

Fonolo Developer API – Version 1.1 build 05

Written by Mike Pultz (mike@fonolo.com)

Created: July 30, 2008

Last Updated: December 15, 2008

JSON-RPC

The fonolo developer API uses JSON-RPC (version 1.1) POSTs, over an HTTPS connection. JSON-RPC requests are remote procedure calls, encoded using the JavaScript Object Notation format, and defined here:

<http://www.json.org>

and

<http://json-rpc.org/wiki/specification>

There are many freely available JSON and JSON-RPC libraries available, for dozens of different languages.

The fonolo developer site also includes links to several libraries, and downloadable, fully functional examples of JSON-RPC classes, in several different languages.

RCP Server

All JSON-RPC requests should be posted to:

<https://json-rpc.live.fonolo.com>

Content-Type

The Content-Type for all JSON-RPC POSTs must be set to:

application/json

Otherwise the request will not be accepted by the system.

Developer Key

Every fonolo API request must include the custom fonolo developer key header with your POST request. The custom HTTP header should be formatted as:

X-Fonolo-Auth: <developer key>

Where “<developer key>” is the developer key generated from the fonolo developer portal. This header, and valid key MUST be included in all API requests, otherwise you will receive a 401 un-authorized response to your requests.

Member Account

Every fonolo API request must include the custom member username and password headers with your POST request. The custom HTTP headers should be formatted as:

X-Fonolo-Username: <username>

X-Fonolo-Password: <password>

Where <username> is a valid fonolo member account name (an e-mail address), and <password> is the password for that member, in plain text format. These headers must be included in all API requests, otherwise you will receive a 401 un-authorized response to your requests.

Development Process

New developer accounts will be assigned a “test” developer key. This key will only permit access and calls to the target companies stored in the fonolo developer “sandbox”. This is done to ensure the reliability of the fonolo system, as well as to limit the potential for abuse on the fonolo system and target company IVR trees.

Once a developer has completed the development phase of their application, they can submit their application for review by the fonolo developer team, through the fonolo developer portal. This evaluation process ensures that the developer adheres to fonolo terms of service and privacy policies.

Once the fonolo development team has evaluated the application, the developer will be assigned a “live” key, which will give them full access to all target companies in the fonolo database.

Terminology

For the purposes of this document, and when referring to the fonolo technology, a few key terms need to be defined.

Term	Definition
Target	<p>“target”, or “target company”, refers to the company being searched for or being called.</p> <p>For example, in a call between a user John, and Air Canada, Air Canada would be considered the “target”.</p>
Client	<p>“client” refers to the user initiating the call, and will presumably be the recipient of the “client leg” of the call.</p> <p>For example, in a call between a user John, and Air Canada, John will be considered the “client”.</p>
Tree	<p>A “tree” is the entire transcribed phone system of the “target” company.</p> <p>This is formatted as a hierarchical tree structure, with the main menu of the phone system, being the “root” of the tree.</p>
Node	<p>Each point or menu option in the phone “tree” is a “node” in the tree.</p> <p>When you deep-dial to a specific place in a phone menu, you are deep-dialing to a specific “node” in a “tree”.</p>
Leg	<p>When making a call, the fonolo system connects both a “target” and a “client”; each half of this call is referred to as a “leg” of the call.</p> <p>In a normal call, there is a “target leg” and a “client leg”.</p>

Standard Objects

Data returned by the fonolo API is grouped in several standard objects, formatted using JSON.

Response Object

The response object is returned by every request, and includes the result of the requested method.

Field Name	Field Type	Description
response_code	integer	This holds the integer response code for the request. 200 is a successful response. 4xx errors are request failures. 5xx errors are internal fatal errors.
response_message	string	This holds a text response message describing the response code.

Company Object

The company object contains all the standard company information returned by the system.

Field Name	Field Type	Description
id	string	The id of this company; this can be used as an argument to the company_search method to search a company by id, or by the company_details method to get tree details about the company. This id is an MD5 key, referencing a company on the fonolo.com system.
name	string	The name of the company.
phone	string	The phone number for this company. The phone number is formatted as area code, prefix and suffix, separated by hyphens.
description	string	A short description of this company; this may be blank.
common_nodes	common nodes object	A common nodes object for this company. See "Common Nodes Object" below for more details.

Common Nodes Object

This is a key-value pair list of common nodes for this company. These are commonly accessed points in the tree, which can be offered to the user as a convenience for quick access.

