Common Lisp ODATA Client

Mariano Montone (marianomontone@gmail.com)

Table of Contents

1 Introduction

CL-ODATA-CLIENT is a Common Lisp client for accessing ODATA services.

2 Installation

3 Usage

4 CL-ODATA-CLIENT Basic Tutorial

This is CL-ODATA-CLIENT version of TripPin tutorial. See the original tutorial at: `https://www.odata.org/getting-started/basic-tutorial/`

The tutorial uses a sample service called TripPin based on OData V4. Generally speaking, TripPin provides a service that can manage people's trips. The service is designed for below purposes:

Build a service that will cover as many features for OData V4 as possible. Build a more real world service, and will use it to build an end-to-end ecosystem. Make a service reference for developers to follow when they try to build an OData service.

`https://services.odata.org/V4/TripPinServiceRW`

4.1 Introduction

The Open Data Protocol (OData) is a data access protocol built on core protocols like HTTP and commonly accepted methodologies like REST for the web. There are various kinds of libraries and tools can be used to consume OData services. But for beginners and those who want to write their own libraries, the pure HTTP requests and responses are also very important. This documentation will not cover every feature details for OData V4 services but will try to cover various typical scenarios. If you want to have a more detailed understanding, please refer to OData Documentation.

We start by defining a package:

```
(defpackage :odata.trip-pin
  (:use :cl :odata-client :odata/lang :arrows :access))
```

```
(in-package :odata.trip-pin)
```

and declaring the service urls:

```
(defparameter +trip-pin-base+ "https://services.odata.org/V4/TripPinServiceRW")
```

TripPin services uses a special url for read/write, so we get that:

```
(defparameter +trip-pin-modify+
  (multiple-value-bind (response status headers modify-url)
      (drakma:http-request +trip-pin-base+)
    (quri:uri (princ-to-string modify-url))))
```

4.2 Requesting Data

OData services support requests for data via HTTP GET requests.

4.2.1 Requesting Entity Collections

The request below returns the the collection of Person People.

```
(-> +trip-pin-modify+
    (collection "People")
    (fetch :collection))
```

```
(((:odata-id
```

```
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:user-name . "russellwhyte") (:first-name . "Russell")
(:last-name . "Whyte") (:emails "Russell@example.com" "Russell@contoso.com")█
(:address-info
 ((:address . "187 Suffolk Ln.")
  (:city (:country-region . "United States") (:name . "Boise")
   (:region . "ID"))))
(:gender . "Male") (:concurrency . 637145265160790083))
((:odata-id
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:user-name . "scottketchum") (:first-name . "Scott")
(:last-name . "Ketchum") (:emails "Scott@example.com")
(:address-info
 ((:address . "2817 Milton Dr.")
  (:city (:country-region . "United States") (:name . "Albuquerque")
   (:region . "NM"))))
(:gender . "Male") (:concurrency . 637145265160790083))
((:odata-id
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:user-name . "ronaldmundy") (:first-name . "Ronald") (:last-name . "Mundy")█
(:emails "Ronald@example.com" "Ronald@contoso.com") (:address-info)
(:gender . "Male") (:concurrency . 637145265160790083))
((:odata-id
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:user-name . "javieralfred") (:first-name . "Javier")
(:last-name . "Alfred") (:emails "Javier@example.com" "Javier@contoso.com")█
(:address-info
 ((:address . "89 Jefferson Way Suite 2")
  (:city (:country-region . "United States") (:name . "Portland")
   (:region . "WA"))))
(:gender . "Male") (:concurrency . 637145265160790083))
((:odata-id
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
```

```
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:user-name . "willieashmore") (:first-name . "Willie")
     (:last-name . "Ashmore") (:emails "Willie@example.com" "Willie@contoso.com")█
     (:address-info) (:gender . "Male") (:concurrency . 637145265160790083))█
   ((:odata-id
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:odata-etag . "W/\"08D7983E303B2043\"")
     (:odata-edit-link
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:user-name . "vincentcalabrese") (:first-name . "Vincent")
     (:last-name . "Calabrese")
     (:emails "Vincent@example.com" "Vincent@contoso.com")
     (:address-info
      ((:address . "55 Grizzly Peak Rd.")
       (:city (:country-region . "United States") (:name . "Butte")
        (:region . "MT"))))
     (:gender . "Male") (:concurrency . 637145265160790083))
   ((:odata-id
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:odata-etag . "W/\"08D7983E303B2043\"")
     (:odata-edit-link
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:user-name . "clydeguess") (:first-name . "Clyde") (:last-name . "Guess")█
     (:emails "Clyde@example.com") (:address-info) (:gender . "Male")
     (:concurrency . 637145265160790083))
   ((:odata-id
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:odata-etag . "W/\"08D7983E303B2043\"")
     (:odata-edit-link
      . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
     (:user-name . "keithpinckney") (:first-name . "Keith")
     (:last-name . "Pinckney") (:emails "Keith@example.com" "Keith@contoso.com")█
     (:address-info) (:gender . "Male") (:concurrency . 637145265160790083)))█
```

