



Version 6.0
December 2010

Calendar API

Hipergate calendar may be accessed from third party applications by using three different client APIs.

Calendar model

The basic work unit of the calendar is the meeting which is represented by class `Hipergate.CalendarMeeting` at Microsoft .NET client API and by class `com.knowgate.addrbook.client.CalendarMeeting` at the Pure Java client.

Meetings are identified by a string which follows iCalendar standard recommendation for unique keys.

Each meeting has an organizer, a title, a start date, an end date and an optional extended description.

Meetings may also have attendants and resources.

The attendants are uniquely identified by their emails.

Resources must have a unique name for each calendar.

Security model

Access to a calendar is granted by checking an email and password against `k_users` table. In hipergate there is a one to one relationship between calendars and application users.

The access to the calendar is done by verifying the user's email and his password. For accessing a calendar, his owner must have been previously created at hipergate. There is a one to one relationship between users and calendars so that the email and password of a user uniquely identified a single calendar.

Authentication process

The calendar service does not maintain server side states. The authentication process requests a security token upon its first call by passing the user email and password. This token is then used at each next call to the calendar service on the same session.

Installing then calendar service at Server side

The calendar service is implemented at servlet
`com.knowgate.http.HttpCalendarServlet`

This servlet may be activated by adding the following lines to /WEB-INF/web.xml file

```
<servlet>
  <servlet-name>HttpCalendarServlet</servlet-name>
  <servlet-class>
    com.knowgate.http.HttpCalendarServlet
  </servlet-class>
  <init-param>
    <param-name>profile</param-name>
    <param-value>hipergate</param-value>
  </init-param>
</servlet>
<servlet-mapping>
  <servlet-name>HttpCalendarServlet</servlet-name>
  <url-pattern>/servlet/HttpCalendarServlet</url-pattern>
</servlet-mapping>
```

Data types

The data types recognized by calendar service function calls are:

id: Maximum 50 characters.

gu: Maximum 32 characters.

type: Maximum 16 characters.

name: Maximum 100 characters.

surname: Maximum 100 characters.

title: Maximum 100 characters.

active: Boolean 0 or 1.

privacy: Boolean 0 or 1.

email: Maximum 100 characters of lowercase letters only. It must match the regular expression: `[\w\x2E_-]+@[\w\x2E_-]+\x2E\D{2,4}`

description: Maximum 254 characters.

comments: Maximum 254 characters.

startdate: Date and time. For the input format for HTTP GET or POST parameters is yyyyMMddHHmmss The XML output format is yyyy-MM-ddTHH:mm:ss

enddate: Same format as startdate

timezone: 6 characters with format `[+|-]hh:mm`

REST API

The access through HTTP GET supports the following commands:

connect

Get a security token for a given email and password.

Because each calendar is associated with a single user, the logged user will be the owner of all meetings created during the session established by calling the connect command.

Request

```
http://servidor/hipergate/servlet/HttpCalendarServlet?command=connect&user=johns@yoourmail.com&password=xxxxxxx
```

Response XML (success) with the security token at tag <value>

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="connect" code="0">
  <error></error>
  <value>pur74y7abpckbpq4h8twxkv94rf8fhjebuyvb8vj</value>
</calendarresponse>
```

Response XML (error)

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="connect" code="-1">
  <error>User not found</error>
  <value></value>
</calendarresponse>
```

disconnect

Close session and discard the active security token.

Request

```
http://servidor/hipergate/servlet/HttpCalendarServlet?command=disconnect&token=xxxxxxx
```

Response XML

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="disconnect" code="0">
  <error></error>
  <value>true</value>
</calendarresponse>
```

getMeetings

Get a list of all meetings at calendar between two dates.

Request