Field Name	Field Type	Description
id	string	The node id of this common node; this can be used in the call_start method to initiate a “deep-dial” to this point in the phone tree. This id is an MD5 key, referencing a node on the fonolo.com system.
title	string	The text-description of this node.
type	string	The type of node this is; this can be: NODE_TYPE_IVR – a normal IVR node. NODE_TYPE_AGENT – a node that goes to an agent. NODE_TYPE_ALIAS – a node that refers to another node.

Node Object

This object contains specific details about a node. A company phone tree is made up of several node objects, in a hierarchical layout.

Field Name	Field Type	Description
id	string	The id of this node; this can be used in the call_start method to initiate a “deep-dial” to this point in the phone tree. This id is an MD5 key, referencing a node on the fonolo.com system.
title	string	The title of this node.
type	string	The type of node this is; this can be: NODE_TYPE_IVR – a normal IVR node. NODE_TYPE_AGENT – a node that goes to an agent. NODE_TYPE_ALIAS – a node that refers to another node.
status	string	The current status of this node; this can be: NODE_STATUS_ACTIVE – the node is validated and active. NODE_STATUS_ERROR – this node is currently in-error, and may not work if used in the call_start function.
depth	integer	The depth of this node in the tree. This can be useful to know how far you need to iterate to build out the whole tree. A node with depth of 0 is the root node.
menu	string	The transcription of this IVR menu node. This can be empty if the node type is NODE_TYPE_AGENT or NODE_TYPE_ALIAS. This text is formatted using standard newline characters (\n), and can be of an unlimited length.
alias	string	If the type of this node is NODE_TYPE_ALIAS, then this will be the id of the node this menu refers to. If the type is anything else, this field will still be included, but will be NULL.

Methods

check_member

Validates the given username and password against the fonolo database.

@arguments:

Provide the username and password of the member account to validate.

@returns:

Nothing.

Request:

```
{
  "version": "1.1",
  "method": "check_member",
  "params": [
    "username", "password"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "check successful.",
      "response_code": 200
    },
    "data": null
  }
}
```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object is NULL.

check_member_number

Validates that the given number belongs to the username and password provided.

@arguments:

Provide the username, password and number of the of the member number to validate.

@returns:

Nothing.

Request:

```
{
  "version": "1.1",
  "method": "check_member_number",
  "params": [
    "username", "password", "416-366-2500"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "check successful.",
      "response_code": 200
    },
    "data": null
  }
}
```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object is NULL.

company_search

Searches the fonolo database for the given company id or search string, and returns a standard company list. This function was renamed from search_companies, which is now deprecated.

@arguments:

Either a phone number, or a search string.

@returns:

A standard company list.

Request:

Using the search string "hotels":

```
{
  "version": "1.1",
  "method": "company_search",
  "params": [
    "hotels"
  ]
}
```

Or using a specific company phone number, formatted using the areacode, prefix and suffix of the phone number, separated by hyphens (-), e.g., using a PERL regex:

```
/^[0-9]{3}\-[0-9]{3}\-[0-9]{4}$/
```

```
{
  "version": "1.1",
  "method": "company_search",
  "params": [
    "888-619-8622"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "search successful",
      "response_code": 200
    },
    "data": {
      "000": {
        "name": "Porter Airlines Flight Reservations - Default Tree",
        "common_nodes": {
          "fd1b39133c5f2c749fdab78b012cae2d": {
            "title": "Reservations",
            "type": "NODE_TYPE_AGENT"
          },
          "11efc9051b9870635830b393a13e718f": {
            "title": "Baggage Inquiry",
            "type": "NODE_TYPE_IVR"
          }
        }
      },
      "id": "1f0e3dad99908345f7439f8ffabdfc4",
      "phone": "888-619-8622",
      "description": ""
    },
    "001": {
      "name": "American Airlines - Default Tree",

```

```

    "common_nodes": null,
    "id": "02e74f10e0327ad868d138f2b4fdd6f0",
    "phone": "800-433-7300",
    "description": ""
  },
  "002": {
    "name": "Delta Airlines - Default Tree",
    "common_nodes": null,
    "id": "8e296a067a37563370ded05f5a3bf3ec",
    "phone": "800-221-1212",
    "description": ""
  }
}
}
}
}
}

```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object is a list of standard company objects, keyed using a three digit incrementing number between 000 and 999, which the developer can use to sort the results by relevance. This value can be NULL if the search string does not return any matches.

company_list

Lists all the companies in the fonolo database.

@arguments:

Provide a limit (max per page), page (the current page to see) and optionally, a date to specify returning only new companies. This lets you cache the company list on the application side for faster access.

The limit value lets you specify how many companies to return per page. (defaults to 25)

The page value is the page number, starting at 0. (defaults to 0)

The optional date argument gives you the ability to incrementally update a local cache, with only new companies added to the fonolo system.

