a Software as a Service (SaaS) public cloud offering, which provides a low cost, out-of-the box solution to address the needs of:

- » Smaller utilities
- » Gas/Water/Electric utilities with daily and monthly register data. If monthly meter reads are collected, then approximately 50% of the analyzed meters need to provide daily reads.

The Oracle DataRaker Standard Edition (DataRaker SE) provides a limited set of pre-defined algorithms that are run on a regular basis (see Table 1: Algorithm Use Cases & Run Frequency). Customers are expected to regularly provide all required data, in a prescribed format, adhering to all DataRaker SE API definitions without interface development support from the Oracle Cloud Operations and Engineering team.

Customers get access to DataRaker's **Dashboards**, **Explore,** and **Export** features and to daily algorithm result reports. During the first year, Oracle DataRaker Data Scientists are available (for up to four hours per month) to answer questions and clarify the analytics that are run for the utility. Customers also have access to recorded training materials that explain how to view dashboards, search for specific meters, transformers, or other object types, create custom filters, view algorithm results, and export data.

Algorithms Supported

DataRaker SE supports two primary algorithm categories:

- » **Meter to Bill:** analyzes meter operations and billing issues
- » **Revenue Protection:** identifies energy diversion issues

In addition, seasonal analytics may be implemented to include temperature data rather than seasonal adjustment based on date.

TABLE 1: ALGORITHM USE CASES & RUN FREQUENCY

Category	Use Cases Supported	Run Frequency
Meter to Bill	» *Meter shows high consumption in comparison to itself.* » *Meter shows low consumption in comparison to itself.* » *Meter shows no consumption in comparison to itself.* » *Meter shows a spike in consumption for specific days.* » *Meter shows reverse rotation.*	Daily
Revenue Protection	» *Inactive meters that show consumption.* » *Active without consumption.* » *Year over year consumption is decreasing.* » *Meter shows no consumption, but a different meter providing commodity to the same household shows consumption.* » *Meter shows long intervals of non-consumption.*	Daily

Seasonal Analytics	» *Seasonal Usage Calculation is a DataRaker method used to calculate the seasons by looking at consumption and weather data to define the start of a season; for example, seasonal heating indicated by five days of increased consecutive consumption and corresponding decrease in temperature data. This is normally a more accurate way to define seasons than using calendar year seasons or other pre-defined data. Customer can select to use this option to define season start date.*	Yearly

Data Requirements

DataRaker SE imports customer data in text files (see Data Files on page 7 for details). The data requirements are based on the following priorities:

» **Required:** data is mandatory to support all algorithms.
» **Optional:** if information is missing, some algorithms may not run properly
» **Nice to have:** if information cannot be provided, not all dashboards will be populated and some algorithms may run less effectively.

TABLE 2: DATARAKER SE DATA ELEMENTS

#	Data Type	Description	Priority	Format	Upload File(s)
1	Meter Identifier	A meter ID comes from the physical device used to measure the amount of gas, water, or electricity used by a customer. It is normally unique by commodity type.	Required	Alphanumeric string	V3.8.0 Interface Meter File V3.8.0 Interface Event File V3.8.0 Interface Reads File
2	Register Read (Read Value)	Register read, also known as scalar read, is the read captured from the meter either through a manual read, drive by, or network solution. The register reads have to be provided with a time stamp when the read was recorded at the meter.	Required	Numeric	V3.8.0 Interface Reads File
3	Read Date	The read's time stamp (`YYYY-MM-DD hh:mm:ss UTC`).	Required	yyyy-mm-dd hh:mm:ss	V3.8.0 Interface Reads File
4	Meter Multiplier	The factor that a given register read must be multiplied in order to obtain desired units that are used for billing.	Required	Integer	V3.8.0 Interface Meter File
5	Meter Dial Count	Number of digits of register read used for billing.	Required	Integer	V3.8.0 Interface Meter File
6	Historic Register Reads	DataRaker expects 12 months of scalar data to be provided. If event data is to be used for analytics, DataRaker should receive 12 months of event data as well.	Required	Numeric	V3.8.0 Interface Reads File
7	Meter Energized Status	This status indicates whether the meter is energized or not. Options supported are: » `Active` » `Inactive` » `Disconnected` » `Removed` » `Power Off`	Required	Alphabetic string	V3.8.0 Interface Meter File