4.3 Requesting an Individual Entity by ID

The request below returns an individual entity of type Person by the given UserName
"russellwhyte"

```
(-> +trip-pin-modify+
    (collection "People")
    (id "russellwhyte")
    (fetch))


((:odata-context
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/$meta
 (:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peopl
```

```
(:odata-etag . "W/\"08D7983E303B2043\"")
(:odata-edit-link
 . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peopl
(:user-name . "russellwhyte") (:first-name . "Russell") (:last-name . "Whyte")█
(:emails "Russell@example.com" "Russell@contoso.com")
(:address-info
 ((:address . "187 Suffolk Ln.")
  (:city (:country-region . "United States") (:name . "Boise")
   (:region . "ID"))))
(:gender . "Male") (:concurrency . 637145265160790083))
```

4.4 Requesting an Individual Property

To address an entity property clients append a path segment containing property name to the URL of the entity. If the property has a complex type, properties of that value can be addressed by further property name composition. First let's take a look at how to get a simple property. The request below returns the Name property of an Airport.

```
(-> +trip-pin-modify+
    (collection "Airports")
    (id "KSFO")
    (property "Name")
    (fetch :value))
```

```
"San Francisco International Airport"
```

Then let's see how to get a property value of a complex type. The request below returns the Address of the complex type Location in an Airport.

```
(-> +trip-pin-modify+
    (collection "Airports")
    (id "KSFO")
    (property "Location")
    (property "Address")
    (fetch :value))
```

```
"South McDonnell Road, San Francisco, CA 94128"
```

4.5 Querying Data

OData supports various kinds of query options for querying data. This section will help you go through the common scenarios for these query options. System Query Option $filter

The $filter system query option allows clients to filter a collection of resources that are addressed by a request URL. The expression specified with $filter is evaluated for each resource in the collection, and only items where the expression evaluates to true are included in the response.

4.5.1 Basic predicates, built-in functions

There are several kinds of basic predicates and built-in functions for $filter, including logical operators and arithmetic operators. For more detailed information, please refer to OData V4

URL Conventions Document. The request below using $filter to get people with FirstName
"Scott".

Just use raw expressions as $filter input.

```
(-> +trip-pin-modify+
    (collection "People")
    ($filter "FirstName eq 'Scott'")
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peopl
   (:odata-etag . "W/\"08D7983E303B2043\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peopl
   (:user-name . "scottketchum") (:first-name . "Scott")
   (:last-name . "Ketchum") (:emails "Scott@example.com")
   (:address-info
    ((:address . "2817 Milton Dr.")
     (:city (:country-region . "United States") (:name . "Albuquerque")
      (:region . "NM"))))
   (:gender . "Male") (:concurrency . 637145265160790083)))
```

Or Lisp based expressions:

```
(-> +trip-pin-modify+
    (collection "People")
    ($filter '(:= "FirstName" "Scott"))
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D7983E303B2043\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:user-name . "scottketchum") (:first-name . "Scott")
   (:last-name . "Ketchum") (:emails "Scott@example.com")
   (:address-info
    ((:address . "2817 Milton Dr.")
     (:city (:country-region . "United States") (:name . "Albuquerque")
      (:region . "NM"))))
   (:gender . "Male") (:concurrency . 637145265160790083)))
```

4.5.2 Filter on Complex Type

$filter can also work on complex type. The request below returns airports with "San
Francisco" contained in its Address. And Address is property of complex type Location.

```
(-> +trip-pin-modify+
    (collection "Airports")
    ($filter "contains(Location/Address, 'San Francisco')")
```

```
        (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Airp
   (:icao-code . "KSFO") (:name . "San Francisco International Airport")
   (:iata-code . "SFO")
   (:location (:address . "South McDonnell Road, San Francisco, CA 94128")█
    (:city (:country-region . "United States") (:name . "San Francisco")
     (:region . "California"))
    (:loc (:type . "Point") (:coordinates -122.374725 37.61889)
    (:crs (:type . "name") (:properties (:name . "EPSG:4326")))))))
(-> +trip-pin-modify+
    (collection "Airports")
    ($filter '(:contains "Location/Address" "San Francisco"))
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Airp
   (:icao-code . "KSFO") (:name . "San Francisco International Airport")
   (:iata-code . "SFO")
   (:location (:address . "South McDonnell Road, San Francisco, CA 94128")█
    (:city (:country-region . "United States") (:name . "San Francisco")
     (:region . "California"))
    (:loc (:type . "Point") (:coordinates -122.374725 37.61889)
    (:crs (:type . "name") (:properties (:name . "EPSG:4326")))))))
```