The format must be: "YYYY-MM-DD"

@returns:

A standard company list.

Request:

A list of companies newer than 2008-12-15, 25 companies per page, starting at page 0 (the first page).

```
{
  "version": "1.1",
  "method": "company_list",
  "params": [
    3,
    0,
    '2008-12-15'
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "list successful",
      "response_code": 200
    },
    "data": {
      "000": {
        "name": "Porter Airlines Flight Reservations - Default Tree",
        "common_nodes": {
          "fd1b39133c5f2c749fdab78b012cae2d": {
            "title": "Reservations",
            "type": "NODE_TYPE_AGENT"
          },
          "11efc9051b9870635830b393a13e718f": {
            "title": "Baggage Inquiry",
            "type": "NODE_TYPE_IVR"
          }
        }
      },
      "id": "1f0e3dad99908345f7439f8ffabdfc4",
      "phone": "888-619-8622",
      "description": ""
    }
  }
}
```

```

    },
    "001": {
      "name": "American Airlines - Default Tree",
      "common_nodes": null,
      "id": "02e74f10e0327ad868d138f2b4fdd6f0",
      "phone": "800-433-7300",
      "description": ""
    },
    "002": {
      "name": "Delta Airlines - Default Tree",
      "common_nodes": null,
      "id": "8e296a067a37563370ded05f5a3bf3ec",
      "phone": "800-221-1212",
      "description": ""
    }
  }
}

```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object is a list of standard company objects, keyed using a three digit incrementing number between 000 and 999, which the developer can use to sort the results by relevance. This value can be NULL if the search string does not return any matches.

company_details

Returns all company details, including the full phone tree. This method was renamed from lookup_company, which is now deprecated.

@arguments:

The company id string of a company returned from the company_search method.

@returns:

A company object, and a phone tree object.

Request:

```
{
  "version": "1.1",
  "method": "company_details",
  "params": [
    "b6d767d2f8ed5d21a44b0e5886680cb9"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "lookup company successful",
      "response_code": 200
    },
    "data": {
      "company": {
        "name": "Air Canada - Default Tree",
        "common_nodes": {
          "a13dc9a33433547183b96e7d8d637a8c": {
            "title": "Flight Info",
            "type": "NODE_TYPE_AGENT"
          },
          "efe8cc8f17ffe0c60564ac7eeebf2e54": {
            "title": "Reservations",
            "type": "NODE_TYPE_IVR"
          }
        },
        "id": "b6d767d2f8ed5d21a44b0e5886680cb9",
        "phone": "888-247-2262",
        "description": ""
      },
      "tree": {
        "00": {
          "00": {
            "01": {
              "node": {
                "status": "NODE_STATUS_ERROR",
                "menu": "Changing your ... per passenger.",
                "id": "efe8cc8f17ffe0c60564ac7eeebf2e54",
                "title": "Changes to reservations",
                "type": "NODE_TYPE_AGENT",
                "depth": "2",
                "alias": null
              }
            },
            "05": {
              "node": {
                "status": "NODE_STATUS_ACTIVE",
                "menu": "For information regarding ... at 204-941-2803.",
                "id": "fbac4b9499c1da83305b44ec262ea820",
                "title": "Ticket information",
                "type": "NODE_TYPE_IVR",

```

```

        "depth": "2",
        "alias": null
    },
    },
    "node": {
        "status": "NODE_STATUS_ACTIVE",
        "menu": "Our lowest and best ... so much more.",
        "id": "5b340c7b7e05d4ef63170fdc9394588f",
        "title": "Main menu English",
        "type": "NODE_TYPE_IVR",
        "depth": "1",
        "alias": null
    },
    "04": {
        "node": {
            "status": "NODE_STATUS_ACTIVE",
            "menu": "If your checked ... call at 1-888-689-
2247. ",
            "id": "77c8f5808ac1a147acdb18217ff574f9",
            "title": "Baggage services",
            "type": "NODE_TYPE_IVR",
            "depth": "2",
            "alias": null
        }
    },
    "02": {
        "node": {
            "status": "NODE_STATUS_ACTIVE",
            "menu": "Changing your ... per passenger.",
            "id": "5cff9f8a286e7143ac498b7613ed9b02",
            "title": "New reservations",
            "type": "NODE_TYPE_AGENT",
            "depth": "2",
            "alias": null
        }
    },
    "07": {
        "node": {
            "status": "NODE_STATUS_ACTIVE",
            "menu": "For aeroplan reward ... call 1-800-361-
5373. ",
            "id": "e5694373352b0588f7af1a23846c9d20",
            "title": "Aeroplan reward reservations",
            "type": "NODE_TYPE_IVR",
            "depth": "2",
            "alias": null
        }
    },
    "03": {
        "node": {
            "status": "NODE_STATUS_ACTIVE",
            "menu": "",
            "id": "9a457cb5d7a5b9833b717754d7225c7e",
            "title": "Aircanada.com assistance",
            "type": "NODE_TYPE_AGENT",
            "depth": "2",
            "alias": null
        }
    },
    "00": {
        "01": {
            "node": {
                "status": "NODE_STATUS_ACTIVE",
                "menu": "Changing your ... dollars per
passenger. ",
                "id": "6e927af3f19bda2d1ab7bfc0cd5a1ba5",
                "title": "Don't know flight number",
                "type": "NODE_TYPE_AGENT",
                "depth": "3",
                "alias": null
            }
        },
        "00": {
            "node": {
                "status": "NODE_STATUS_ACTIVE",
                "menu": "Please enter the ... the pound key.",
                "id": "a13dc9a33433547183b96e7d8d637a8c",
                "title": "Know flight number",
                "type": "NODE_TYPE_IVR",
            }
        }
    }
}