#	Data Type	Description	Priority	Format	Upload File(s)
		» `Not Defined` Meter statuses must be mapped to one of these options; abbreviations, such as A, I, *etc.*, are not accepted.			
8	Meter Status	This status indicates whether the meter is expected to show consumption or not. Options supported are: » `Active` » `Inactive` » `Not Defined` Meter statuses must be mapped to one of these options; abbreviations, such as A, I, *etc.*, are not accepted.	Required	Alphabetic string	V3.8.0 Interface Meter File
9	Meter Commodity Type	Also known as Point Type Code, The Meter Commodity Types indicates the commodity being measured. Supported types are Electric Meters, Gas Meters, and Water Meters using the following respective codes: » `EM` » `GM` » `WM` Other options are not supported and will result in an error.	Required	Alphabetic string	V3.8.0 Interface Meter File V3.8.0 Interface Event File V3.8.0 Interface Reads File
10	Service Point / Premise Identifier	This is the Premise identifier that uniquely identifies the location of the meter. If multiple service points are allowed per premise, then the service point and premise identifiers have to be concatenated to provide a unique ID. Unique identifiers comprised of any combination of premise/socket/location IDs are accepted.	Required	Alphanumeric string	V3.8.0 Interface Meter File V3.8.0 Interface Event File V3.8.0 Interface Reads File
11	Meter Installation Date	The date and time (`YYYY-MM-DD hh:mm:ss UTC`) when the meter was installed at its current location.	Required	yyyy-mm-dd	V3.8.0 Interface Meter File
12	Customer Type (Rate Class)	Customer type is a code that signifies whether the customer is Residential, Commercial, Industrial, Agricultural, or Not Applicable using the following respective codes: » `R` » `C` » `I` » `A` » `NA` Other options are not supported and will result in an error.	Required	Alphanumeric string	V3.8.0 Interface Meter File
13	Units of Measure	When uploading register reads, DataRaker also needs to know what the units of measure are for the readings uploaded. For electric meters the following UOM are supported: » `kWh` » `kWh Received` » `kVarh` » `On Peak kWh`	Required	Alphanumeric string	» *V3.8.0 Interface Meter File* » *V3.8.0 Interface Reads File*

#	Data Type	Description	Priority	Format	Upload File(s)
		» `Mid Peak kWh` » `Off Peak kWh` » `kW` » `On Peak kW` » `Mid Peak kW` » `Off Peak kW` » `Max kW` » `kVA` » `kVAr` » `Max kVA` » `Max KVAr` For water meters the following UOM are supported: » `CF` » `Liters` For electric gas the following UOM are supported: are: » `CCF` » `Cubic Meters`			
14	Meter Manufacturer	This is the company that manufactured the meters; for example, General Electric, Landis & Gyr, Itron, Siemens, *etc.* Note: Do not provide meter type such as Itron Openway or Siemens MS17TB.	Required	Alphanumeric string	V3.8.0 Interface Meter File
15	Rates	This is the classification the utility uses to define the different billing rates.	Required	Alphanumeric string	V3.8.0 Interface Meter File
16	Premise Address 1	The street address where the meter is located. Required so DataRaker can calculate the geographical location (long/lat) to display the meter in a map. Information required are: » `House Number Street Name` Example: `123 Main Street`	Required	Alphanumeric string	V3.8.0 Interface Meter File
17	Premise Postal Code	» *For the United States, the five digit Zip Code.* » *For other countries, the applicable Postal Code.*	Required	Alphanumeric	V3.8.0 Interface Meter File
18	Read Frequency	This parameter provides information regarding how often the meter is expected to be read (monthly, daily) using the following codes: » `M` » `D`	Required	Alphanumeric	V3.8.0 Interface Read File
19	Landlord Code	The landlord code provides more detailed information on who occupies the dwelling. Options supported are: » `Vacant` » `Owner Occupied` » `Tenant Occupied` » `Owner on Agreement (remodeling etc.)`	Optional	Alphanumeric string	V3.8.0 Interface Meter File