4.5.3 Filter on Enum Properties

The request below returns all female People of entity type Person. The Gender is a property
of Enum type.

```
(-> +trip-pin-modify+
    (collection "People")
    ($filter `(:eq "Gender" ,+person-gender/female+))
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D7983E303B2043\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:user-name . "elainestewart") (:first-name . "Elaine")
   (:last-name . "Stewart") (:emails "Elaine@example.com" "Elaine@contoso.com")█
   (:address-info) (:gender . "Female") (:concurrency . 637145265160790083))█
```

```
((:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:odata-etag . "W/\"08D7983E303B2043\"")
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:user-name . "salliesampson") (:first-name . "Sallie")
 (:last-name . "Sampson") (:emails "Sallie@example.com" "Sallie@contoso.com")█
 (:address-info
  ((:address . "87 Polk St. Suite 5")
   (:city (:country-region . "United States") (:name . "San Francisco")
    (:region . "CA")))
  ((:address . "89 Chiaroscuro Rd.")
   (:city (:country-region . "United States") (:name . "Portland")
    (:region . "OR"))))
 (:gender . "Female") (:concurrency . 637145265160790083))
((:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:odata-etag . "W/\"08D7983E303B2043\"")
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:user-name . "jonirosales") (:first-name . "Joni") (:last-name . "Rosales")█
 (:emails "Joni@example.com" "Joni@contoso.com") (:address-info)
 (:gender . "Female") (:concurrency . 637145265160790083))
((:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:odata-etag . "W/\"08D7983E303B2043\"")
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:user-name . "georginabarlow") (:first-name . "Georgina")
 (:last-name . "Barlow")
 (:emails "Georgina@example.com" "Georgina@contoso.com") (:address-info)█
 (:gender . "Female") (:concurrency . 637145265160790083))
((:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:odata-etag . "W/\"08D7983E303B2043\"")
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:user-name . "angelhuffman") (:first-name . "Angel")
 (:last-name . "Huffman") (:emails "Angel@example.com") (:address-info)
 (:gender . "Female") (:concurrency . 637145265160790083))
((:odata-id
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:odata-etag . "W/\"08D7983E303B2043\"")
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
 (:user-name . "laurelosborn") (:first-name . "Laurel")
 (:last-name . "Osborn") (:emails "Laurel@example.com" "Laurel@contoso.com")█
```

```
    (:address-info) (:gender . "Female") (:concurrency . 637145265160790083))█
  ((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D7983E303B2043\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:user-name . "sandyosborn") (:first-name . "Sandy") (:last-name . "Osborn")█
   (:emails "Sandy@example.com" "Sandy@contoso.com") (:address-info)
   (:gender . "Female") (:concurrency . 637145265160790083))
  ((:odata-id
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D7983E303B2043\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(wgkqtxlasrgtwhbtgoqbxc1l))/TripPinServiceRW/Peop
   (:user-name . "ursulabright") (:first-name . "Ursula")
   (:last-name . "Bright") (:emails "Ursula@example.com" "Ursula@contoso.com")█
   (:address-info) (:gender . "Female") (:concurrency . 637145265160790083)))█
```

4.5.4 System Query Option $orderby

The $orderby system query option allows clients to request resources in either ascending
order using asc or descending order using desc. If asc or desc not specified, then the resources
will be ordered in ascending order. The request below orders Trips on property EndsAt in
descending order.

```
    (-> +trip-pin-modify+
        (collection "People")
        (id "scottketchum")
        (property "Trips")
        ($orderby "EndsAt" :desc)
        (fetch :collection))


(((:trip-id . 2004) (:share-id . "f94e9116-8bdd-4dac-ab61-08438d0d9a71")
  (:description . "Trip from Shanghai to Beijing") (:name . "Trip in Beijing")█
  (:budget . 11000) (:starts-at . "2014-02-01T00:00:00Z")
  (:ends-at . "2014-02-04T00:00:00Z") (:tags "Travel" "Beijing"))
 ((:trip-id . 0) (:share-id . "9d9b2fa0-efbf-490e-a5e3-bac8f7d47354")
  (:description
   . "Trip from San Francisco to New York City. Nice trip with two friends. It is a 4
  (:name . "Trip in US") (:budget . 3000) (:starts-at . "2014-01-01T00:00:00Z")█
  (:ends-at . "2014-01-04T00:00:00Z")
  (:tags "Trip in New York" "business" "sightseeing")))
```

4.5.5 System Query Option $top and $skip

The $top system query option requests the number of items in the queried collection to be
included in the result. The $skip query option requests the number of items in the queried
collection that are to be skipped and not included in the result. The request below returns
the first two people of the People entity set.

```
(-> +trip-pin-modify+
    (collection "People")
    ($top 2)
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D799E6ED7BA93B\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:user-name . "russellwhyte") (:first-name . "Russell")
   (:last-name . "Whyte") (:emails "Russell@example.com" "Russell@contoso.com")█
   (:address-info
    ((:address . "187 Suffolk Ln.")
     (:city (:country-region . "United States") (:name . "Boise")
      (:region . "ID"))))
   (:gender . "Male") (:concurrency . 637147089402046779))
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D799E6ED7BA93B\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:user-name . "scottketchum") (:first-name . "Scott")
   (:last-name . "Ketchum") (:emails "Scott@example.com")
   (:address-info
    ((:address . "2817 Milton Dr.")
     (:city (:country-region . "United States") (:name . "Albuquerque")
      (:region . "NM"))))
   (:gender . "Male") (:concurrency . 637147089402046779)))
```