```
http://servidor/hipergate/servlet/HttpCalendarServlet?command=
getMeetings&token=xxxxxxx&startdate=19900101000000&enddate=
20200101000000&type=meeting
```

The type is optional.

The format for start and end dates must be yyyyMMddHHmmss

Response XML

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="getMeetings" code="0">
  <error></error>
  <value>2</value>
  <meetings count="2">
    <meeting type="meeting">
      <id>c0a810a212c400a563e100004dead712@hipergate.org</id>
      <gu>c0a810a212c400a563e100004dead712</gu>
      <startdate>2010-11-08T09:00:00</startdate>
      <enddate>2010-11-08T13:00:00</enddate>
      <privacy>false</privacy>
      <title>Something to talk about</title>
      <description></description>
      <rooms count="0"></rooms>
      <attendants count="1">
        <attendant>
          <id>1232</id>
          <gu>x0a8w0a212c400a563e100004orgv335</gu>
          <name>John</name>
          <surname>Smith</surname>
          <email>Johns@yourmail.com</email>
          <timezone>+01:00</timezone>
        </attendant>
      </attendants>
    </meeting>
    <meeting type="meeting">
      <id>c0a810a212c4005fa47100003ea5d84b@hipergate.org</id>
      <gu>c0a810a212c4005fa47100003ea5d84b</gu>
      <startdate>2010-11-11T09:00:00</startdate>
      <enddate>2010-11-11:15:00:00</enddate>
      <privacy>0</privacy>
      <title>Wednesday at three o'clock</title>
      <description></description>
      <organizer>
        <id>1232</id>
        <gu>x0a8w0a212c400a563e100004orgv335</gu>
        <name>John</name>
        <surname>Smith</surname>
        <email>Johns@yourmail.com</email>
        <timezone>+01:00</timezone>
      </organizer>
    </meeting>
  </meetings>
</calendarresponse>
```

```

<rooms count="0"></rooms>
<attendants count="2">
  <attendant>
    <id>1232</id>
    <gu>x0a8w0a212c400a563e100004orgv335</gu>
    <name>John</name>
    <surname>Smith</surname>
    <email>Johns@yourmail.com</email>
    <timezone>+01:00</timezone>
  </attendant>
  <attendant>
    <id>241</id>
    <gu>e0a8w0a212c400a563e100004orgv887</gu>
    <name>Paul</name>
    <surname>Brown</surname>
    <email>Paulb@yourmail.com</email>
    <timezone>+00:00</timezone>
  </attendant>
</attendants>
</meeting>
</meetings>
</calendarresponse>

```

getMeetingsForRoom

Get list of meetings that use a given resource between two dates.

Request

```

http://servidor/hipergate/servlet/HttpCalendarServlet?command=
getMeetings&token=xxxxxxx&startdate=19900101000000&enddate=
20200101000000&room=EINSTEIN

```

getMeeting

Get all the details about a meeting identified by its iCalendar Id..

Request

```

http://servidor/hipergate/servlet/HttpCalendarServlet?command=
getMeetings&token=xxxxxxx&meeting=icalendar_id_of_meeting@hiper
gate.org

```

Respuesta XML

```

<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="getMeeting" code="0">
  <error></error>
  <value>true</value>
  <meetings count="1">
    <meeting type="meeting">
      <id>c0a810a212c5a74a926100000a9716f3@hipergate.org</id>
      <gu>c0a810a212c5a74a926100000a9716f3</gu>
      <startdate>2010-11-17T09:00:00</startdate>

```

```

<enddate>2010-11-17T15:00:00</enddate>
<privacy>0</privacy>
<title>Encounter X</title>
<description>Description of encounter X</description>
<organizer>
  <id>1232</id>
  <gu>x0a8w0a212c400a563e100004orgv335</gu>
  <name>John</name>
  <surname>Smith</surname>
  <email>Johns@yourmail.com</email>
  <timezone>+01:00</timezone>
</organizer>
<rooms count="1">
  <room type="CLASSROOM" active="1">
    <name>EINSTEIN</name>
    <comments></comments>
  </room>
</rooms>
<attendants count="2">
  <attendant>
    <id>1232</id>
    <gu>x0a8w0a212c400a563e100004orgv335</gu>
    <name>John</name>
    <surname>Smith</surname>
    <email>Johns@yourmail.com</email>
    <timezone>+01:00</timezone>
  </attendant>
  <attendant>
    <id>241</id>
    <gu>e0a8w0a212c400a563e100004orgv887</gu>
    <name>Paul</name>
    <surname>Brown</surname>
    <email>Paulb@yourmail.com</email>
    <timezone>+00:00</timezone>
  </attendant>
</attendants>
</meeting>
<meetings>
</calendarresponse>

```

getRooms

Get a list of all resources.