```


call_start

Initiates a deep-dial to a specific location in a phone tree

@arguments:

Provide the node id to deep dial to, and the phone number, formatted as area code, prefix, suffix, separated by hyphens, or a SIP address to call back.

This must be a valid phone number or SIP address, previously added to the member account (defined by the X-Fonolo-Username) via the fonolo web portal, otherwise you will receive a 401 un-authorized error.

@returns:

Returns a session id of the call session.

Request:

```
{
  "version": "1.1",
  "method": "call_start",
  "params": [
    "176899be3aada6b2493e1226448ed2e3",
    "416-366-2500"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "call started successfully.",
      "response_code": 200
    },
    "data": {
      "session_id": "4f1b34e56bf3163571bcfee036439e7"
    }
  }
}
```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object holds details returned by this method.
session_id	string	The session id of the initiated call.
		This can be passed to the call_status and call_cancel methods to get the status on, or cancel the referenced call.

call_status

Returns status details about an existing call, initiated from a call_start command.

@arguments:

Provide the session id returned from the call_start command.

@returns:

Returns the details about the call session.

Request:

```
{
  "version": "1.1",
  "method": "call_status",
  "params": [
    "4f1b34e56bf3163571bcfee036439e7"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "call status successful",
      "response_code": 200
    },
    "data": {
      "client_status_code": "SIP_STATUS_RINGING",
      "target_status_code": "SIP_STATUS_CONNECTED",
      "target_status_message": "call connected.",
      "client_status_message": "ringing client number"
    }
  }
}
```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object holds details returned by this method.
client_status_code	string	<p>The current status of the client leg of the call. The “client” leg, is the call to the end-user.</p> <p>Possible values are:</p> <p>SIP_STATUS_INIT – the call object has be initialized.</p> <p>SIP_STATUS_CONNECTING – the call objecting is connecting.</p> <p>SIP_STATUS_RINGING – the call object is ringing.</p> <p>SIP_STATUS_CONNECTED – the call object is connected.</p> <p>SIP_STATUS_DISCONNECTED - the call object has disconnected (the phone was hung-up).</p> <p>SIP_STATUS_TIMEOUT – the call object timed-out while attempting to dial.</p> <p>SIP_STATUS_FAILED – the call object failed. This could be due to insufficient resources, or a failure in transmission.</p> <p>SIP_STATUS_CANCELLED – the call progress was cancelled by the system or by the user.</p> <p>SIP_STATUS_BUSY – the remote end of the call returned busy (ie- a busy signal).</p>
client_status_message	string	A text message describing the call status.
target_status_code	string	<p>The current status of the target leg of the call. The “target” leg, is the call to the company.</p> <p>See client_status_code above for all the possible status values.</p>
target_status_message	string	A text message describing the call status.

call_cancel

Cancels an existing call, initiated from a call_start command.

@arguments:

Provide the session id returned from the call_start command.

@returns:

Nothing.

Request:

```
{
  "version": "1.1",
  "method": "call_cancel ",
  "params": [
    "4f1b34e56bf3163571bcfee036439e7"
  ]
}
```

Response:

```
{
  "version": "1.1",
  "result": {
    "head": {
      "response_message": "call_cancelled successfully.",
      "response_code": 200
    },
    "data": null
  }
}
```

The response fields are outlined as follows:

Field Name	Field Type	Description
head	response object	A standard response object with the result of this request.
data	object	This object is NULL.