The request below returns people starting with the 19th people of the entity set People

```
(-> +trip-pin-modify+
    (collection "People")
    ($skip 18)
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:odata-etag . "W/\"08D799E6ED7BA93B\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:user-name . "genevievereeves") (:first-name . "Genevieve")
   (:last-name . "Reeves")
   (:emails "Genevieve@example.com" "Genevieve@contoso.com") (:address-info)█
   (:gender . "Female") (:concurrency . 637147089402046779))
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
```

```
   (:odata-etag . "W/\"08D799E6ED7BA93B\"")
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peop
   (:user-name . "kristakemp") (:first-name . "Krista") (:last-name . "Kemp")█
   (:emails "Krista@example.com") (:address-info) (:gender . "Female")
   (:concurrency . 637147089402046779)))
```

4.5.6 System Query Option $count

The $count system query option allows clients to request a count of the matching resources
included with the resources in the response. The request below returns the total number of
people in the collection.

4.5.7 System Query Option $expand

The $expand system query option specifies the related resources to be included in line with
retrieved resources. The request below returns people with navigation property Friends of
a Person

```
   (-> +trip-pin-modify+
       (collection "People")
       (id "keithpinckney")
       ($expand "Friends")
       (fetch))


   ((:odata-context
     . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/$meta
    (:odata-id
     . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peopl
    (:odata-etag . "W/\"08D799E6ED7BA93B\"")
    (:odata-edit-link
     . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peopl
    (:user-name . "keithpinckney") (:first-name . "Keith")
    (:last-name . "Pinckney") (:emails "Keith@example.com" "Keith@contoso.com")█
    (:address-info) (:gender . "Male") (:concurrency . 637147089402046779)
    (:friends
     ((:odata-id
       . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peo
      (:odata-etag . "W/\"08D799E6ED7BA93B\"")
      (:odata-edit-link
       . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peo
      (:user-name . "clydeguess") (:first-name . "Clyde") (:last-name . "Guess")█
      (:emails "Clyde@example.com") (:address-info) (:gender . "Male")
      (:concurrency . 637147089402046779))
     ((:odata-id
       . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peo
      (:odata-etag . "W/\"08D799E6ED7BA93B\"")
      (:odata-edit-link
       . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Peo
```

```
(:user-name . "marshallgaray") (:first-name . "Marshall")
(:last-name . "Garay")
(:emails "Marshall@example.com" "Marshall@contoso.com") (:address-info)█
(:gender . "Male") (:concurrency . 637147089402046779))))
```

4.5.8 System Query Option $select

The $select system query option allows the clients to requests a limited set of properties for
each entity or complex type. The request below returns Name and IcaoCode of all Airports.

```
(-> +trip-pin-modify+
    (collection "Airports")
    ($select '("Name" "IcaoCode"))
    (fetch :collection))


((((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "San Francisco International Airport") (:icao-code . "KSFO"))
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "Los Angeles International Airport") (:icao-code . "KLAX"))
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "Shanghai Hongqiao International Airport") (:icao-code . "ZSSS"))█
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "Beijing Capital International Airport") (:icao-code . "ZBAA"))█
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "John F. Kennedy International Airport") (:icao-code . "KJFK"))█
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:name . "Rome Ciampino Airport") (:icao-code . "LIRA"))
  ((:odata-id
    . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
   (:odata-edit-link
```

```
. "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
 (:name . "Toronto Pearson International Airport") (:icao-code . "CYYZ"))█
((:odata-id
  . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/Airp
 (:name . "Sydney Airport") (:icao-code . "YSSY")))
```

4.5.9 System Query Option $search

The $search system query option restricts the result to include only those entities matching
the specified search expression. The definition of what it means to match is dependent upon
the implementation. The request below get all People who has 'Boise' in their contents.