#	Data Type	Description	Priority	Format	Upload File(s)
20	Vacancy Code	Information if dwelling is occupied or not. Options supported are: » `Vacant` » `Occupied`	Optional	Alphanumeric string	V3.8.0 Interface Meter File
21	Zip+4	For the United States, the Zip Code + 4 may be used, if provided.	Optional	Alphanumeric	V3.8.0 Interface Meter File
22	Account Identifier	In some instances, service at a particular location is assigned to an account. Accounts can cover one physical or multiple physical locations	Optional	Alphanumeric string	V3.8.0 Interface Meter File
23	Meter Events	These are events that the meter records. If event information is provided, outage events from the head-end system are required. All other event information will be uploaded and displayed in the UI as well. Provide the event ID (as received from the meter) so it can be mapped to the different meter types (*e.g.*, Landis and Gyr).	Optional		V3.8.0 Interface Event File
24	Event Date	The event time stamp as recorded in the meter. (`YYYY-MM-DD hh:mm:ss UTC`).	Optional	yyyy-mm-dd hh:mm:ss	V3.8.0 Interface Event File
25	Meter Lock Status	The meter lock status will, in some cases, indicate whether the meter or service point is locked and, possibly, where it is locked. The lock status may be meter-based, location-based (*i.e.*, socket), or both. Meter lock statuses supported are: » `Locked` » `Unlocked` » `Locked at Pole` » `Not defined` Meter lock statuses must be mapped to one of these options; abbreviations, such as L, U, *etc.*, are not accepted.	Optional	Alphanumeric string	V3.8.0 Interface Meter File
26	Irregular Use Code	The irregular use codes identifies whether there is irregular usage such as a fire alarm, elevator, pool, shed, etc. This information is normally provided as a flag (1, 0, NA) for irregular use: » `1:` *Yes* » `0:` *No* » `NA:` *not applicable*	Optional	Alphanumeric string	V3.8.0 Interface Meter File
27	Seasonal Use Code	The seasonal use code indicates whether consumption is expected for only part of the year (*e.g.*, for vacation homes or customers who only have summer or winter usage). This information is normally provided as a flag (1, 0, NA) for seasonal usage: » `1:` *Yes* » `0: No` » `NA:` *not applicable*	Optional	Alphanumeric string	V3.8.0 Interface Meter File
28	Module Identifier	If meters and modules can be paired and unpaired, if module has been implemented into the meter or is connected to the meter. Provide unique module identifiers as seen on the module.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File

#	Data Type	Description	Priority	Format	Upload File(s)
29	Transformer Identifier	The ID of the transformer that the meter is connected to.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
30	Physical Socket Identifier	Identifier that the utility uses to define the physical socket location. The identifier is typically a number rather than a description of the location. (May also be referred to as the Meter Point.)	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
31	Bill Cycle	Ordinal number that corresponds to a day of the month when the billing read is obtained according to a set schedule. This is usually M-F, not including holidays, and reflects the historical workday of a meter reader.	Nice to have for Dashboard	Alphanumeric string	V3.8.0 Interface Meter File
32	Meter Location	Text description of the meter location; for example in a vault, left side of the house, basement, *etc.*	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
33	Data Collection System (Meter Type)	This is the system used to collect the register reads and events, if available. Options supported are: » **AMI:** *meters that support two way communication over a network* » **AMR:** *reads collected though a handheld system (not manual read) , or drive-by solutions* » **Manual:** *reads collected and manually entered into the system; for example, by entering the data into a form on a handheld device.*	Nice to have for Dashboard	Alphanumeric string	V3.8.0 Interface Meter File
34	Premise Address 2	This is more detailed information regarding the address (*e.g.*, **Apartment 5A** or **Suite 344**). A premise describes a location at which your company supplies some type of service.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
35	Premise City	This is the name of the city (*e.g.,* **San Francisco**). A premise describes a location at which your company supplies some type of service.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
36	Premise State	The state for US information should be provided in the common two character State Code abbreviations (*e.g.*, **CA** for California). A premise describes a location at which your company supplies some type of service.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
37	Meter/Read Route	The route that the meter is read on. Previously defined by the route the meter reader or drive-by solution took to read the meters.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
38	Account Status	Account Status, indicating whether the Account is active or not Options supported are: » **Active** » **Inactive** » **Frozen** The account statuses have to be mapped to one of these options. Abbreviations, such as A, I, *etc.,* are not accepted.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File