Request

<http://servidor/hipergate/servlet/HttpCalendarServlet?command=getRooms&token=xxxxxxx&type=CLASSROOM>

The type parameter is optional.

Response XML

```

<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="getMeeting" code="0">
  <error></error>
  <value>2</value>

```

```

<rooms>
  <room type="CLASSROOM" active="1">
    <name>EINSTEIN</name>
    <comments></comments>
  </room>
  <room type="CLASSROOM" active="0">
    <name>NEWTON</name>
    <comments>This classroom is closed</comments>
  </room>
</rooms>
</calendarresponse>

```

getAvailableRooms

Get a list of resources that are available between two dates.

Request

```

http://servidor/hipergate/servlet/HttpCalendarServlet?command
=getAvailableRooms&token=xxxxxxx&startdate=19900101000000&en
ddate=20200101000000

```

El formato para la fecha de inicio y fin debe ser yyyyMMddHHmmss

Response XML

```

<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="getAvailableRooms" code="0">
  <error></error>
  <value>2</value>
  <rooms>
    <room type="AULA" active="1">
      <name>EINSTEIN</name>
      <comments></comments>
    </room>
    <room type="AULA">
      <name>NEWTON</name>
      <comments>Podría estar reservada</comments></room>
  </rooms>
</calendarresponse>

```

isAvailableRoom

Get whether a resource is available between two dates.

Request

```

http://servidor/hipergate/servlet/HttpCalendarServlet?command
=isAvailableRoom&token=xxxxxxx&startdate=19900101000000&endda
te=20200101000000&room=EINSTEIN

```

The format for start and end date must be yyyyMMddHHmmss

Response XML (true at element <value> if the resource is available, false otherwise)

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="isAvailableRoom" code="0">
  <error></error>
  <value>true</value>
</calendarresponse>
```

storeMeeting

Insert a new meeting or update a previously existing one.

Request

```
http://servidor/hipergate/servlet/HttpCalendarServlet?command=
storeMeeting&token=xxxxxx&meeting=idicalendar@hipergate.org&
title=Activity%20Title&startdate=20101117184000&enddate=20101
118195000&rooms=ROOM1,ROOM2&attendants=guest@mail.com
```

The parameters meeting, title, startdate and enddate are all required.

The meeting parameter must be the unique iCalendar identifier of the meeting.

The parameters rooms and attendants are optional. They must contain a list of comma separated values with the names of resources and mails of the attendants. These emails must belong to user accounts that already exist at the same domain of the calendar.

The logged user is always added as attendant and organizer of the meeting, even if his email is not explicitly added to the attendants list at the request.

Response

```
<?xml version="1.0" encoding="UTF-8"?>
<calendarresponse command="storeMeeting" code="0">
  <error></error>
  <value>true</value>
  <meetings count="1">
    <meeting type="meeting">
      <id>idprueba@hipergate.org</id>
      <gu>c0a8012112c5bb9435b100000c940265</gu>
      <startdate>2010-11-17T18:40:00</startdate>
      <enddate>2010-11-18T19:50:00</enddate>
      <privacy>0</privacy>
      <title>Activity Title</title>
      <description>Activity Description</description>
      <organizer>
        <id>458</id>
        <gu>uuu8w0a212c400a563e100004orga765</gu>
        <name>Organizer</name>
```