serviceRoot/People?$search=United

```
(-> +trip-pin-modify+
    (collection "People")
    ($search "United")
    (fetch :collection))


((((:odata-id
   . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
  (:odata-etag . "W/\"08D79AA36F81DD60\"")
  (:odata-edit-link
   . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
  (:user-name . "russellwhyte") (:first-name . "Russell")
  (:last-name . "Whyte") (:emails "Russell@example.com" "Russell@contoso.com")█
  (:address-info
   ((:address . "187 Suffolk Ln.")
    (:city (:country-region . "United States") (:name . "Boise")
     (:region . "ID"))))
  (:gender . "Male") (:concurrency . 637147899037343072))
 ((:odata-id
   . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
  (:odata-etag . "W/\"08D79AA36F81DD60\"")
  (:odata-edit-link
   . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
  (:user-name . "scottketchum") (:first-name . "Scott")
  (:last-name . "Ketchum") (:emails "Scott@example.com")
  (:address-info
   ((:address . "2817 Milton Dr.")
    (:city (:country-region . "United States") (:name . "Albuquerque")
     (:region . "NM"))))
  (:gender . "Male") (:concurrency . 637147899037343072))
 ((:odata-id
   . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
  (:odata-etag . "W/\"08D79AA36F81DD60\"")
  (:odata-edit-link
```

```
        . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
       (:user-name . "javieralfred") (:first-name . "Javier")
       (:last-name . "Alfred") (:emails "Javier@example.com" "Javier@contoso.com")█
       (:address-info
        ((:address . "89 Jefferson Way Suite 2")
         (:city (:country-region . "United States") (:name . "Portland")
          (:region . "WA"))))
       (:gender . "Male") (:concurrency . 637147899037343072))
      ((:odata-id
        . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
       (:odata-etag . "W/\"08D79AA36F81DD60\"")
       (:odata-edit-link
        . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
       (:user-name . "vincentcalabrese") (:first-name . "Vincent")
       (:last-name . "Calabrese")
       (:emails "Vincent@example.com" "Vincent@contoso.com")
       (:address-info
        ((:address . "55 Grizzly Peak Rd.")
         (:city (:country-region . "United States") (:name . "Butte")
          (:region . "MT"))))
       (:gender . "Male") (:concurrency . 637147899037343072))
      ((:odata-id
        . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
       (:odata-etag . "W/\"08D79AA36F81DD60\"")
       (:odata-edit-link
        . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Peop
       (:user-name . "salliesampson") (:first-name . "Sallie")
       (:last-name . "Sampson") (:emails "Sallie@example.com" "Sallie@contoso.com")█
       (:address-info
        ((:address . "87 Polk St. Suite 5")
         (:city (:country-region . "United States") (:name . "San Francisco")
          (:region . "CA")))
        ((:address . "89 Chiaroscuro Rd.")
         (:city (:country-region . "United States") (:name . "Portland")
          (:region . "OR"))))
       (:gender . "Female") (:concurrency . 637147899037343072)))
```

4.5.10 Lambda Operators

OData defines two operators any and all that evaluate a Boolean expression on a collection.
They can work on either collection properties or collection of entities.

The request below returns People with Emails containing "ll@contoso.com". The Emails
is a collection of primitive type string.

GET serviceRoot/People?$filter=Emails/any(s:endswith(s, 'contoso.com'))

4.6 Data Modification

Updatable OData services support Create, Update and Delete operation for some or all
exposed entities.

4.6.1 Create an Entity

To create an entity in a collection, the client sends a POST request to that collection's
URL. The POST body MUST contain a single valid entity representation. The request
below creates a Person which contains complex type and collection property.

```
(defparameter *p* '(("@odata.type" . "Microsoft.OData.SampleService.Models.TripPin.Per
                     ("UserName" . "teresa")
                     ("FirstName" . "Teresa")
                     ("LastName" . "Gilbert")
                     ("Gender" . "Female")
                     ("Emails" . ("teresa@example.com" "teresa@contoso.com"))
                     ("AddressInfo" . ((("Address" . "1 Suffolk Ln.")
                                        ("City" . (("CountryRegion" . "United
                                                   ("Name" . "Boise")
                                                   ("Region" . "ID")))))))))

(princ
  (-> +trip-pin-modify+
    (collection "People")
    (post *p*)))

((odata-context
  . http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/$metad
 (odata-id
  . http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/People
 (odata-etag . W/"08D799ECFD4995C4")
 (odata-edit-link
  . http://services.odata.org/V4/(S(dbwnjvoyanwfvbktg1eu5zvg))/TripPinServiceRW/People
 (user-name . teresa) (first-name . Teresa) (last-name . Gilbert)
 (emails teresa@example.com teresa@contoso.com)
 (address-info
  ((address . 1 Suffolk Ln.)
   (city (country-region . United States) (name . Boise) (region . ID))))
 (gender . Female) (concurrency . 637147115437004228))
```

4.6.2 Remove an Entity

The request below deletes the Person with UserName 'vincentcalabrese'. DELETE service-
Root/People('vincentcalabrese')

```
(-> +trip-pin-modify+
    (collection "People")
    (id "vincentcalabrese")
```

```
   (del))


nil
```

4.6.3 Update an Entity

The OData services SHOULD support PATCH as the preferred means of updating an entity.
But also services MAY additionally support PUT. The request below update the Emails of
a person using PATCH.

```
(-> +trip-pin-modify+
   (collection "People")
   (id "russellwhyte")
   (update '(("@odata.type" . "Microsoft.OData.SampleService.Models.TripPin.Person")█
              ("Emails" . ("Russell@example.com" "Russell@contoso.com" "newRussell@cont
```