#	Data Type	Description	Priority	Format	Upload File(s)
39	Account Name	The name that is associated with the account.	Nice to have	Alphanumeric string	V3.8.0 Interface Meter File
40	Account Activation Date	This is the date the account became active; typically the date when a new tenant/owner moves in.	Nice to have	yyyy-mm-dd	V3.8.0 Interface Meter File

Data Files

The V3.8.0 Interface files supply data for meter information, read information (current and historical read) upload, and meter event data:

» V3.8.0 Interface Meter File
» V3.8.0 Interface Reads File
» V3.8.0 Interface Event File

Note: the Oracle DataRaker Online Documentation Library contains an archive of template files for each of the interface data file types (OracleDataRakerSE_InterfaceDataFileTemplates.zip):

  » Template_V3.8.0_InterfaceMeterFile_1.txt
  » Template_V3.8.0_InterfaceMeterFile_2.txt
  » Template_V3.8.0_Interface_Event_File.txt
  » Template_V3.8.0_InterfaceReadFile.txt

The **Reads** and **Event** files should be provided on a daily basis. The **Meter File** is expected to be provided daily for any changes that have occurred and on a weekly basis to guarantee that the two systems are always in sync.

Note: Oracle cannot guarantee that the complete set of algorithms will work properly if these files do not contain all of the required information.

The data files are pipe (|) delimited text. The order of the data elements is not important, but data rows must follow the sequence of the heading row with any missing data indicated by a pipe.

Example 1: Missing account status data

```
meter_identifier |service_point_identifier | account_status | premise_identifier

VZ518534         | 0001                    | inactive       | 922896273

VZ522713         | 0002                    | active         | 064949824

VZ254061         | 0003                    |                | 821846091
```

Example 2: Same data as in Example 1, but with a different sequence

```
meter_identifier | account_status | premise_identifier | service_point_identifier

VZ518534         | inactive       | 922896273          | 0001

VZ522713         | active         | 064949824          | 0002

VZ254061         |                | 821846091          | 0003
```

V3.8.0 Interface Meter File

Customers need to provide the V3.8.0 Meter Interface File at the start of a project implementation. It should provide all of the static information for meters, premises, and accounts that is not expected to change frequently. Once the project implementation is completed, customers must provide regular data uploads to guarantee that the two systems are in sync and that, consequently, the algorithms analyze "good" data.

The V3.8.0 Meter Interface File is provided in two primary ways:

» A full data synchronization file must be provided at the end of every month. This step will guarantee that Oracle DataRaker SE and the CIS system have the same data.

» Incremental files containing changes (*deltas*) that have occurred on a daily basis (*e.g.*, a meter replacement). The incremental file should only list the meters for which changes have occurred.

The V3.8.0 Interface Meter File can be used to upload the following information:

» Meter Identifier
» Service Point / Premise Identifier
» Physical Socket Identifier
» Account Identifier
» Meter Multiplier
» Meter Dial Count
» Meter Status
» Meter Commodity Type
» Meter Manufacturer
» Meter Location
» Data Collection System
» Meter/Read Route
» Meter Lock Status

» Irregular Use Code
» Meter Installation Date
» Transformer Identifier
» Module Identifier
» Rates
» Customer Type
» Account Status
» Account Name
» Account Activation Date
» Vacancy Code
» Landlord Code
» Bill Cycle
» Address 1
» Address 2

» Premise City
» Premise State
» Premise Postal Code
» Meter Energized Status
» Seasonal Use Code
» Premise Zip +4
» Custom Data 1 Definition / Value
» …
» Custom Data 10 Definition / Value

