

# Kiwano

a Scalable Infrastructure for Hybrid Worlds  
and Realtime Location Data Processing

## Stream API version alpha2 documentation

contact: {joaquin.keller,raluca.diaconu}@orange.com



**Kiwano** –scientific name: cucumis metuliferus– is a fruit and thus totally unrelated with the topic. But after endless discussions within the team we finally agreed on this name. Anyhow, the former name, SDDS, Scalable Dynamic Spatial Database, was neither explicit nor elegant.

## Overview

In essence, Kiwano infrastructure receives streams of location data from moving objects (users, avatars, bots, drones, ...) and sends back notifications about the neighborhood, updating the list of neighbors with their new positions and states. It also implements communication between moving objects with methods for sending data within neighbors.

The stream API is aimed at programmers to allow access to Kiwano infrastructure.

After opening a TCP/IP connection with the API, messages can be sent to and received from Kiwano.

Messages are encoded in json format and separate by a string (called the separator;-). The TCP/IP connection is maintained permanently between the client and the Kiwano infrastructure.

For more details and to gained an access to Kiwano please contact us.

## Upstream messages (from client to server)

Update position and state:

```
{
  method:"update", # update position (and other dynamic data)
  urlid:"http://hybridearth.net/u/john.smith", # this serves as an id
  # and as a mean to get more data this class of objects
  iid:"u6nMnT", # moving object id, an instance of urlid class
  # format: letters and digits
  # the combination urlid+iid is unique
  # the combination is called moid

  # above urlid and iid fields are not used on websocket version

  notifications:"on", # optional, values on/off
  # signal the API to start/stop send notifications
  # about the neighborhood of this moving object

  lng:52.5, # the new position
  lat:15.3,
  angle:40, # cardinal direction in degrees, optional
  alt:40, # altitude in meters, optional

  data:{ # optional application specific dynamic data
    <appid>: <specific data>, # many or zero fields like this

    # example
    hybridearth_v0: {
      available_for_chat:"yes",
      avatar:{
        # wavefront OBJ format in json
      }
    }
  }
}
```

Remove a moving object:

```
{
  # remove is not used on websocket version
  method:"remove", # object identified as urlid+iid leaved the system
  urlid:"http://hybridearth.net/u/john.smith",
  iid:"u6nMnT"
}
```

Get all neighbors:

```
{
  method:"neighbors", # get all neighbors for urlid+iid
  urlid:"http://hybridearth.net/u/john.smith",
  iid:"u6nMnT"
  # above urlid and iid fields are not used on websocket version
}
```

```
}
```

Send a message to all neighborhood:

```
{
  method:"msg2all",
  msgid:"w28QPu6nMnTmsAWL", # optional
                                # to avoid duplications, used for replies
                                # format: letters and digits
  urlid:"http://hybridearth.net/u/john.smith",
  iid:"u6nMnT",
  # above urlid and iid fields are not used on websocket version
  msg:"Hello you all, nice to meet you", # utf-8, optional

  data:{ # optional application specific data
    <appid>: <specific data>, # many or zero fields like this
    # example
    funchat: {
      method:"move_my_avatar",
      mvnt: {
        # avatar movement description
      }
    }
  }
}
```

Send a private message to some neighbors:

```
{
  method:"msg",
  msgid:"DAwqNIcsBUYzUKDT", # optional
  urlid:"http://hybridearth.net/u/john.smith",
  iid:"u6nMnT",
  # above urlid and iid fields are not used on websocket version
  recipients: # list of recipients
  [
    ("http://hybridearth.net/u/alice.smith","w28QPu"),
  ],
  msg:"Hi Alice, how are you ?", # utf-8

  data:{ # optional application specific data
    <appid>: <specific data>, # many or zero fields like this
  }
}
```

## Downstream messages (from server to client)

Full list of neighbors for one moving object:

```
{
  method:"neighbors",
  urlid:"http://hybridearth.net/u/john.smith",
  iid:"u6nMnT", # whom the neighbors are
  # above urlid and iid fields are not used on websocket version
  nbors: # list of neighbors
  [
    {
      urlid:"http://hybridearth.net/u/alice.smith",
      iid:"w28QPu",
      lng:52.5,
      lat:15.3,
      angle:40, # cardinal direction in degrees, optional
      alt:40, # altitude in meters, optional
      data:{ # optional
        <appid>: <specific data>, # many or zero fields like this
      }
    }
  ]
}
```

Notification of changes in the neighborhood:

```
{
  method:"updates",
  moids: # list of moving objects concerned by the updates
        # should not be empty
  [
    ("http://hybridearth.net/u/john.smith","u6nMnT"),
  ],
  # above moids field is not used on websocket version

  nbors: # list of neighbors with updated data
  [
    {
      urlid:"http://hybridearth.net/u/alice.smith",
      iid:"w28QPu",
      lng:52.5,
      lat:15.3,
      angle:40, # cardinal direction in degrees, optional
      alt:40, # altitude in meters, optional
      data: {} # optional
    }
  ]
}
```

}

Notify about moving objects leaving the neighborhood:

```
{
  method:"removes",
  moids:  # list of moving objects concerned by the removes
          # should not be empty
  [
    ("http://hybridearth.net/u/john.smith","u6nMnT"),
    ("http://hybridearth.net/u/alice.smith","w28QPu")
  ],
  # above moids field is not used on websocket version

  removes: # list of objects that are not anymore neighbors of the above
  [
    ("http://hybridearth.net/u/bob.schmidt","YzUKD3")
  ]
}
```

Send a message to some moving objects:

```
{
  method:"msg",
  msgid:"6ExIB4euG5Da825l", # optional
  replyid: "DAwqNIcsBUYzUKDT", # optional
          # to indicate this is a reply to a specific msg
  from:("http://hybridearth.net/u/john.smith","u6nMnT")

  moids:  # list of moving objects concerned by the message
          # should not be empty
  [
    ("http://hybridearth.net/u/john.smith","u6nMnT"),
    ("http://hybridearth.net/u/alice.smith","w28QPu")
  ],
  # above moids field is not used on websocket version
  msg:"Hi Alice, Hi John, how are you all ?", # utf-8

  data:{ # optional
    <appid>: <specific data>, # many or zero fields like this
  }
}
```