4.6.4 Relationship Operations

Relationships from one entity to another are represented as navigation properties. Link to
Related Entities

A successful POST request to a navigation property's references collection adds a re-
lationship to an existing entity. The request below adds 'vincentcalabrese' to friends of
'scottketchum'.

```
(-> +trip-pin-modify+
   (collection "People") (id "scottketchum")
   (property "Friends") ($ref)
   (link '(("@odata.context" . ,(quri:render-uri +trip-pin-modify+))
           ("@odata.id" . "People('vincentcalabrese')"))))


nil
```

4.6.5 Change a Link

A successful PUT request to a single-valued navigation property's reference resource changes
the related entity. The request below change the Airline of a Flight.

```
(-> +trip-pin-modify+
            (collection "People")
            (id "russellwhyte")
            (path "Trips(0)"
                  "PlanItems(11)"
                  "Microsoft.OData.SampleService.Models.TripPin.Flight"█
                  "Airline")
            (update-link '(("@odata.context" . ,(quri:render-uri +trip-pin-modify+))|
          ("@odata.id" . "Airlines('FM')"))))


nil
```

4.6.6 Functions and Actions

OData supports custom operations (Actions and Functions). Functions are operations exposed by an OData service that MUST return data and MUST have no observable side effects. Actions are operations exposed by an OData service that MAY have side effects when invoked. Functions and actions both MAY bound to an entity type, primitive type, complex type, or a collection.

1. Invoking Unbound Functions

   The function below returns the nearest airport with the input geography point.

   ```
   (-> +trip-pin-modify+
       (fcall 'get-nearest-airport :lat 33 :lon -118)
       (fetch))


   ((:odata-context
     . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/$m
    (:odata-id
     . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Ai
    (:odata-edit-link
     . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Ai
    (:icao-code . "KLAX") (:name . "Los Angeles International Airport")
    (:iata-code . "LAX")
    (:location (:address . "1 World Way, Los Angeles, CA, 90045")
     (:city (:country-region . "United States") (:name . "Los Angeles")
      (:region . "California"))
     (:loc (:type . "Point") (:coordinates -118.40806 33.9425)
      (:crs (:type . "name") (:properties (:name . "EPSG:4326"))))))
   ```

4.6.7 Invoking Bound Functions

The request below returns the favorite airline of a person, in TripPin service, "favorite airline" means airline which user choose most times. The function GetFavoriteAirline() is bound to Person.

```
(-> +trip-pin-modify+
    (collection "People")
    (id "russellwhyte")
    (fcall "Microsoft.OData.SampleService.Models.TripPin.GetFavoriteAirline")█
    (fetch))


((:odata-context
  . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/$meta
 (:odata-id
  . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Airli
 (:odata-edit-link
  . "http://services.odata.org/V4/(S(jreo0syd1dcfhrharzudcfgy))/TripPinServiceRW/Airli
 (:airline-code . "AA") (:name . "American Airlines"))
```

4.6.8 Invoking Unbound Actions

TripPin currently has no scenario supported for unbound actions.

4.6.9 Invoking Bound Actions

The action below shares a trip to one of his friend. In TripPin service, by "share a trip" we mean that the owner and his friend now both have the trip and the trip share the same ShareId property.

```
(-> +trip-pin-modify+
    (collection "People")
    (id "russellwhyte")
    (path "Microsoft.OData.SampleService.Models.TripPin.ShareTrip")
    (post '(("userName" . "scottketchum")
            ("tripId" . 0))))

nil
```

5 API

5.1 ODATA-CLIENT package

ODATA-CLIENT   [PACKAGE]
    Provides core functions for interacting with an ODATA service.

External definitions

Variables

ACCESS-TOKEN   [ODATA-CLIENT]
    ODATA service api token.

SERVICE-ROOT   [ODATA-CLIENT]
    ODATA service base url.

Macros

ODATA-CLIENT:WITH-SERVICE-ROOT (*base* **&body** *body*)   [Macro]

Functions

ODATA-CLIENT:ODATA-POST (*uri data* **&key** *(json-encode t)*   [Function]
        authorization)
    Make a POST request to ODATA service at *URI*.
    DATA is the *data* to be posted. It is encoded using ⟨undefined⟩ [ENCODE-JSON-TO-STRING], page ⟨undefined⟩.

ODATA-CLIENT:ODATA-GET* (*uri* **&rest** *args* **&key** *$filter $expand*)   [Function]
    Make an [ODATA-GET], page 21 request using [*SERVICE-ROOT*], page 21 as URL base.

ODATA-CLIENT:ODATA-PUT (*uri data* **&key** *(json-encode t)*   [Function]
        authorization)
    Make a PUT (update) request to ODATA service at *URI*.
    DATA is the *data* to be posted. It is encoded using ⟨undefined⟩ [ENCODE-JSON-TO-STRING], page ⟨undefined⟩.