Meter Identifier	Service Point / Premise Identifier	Physical Socket Identifier	Account Identifier	Meter Multiplier	Meter Dial Count
meter_identifier	service_point_identifier	premise_identifier	account_identifier	meter_multiplier	meter_dial_count
VZ518534	0001	922896273	9254401387	1	5
VZ522713	0001	064949824	0643025503	1	5
NZ242142	0001	624636998	6227722201	1	5
VXZ13622	0001	042472439	0424809183	1	5
VZ458568	0002	548488562	5481077020	1	5
VZ227077	0001	300582317	3007918722	1	5
VZ135814	0001	289281838	2831777605	1	6
VZ254061	0001	821846091	8216326179	1	5
VZ219243	0001	584585588	5897286864	1	5

Meter Status	Meter Commodity Type	Meter Manufacturer	Meter Location	Data Collection System	Meter/Read Route	Meter Lock Status	Irregular Use Code
meter_status	meter_commodity_type	meter_mfg	meter_location	meter_type	meter_route	meter_lock_status	irregular_use_code
Active	EM	Itron	Vault	AMI	R1_4580	locked	0
Active	EM	Itron	East side	AMR	R1_4580	unlocked	0
Active	EM	Itron		AMI	R1_4580	locked	0
Active	EM	Siemens	In Garage	AMR	R25_98098	unlocked	0
Active	EM	GE		AMR	R13-3879	locked at pole	0
Active	EM	Itron	Underground	AMR	R13_1530	locked at pole	0
Active	EM	GE		AMI	R25_98098	locked	0
Active	EM	GE		Manual	R25_98098	locked at pole	0
Active	EM	GE		AMR	R25_98098	locked at pole	0

Meter Installation Date	Transformer Identifier	Module Identifier	Rates	Customer Type	Account Status	Account Name
meter_install_date	upstream_device_level1	module_identifier	rate_class	rate_class_type	account_status	account_name
2014-08-31	tx1	module3	R-011	R	inactive	John Dual
2014-12-25	tx3	module1	R-011	R	frozen	John Dual Jr.
2014-08-31	tx3	module2	R-011	R	inactive	John Lu
2014-08-31	tx3	module2	R-011	R	frozen	Melquiades
2014-08-31	tx2	module2	R-011	R	frozen	John Lo Jr.
2014-01-01	tx1	module1	R-011	R	inactive	Melquiades
2014-01-01	tx3	module1	R-107	R	inactive	Ursula Iguaran
2014-08-31	tx2	module1	R-011	R	inactive	Jose Buendia
2014-08-31	tx1	module2	R-011	R	active	Pietro Crespi

Account Activation Date	Vacancy Code	Landlord Code	Bill Cycle	Address 1	Address 2	Premise City	Premise State	Premise Postal Code
account_activation_date	vacancy_code	landlord_code	bill_cycle	address1	address2	city	state	postal_code
2014-01-01	Vacant	Vacant	2	1900 Oracle Way	Suite 100	Reston	VA	20191
2014-12-25	Vacant	Vacant	2	1900 Oracle Way	Suite 101	Reston	VA	20191
2014-12-25	Occupied	Owner Occupied	1	1900 Oracle Way	Suite 102	Reston	VA	20191
2014-01-01	Vacant	Tenant Occupied	2	1900 Oracle Way	Suite 103	Reston	VA	20191
2014-08-31	Occupied	Vacant	2	1900 Oracle Way	Suite 104	Reston	VA	20191
2014-01-01	Vacant	Tenant Occupied	1	1900 Oracle Way	Suite 105	Reston	VA	20191
2014-08-31	Vacant	Vacant	2	1900 Oracle Way	Suite 106	Reston	VA	20191
2014-08-31	Occupied	Owner Occupied	1	1900 Oracle Way	Suite 107	Reston	VA	20191
2014-12-25	Vacant	Tenant Occupied	3	1900 Oracle Way	Suite 108	Reston	VA	20191