ODATA-CLIENT:ODATA-PATCH (*uri data* **&key** *(json-encode t)*   [Function]
        authorization)
    Make a PATCH request to ODATA service at *URI*.
    DATA is the *data* to be posted. It is encoded using ⟨undefined⟩ [ENCODE-JSON-TO-STRING], page ⟨undefined⟩.

ODATA-CLIENT:ODATA-GET (*url* **&key** *$filter $expand authorization*)   [Function]
    ⟨undefined⟩ [GET], page ⟨undefined⟩ request on an ODATA service at *URL*.
    $filter is an ODATA *$filter* expression.
    $expand is an ODATA *$expand* expression.

AUTHORIZATION is the *authorization* token.

See: `http://docs.oasis-open.org/odata/odata/v4.0/errata03/os/complete/`
`part1-protocol/odata-v4.0-errata03-os-part1-protocol-complete.html#`
`_The_`$filter_System

Classes

`ODATA-CLIENT:ODATA-REQUEST-ERROR`   [Class]
  Class precedence list: `odata-request-error`, `error`, `serious-condition`,
  `condition`, `t`

5.2 ODATA/LANG package

`ODATA/LANG`   [PACKAGE]
  This package exports functions that are meant to be used with arrows syntax to
  interact with an ODATA service.

External definitions

Functions

`ODATA/LANG:FETCH` (*msgraph::url* **&optional** *type*)   [Function]

`ODATA/LANG:$FILTER` (*url exp*)   [Function]
  Add ODATA $filter parameter to *URL*.

  The $filter system query option allows clients to filter a collection of resources that
  are addressed by a request *URL*. The expression specified with $filter is evaluated for
  each resource in the collection, and only items where the expression evaluates to true
  are included in the response. Resources for which the expression evaluates to false
  or to null, or which reference properties that are unavailable due to permissions, are
  omitted from the response.

  See: ⟨undefined⟩ [ODATA-CLIENT::compile-$filter], page ⟨undefined⟩
  See: `https://www.odata.org/getting-started/basic-tutorial/#filter`

`ODATA/LANG:$VALUE` (*url*)   [Function]
  Address the raw value of a primitive property.

  Example: returns the raw value of property Name of an Airport.
  (-> +trip-pin-modify+
  (collection "Airports") (id "KSFO")
  (property "Name") ($value)

  See: `https://www.odata.org/getting-started/basic-tutorial/#propertyVal`

ODATA/LANG:PROPERTY (*url name*)   [Function]
    Access the resource property with *name NAME*.

ODATA/LANG:COLLECTION (*url name*)   [Function]
    Access the resource collection at *NAME*.

ODATA/LANG:$SKIP (*url skip*)   [Function]
    The $skip query option requests the number of items in the queried collection that
    are to be skipped and not included in the result.

    See: `https://www.odata.org/getting-started/basic-tutorial/#topskip`

ODATA/LANG:PATCH (*url data*)   [Function]
    Perform a resource [PATCH], page 23 request with *DATA* to ODATA service at *URL*.

ODATA/LANG:$TOP (*url top*)   [Function]
    The $top system query option requests the number of items in the queried collection
    to be included in the result.

    See: `https://www.odata.org/getting-started/basic-tutorial/#topskip`

ODATA/LANG:SINGLETON (*url name*)   [Function]
    Access the ODATA singleton with *name NAME* at *URL*.
    See: `https://www.odata.org/getting-started/advanced-tutorial/#`
    `querySingleton` .

ODATA/LANG:POST (*msgraph::url* **&optional** *msgraph::data*)   [Function]

ODATA/LANG:$COUNT (*url*)   [Function]
    The $count system query option allows clients to request a count of the matching
    resources included with the resources in the response.

ODATA/LANG:UPDATE-LINK (*url data*)   [Function]
    Update an already existent link.

    A successful PUT request to a single-valued navigation property's reference resource
    changes the related entity.

    Example: change the Airline of a Flight

```
(-> +trip-pin-modify+
(collection "People")
(id "russellwhyte")
(path "Trips(0)"
"PlanItems(11)"
"Microsoft.OData.SampleService.Models.TripPin.Flight"
"Airline")
(update-link '(("@odata.context" . ,(⟨undefined⟩ [quri:render-uri], page ⟨undefined⟩
+trip-pin-modify+))
("@odata.id" . "Airlines('FM')"))))
```

ODATA/LANG:LINK (*url data*)   [Function]
> Add a link to a related entity.
>
> Relationships from one entity to another are represented as navigation properties.
> A successful [POST], page 23 request to a navigation property's references collection
> adds a relationship to an existing entity.
>
> Example: add 'vincentcalabrese' to friends of 'scottketchum'
>
> (-> +trip-pin-modify+
> (collection "People") (id "scottketchum")
> (property "Friends") ($ref)
> (link '(("@odata.context" . ,(⟨undefined⟩ [quri:render-uri], page ⟨undefined⟩ +trip-
> pin-modify+))
> ("@odata.id" . "People('vincentcalabrese')"))))

ODATA/LANG:$ORDERBY (*url property* **&optional** (*order :asc*))   [Function]
> The $orderby system query option allows clients to request resources in either ascend-
> ing *order* using *asc* or descending *order* using desc. If *asc* or desc not specified, then
> the resources will be ordered in ascending *order*.