Meter Energized Status	Seasonal Use Code	Premise Zip+4	Custom Data 1 Definition	Custom Date 1 Value	....	
meter_energized_status	seasonal_use_code	premise_zip+4	generic_1_def	generic_1_value		
Active	0	3768	generic_1_def_stuff3	generic_1_value_stuff2		
Disconnected	0	3768	generic_1_def_stuff3	generic_1_value_stuff3		
Active	0	3768	generic_1_def_stuff1	generic_1_value_stuff1		
Active	0	3768	generic_1_def_stuff2	generic_1_value_stuff1		
Active	1	3768	generic_1_def_stuff2	generic_1_value_stuff3		
Removed	0	3768	generic_1_def_stuff2	generic_1_value_stuff3		
Active	0	3768	generic_1_def_stuff1	generic_1_value_stuff2		
Active	0	3768	generic_1_def_stuff1	generic_1_value_stuff3		
Not Defined	0	3768	generic_1_def_stuff1	generic_1_value_stuff3		

....	....	....	....	....	....	....	Custom Data 10 Definition	Custom Date 10 Value
							generic_10_def	generic_10_value
							generic_10_def_stuff3	generic_10_value_stuff2
							generic_10_def_stuff1	generic_10_value_stuff2
							generic_10_def_stuff3	generic_10_value_stuff2
							generic_10_def_stuff3	generic_10_value_stuff1
							generic_10_def_stuff1	generic_10_value_stuff2
							generic_10_def_stuff1	generic_10_value_stuff3
							generic_10_def_stuff2	generic_10_value_stuff3
							generic_10_def_stuff1	generic_10_value_stuff2
							generic_10_def_stuff1	generic_10_value_stuff2

Legend

» Required data is highlighted in the red/pink boxes.

» Optional data is highlighted in the blue/purple boxes.

» Nice to have data is indicated by the green boxes.

See **Table 2: DataRaker SE Data Elements** to make sure that the required information has been populated.

The same file can support multiple commodity types. If customers are not able to provide the information for multiple commodities in one file, separate files for each commodity type may be provided.

If an entry contains a space, the space may be left in the entry or replaced with an underscore. For example, Silver Springs may be uploaded as `Silver Springs` or `Silver_Springs`.

V3.8.0 Interface Reads File

The V3.8.0 Interface Reads File is provided at the start of implementation to provide twelve months of historical register reads. Once the project implementation is completed, this file should be provided to Oracle DataRaker on a daily basis.

If data was missing or incorrect, new and corrected data may be provided with correct read time stamps. The supplemental/corrected information may be included in the daily upload and will be loaded to DataRaker with the other daily meter read data.

The V3.8.0 Interface Reads File can be used to upload the following information:

» Meter Identifier

» Meter Commodity Type

» Service Point / Premise Identifier

» Event Date

» Register Read (Read Value)

» Read Frequency

» Units of Measure

Meter Identifier	Meter Commodity Type	Service Point / Premise Identifier	Event Date	Read Frequency	Units of Measure	Register Read (Read Value)
meter_commodity_type	meter_identifier	service_point_identifier	read_date_time	read_type	read_uom	read_value
EM	VZ518534	P4589k	2012-01-01 23:56:42	D	KWH	51872.205
EM	VZ518534	P4589k	2012-01-02 23:51:39	D	KWH	51895.55
EM	VZ518534	P4589k	2012-01-03 23:59:43	D	KWH	51918.111
EM	VZ518534	P4589k	2012-01-04 23:56:32	D	KWH	51941.232
EM	VZ518534	P4589k	2012-01-05 23:56:20	D	KWH	51961.518
EM	VZ518534	P4589k	2012-01-06 23:56:17	D	KWH	51977.954
EM	VZ518534	P4589k	2012-01-07 23:56:13	D	KWH	51999.08
EM	VZ518534	P4589k	2012-01-08 23:56:11	D	KWH	52022.131
EM	VZ518534	P4589k	2012-01-09 23:51:05	D	KWH	52061.023
GM	Qd89dZ90	P4589k	2012-01-08 23:56:11	M	CCF	7908.16
GM	Qd89dZ90	P4589k	2012-02-09 23:51:05	M	CCF	8045.03

The same file can support multiple commodity types. If customers are not able to provide the information for multiple commodities in one file, separate files for each commodity type may be provided.

If an entry contains a space, the space may be left in the entry or replaced with an underscore. For example, EM 3790 may be uploaded as `EM 3790` or `EM_3790`.

V3.8.0 Interface Event File

The V3.8.0 Interface Event File is not required, but event information will make the analytics stronger. Ideally, Oracle would like twelve months of historical event information uploaded at the beginning of the project, if available. Once a project is in production, data should be uploaded either on a daily basis or when available. At a minimum, power outage events have to be received from the head-end system to include event information in the algorithms.

If data was missing or incorrect, new and corrected data may be provided with correct read time stamps for when the event occurred. The supplemental/corrected information may be included in the daily Interface Event File and will be loaded to DataRaker with the other daily event data.

The V3.8.0 Interface Event File can be used to upload the following information:
» Meter Identifier
» Meter Commodity Type
» Service Point / Premise Identifier
» Event Date
» Meter Events

Meter Identifier	Meter Commodity Type	Service Point / Premise Identifier	Event Date	Meter Events
meter_commodity_type	meter_identifier	service_point_identifier	event_date_time	event_code
EM	VZ518534	2345P	2013-03-06 00:00:00	Register Gap Flag
EM	VZ518534	2345P	2014-09-27 00:00:00	Register Gap Flag
EM	VZ518534	2345P	2014-10-17 00:00:00	Register Gap Flag
EM	VZ518534	2345P	2015-02-11 00:00:00	Up Tick Flag
EM	VZ518534	2345P	2014-02-22 00:00:00	L+G Power Outage Flag
EM	VZ518534	2345P	2014-09-07 00:00:00	L+G Power Outage Flag
EM	VZ522713	0012P	2012-04-04 00:00:00	Register Gap Flag
EM	VZ522713	0012P	2014-01-24 00:00:00	Register Gap Flag
EM	VZ522713	0012P	2014-01-25 00:00:00	Register Gap Flag

The same file can support multiple commodity types. If customers are not able to provide the information for multiple commodities in one file, separate files for each commodity type may be provided.

If an entry contains a space, the space may be left in the entry or replaced with an underscore. For example, Register Gap Flag may be uploaded as `Register Gap Flag` or `Register_Gap_Flag`.

Report Export

A daily report is sent to customers in a flat column delimited file format that lists algorithm results and any actions that should be taken.

The report contains the following information:

- » Product Name where DRSE stands for DataRaker Standard Edition
- » Algorithm Name (Point Groups)
- » Meter Type (Electric Meter EM, Gas Meter GM, Water Meter WM)
- » Utility ID
- » Meter ID
- » Date when the algorithm was run.
- » Algorithm Look-up, which is a DataRaker code that is typically the algorithm name with spaces replaced with underscores or may be name defined in the database.

The standard export file format lists the data, delimited with the pipe (|) character, as follows:

`DRSE|ALGORITHM_NAME|METER_TYPE|UTILITY_ID|METER_ID|ALGORITHM_DATE|ALGORITHM_LOOKUP|`

Example

- » `DRSE|0100 No Cons: High|GM|6001994514|0001288165|2015-10-11|0100_HIGH_PRIORITY|`
- » `DRSE|0100 No Cons: High|GM|6001039590|0001274456|2015-10-11|0100_HIGH_PRIORITY|`
- » `DRSE|0100 No Cons: High|GM|6001416653|0000713299|2015-10-11|0100_HIGH_PRIORITY|`

If no actions are taken to resolve the issue, or if the meter data is still within the algorithm criteria, the same Meter ID and algorithm name will appear in consecutive daily reports.

CONNECT WITH US

B blogs.oracle.com/oracle

f facebook.com/oracle

y twitter.com/oracle

o oracle.com

Integrated Cloud Applications & Platform Services

Oracle DataRaker Standard Edition Overview
Part Number E70420-01
January 2016

Oracle is committed to developing practices and products that help protect the environment