ODATA/LANG:DEL (*url*)   [Function]
> Perform a resource deletion request at ODATA service at *URL*.

ODATA/LANG:$EXPAND (*url exp*)   [Function]
> Add ODATA $expand parameter to *URL*.
>
> The $expand system query option specifies the related resources to be included in
> line with retrieved resources.
>
> *EXP* is the list of things to expand.
>
> Examples:
>
> '("asdf" "foo")) => "asdf,foo"
> '("asdf" "foo" ("Bar" "Baz")) => "asdf,foo,Bar/Baz"
>
> See: ⟨undefined⟩ [ODATA-CLIENT::COMPILE-$EXPAND], page ⟨undefined⟩ .
> See: https://www.odata.org/getting-started/basic-tutorial/#expand .

ODATA/LANG:PATH (*url* **&rest** *path*)   [Function]
> Access entity in a *PATH*.
>
> Example:
>
> (-> +trip-pin-modify+

```
(collection "People")
(id "russellwhyte")
(path "Trips(0)"
"PlanItems(11)"
"Microsoft.OData.SampleService.Models.TripPin.Flight"
"Airline"))
```

ODATA/LANG:CREATE (*url data*)   [Function]
    Perform a resource creation request with *DATA* at ODATA service at *URL*.

ODATA/LANG:$SELECT (*url exp*)   [Function]
    Adds ODATA $select parameter to *URL*.

    The $select system query option allows the clients to requests a limited set of properties for each entity.

    EXP can be either a string or a list of strings.
    Elements of *EXP* are just separated by comma.

    Examples:
    (compile-$select "name") => "foo"
    (compile-$select '("name" "surname")) => "name,surname"

    See: ⟨undefined⟩ [ODATA-CLIENT::COMPILE-$SELECT], page ⟨undefined⟩
    See: `https://www.odata.org/getting-started/basic-tutorial/#select`

ODATA/LANG:UPDATE (*url data*)   [Function]
    Perform a resource update (PUT request) with *DATA* to ODATA service at *URL*.

ODATA/LANG:ID (*url id*)   [Function]
    Get ODATA resource *id*.

ODATA/LANG:FCALL (*url name* **&rest** *args*)   [Function]
    Call the ODATA action with *name NAME* and arguments *ARGS*.

    Actions are operations exposed by an OData service that MAY have side effects when invoked. Actions MAY return data but MUST ⟨undefined⟩ [NOT], page ⟨undefined⟩ be further composed with additional path segments.

    See: `https://docs.oasis-open.org/odata/odata/v4.01/odata-v4.01-part1-protocol.html#sec_Actions`

ODATA/LANG:$SEARCH (*url exp*)   [Function]
    The $search system query option restricts the result to include only those entities matching the specified search expression.

    See: `https://www.odata.org/getting-started/basic-tutorial/#search`

`ODATA/LANG:$REF` (*url*) [Function]

> A successful [POST], page 23 request to a navigation property's references collection adds a relationship to an existing entity.

5.3 ODATA/ENTITY package

`ODATA/ENTITY` [PACKAGE]

External definitions

Macros

`ODATA/ENTITY:WITH-PROPERTIES` (*properties entity* **&body** *body*) [Macro]

Generic functions

`ODATA/ENTITY:ODATA-CONTEXT` (*sb-pcl::object*) [Generic-Function]

`ODATA/ENTITY:ODATA-NEXT-LINK` (*sb-pcl::object*) [Generic-Function]

`ODATA/ENTITY:ODATA-ID` (*sb-pcl::object*) [Generic-Function]

`ODATA/ENTITY:ENTITY-SET-ELEMENTS` (*sb-pcl::object*) [Generic-Function]

`ODATA/ENTITY:ODATA-ETAG` (*sb-pcl::object*) [Generic-Function]

`ODATA/ENTITY:ODATA-EDIT-LINK` (*sb-pcl::object*) [Generic-Function]

Functions

`ODATA/ENTITY:READ-ODATA-RESPONSE` (*data*) [Function]

> Unserializes an ODATA request response.

`ODATA/ENTITY:FIND-SCHEMA` (*schema-url*) [Function]

`ODATA/ENTITY:FETCH-SCHEMA` (*schema-url*) [Function]

`ODATA/ENTITY:GET-PROPERTY` (*entity property-name*) [Function]

Classes

`ODATA/ENTITY:ODATA-ENTITY-SET` [Class]

> And ODATA entity set.
>
> Class precedence list: `odata-entity-set, standard-object, t`

`ODATA/ENTITY:ODATA-ENTITY` [Class]

> An ODATA entity.
>
> Class precedence list: `odata-entity, standard-object, t`

6 Index

(Index is nonexistent)