```

        <surname>Attendant</surname>
        <email>user@mail.com</email>
        <timezone>+01:00</timezone>
    </organizer>
    <rooms count="2">
        <room type="" active="1">
            <name>ROOM1</name>
            <comments></comments>
        </room>
        <room type="" active="1">
            <name>ROOM2</name>
            <comments></comments>
        </room>
    </rooms>
    <attendants count="2">
        <attendant>
            <id>458</id>
            <gu>uua8w0a212c400a563e100004orga765</gu>
            <name>Organizer</name>
            <surname>Attendant</surname>
            <email>user@mail.com</email>
            <timezone>+01:00</timezone>
        </attendant>
        <attendant>
            <id>387</id>
            <gu>ppa8w0a212c400a563e100004orgd937</gu>
            <name>Guest</name>
            <surname>Attendant</surname>
            <email>guest@mail.com</email>
            <timezone>+01:00</timezone>
        </attendant>
    </attendants>
</meeting>
</meetings>
</calendarresponse>

```

.NET client library

The library `HipergateCalendarClient.dll` contains the classes `CalendarMeeting`, `CalendarRoom`, `CalendarAttendant` y `CalendarClient`.

The main class for accessing the calendar service from a .NET client program is `Hipergate.CalendarClient`.

Class `Hipergate.CalendarClient`

This class keeps the session at client side and provides access to all relevant methods for calendar interaction.

Connect (sServiceUrl As String, sUserEmail As String, sPassword As String) As Boolean

Before calling any method of `CalendarClient` it is necessary to establish a session by calling `Connect()`.

sServiceUrl: Base URL for service. Usually
http://hostname.com/hipergate/servlet/HttpCalendarServlet

sUserEmail: E-mail of user owner the calendar. This email must be present at tables k_users and k_fellows of hipergate.

sPassword: User password. The same that the user has at k_users table of hipergate.

Disconnect () As Boolean

Close session.

IsAvailable (sRoom As String, dtStart As Date, dtEnd As Date) As Boolean

Check whether a resource is available between two given dates.

sRoom: Resource name.

dtStart: Start date.

dtEnd: End date.

GetRooms () As CalendarRoom()

Get an array with all resources, no matter if they are available or not. If there are no resources then it returns **Nothing**.

GetRooms(sType As String) As CalendarRoom()

Get an array with all resources of a given type no matter if they are available or not. If there are no resources of such type then it returns **Nothing**.

sType: Resource type.

GetAvailableRooms(dtStart As Date, dtEnd As Date) As CalendarRoom()

Get an array with all resources that are available between two given dates. If there are no available resources then it returns `Nothing`.

dtStart: Start date.

dtEnd: End date.

GetAvailableRooms(sType As String, dtStart As Date, dtEnd As Date) As CalendarRoom()

Get an array with all available resources of a given type between two dates. If there are no available resources of such type then it returns `Nothing`.

sType: Resource type.

dtStart: Start date.

dtEnd: End date.

GetMeeting (sMeetingId As String) As CalendarMeeting

Retrieve a meeting object from its iCalendar Id. or its GUID.

sMeetingId: iCalendar Id. or GUID of meeting (any of both).

GetMeetings(dtStart As Date, dtEnd As Date) As CalendarMeeting()

Get an array with all meetings between two dates.

dtStart: Start date.

dtEnd: End date.

GetMeetingsOfType(dtStart As Date, dtEnd As Date, sType As String) As CalendarMeeting()

Get an array with all meetings of a given type between two dates.

dtStart: Start date.

dtEnd: End date.

sType: Meeting type. The standard types are:

meeting
call
followup
breakfast
lunch
course
demo
workshop
congress
tradeshow

GetMeetingsForRoom (dtStart As Date, dtEnd As Date, sRoom As String) As CalendarMeeting()

Get an array with all meetings that make use of a given resource between two dates.

dtStart: Start date.

dtEnd: End date.

sRoom: Resource name.

StoreMeeting (oMeet As CalendarMeeting) As CalendarMeeting

Store a new meeting or update an already existing one. Meetings are uniquely identified by their iCalendar Id.

The iCalendar Id. may be explicitly set by the caller program or, if it is left blank, a default value is assigned to it before storing the meeting at Server side.

This function returns the saved object as it is stored at Server side. The returned object may have slight changes from the one sent. The automatically assigned iCalendar Id. and GUID are two of those changes.

oMeet: Meeting to be stored.

DeleteMeeting (sMeetingId As String) As CalendarMeeting

Delete a meeting.

oMeet: iCalendar Id. or GUID.

Example of how to use the .NET client from VisualBASIC

```
' Create a client calendar object
Dim c As New Hipergate.CalendarClient

' Connect to service
c.Connect("http://localhost/hipergate/servlet/HttpCalendarServlet",
"user@test.com", "TEST")

' Check if the resource of name NEWTON is available right now
Dim a As Boolean = c.IsAvailable("NEWTON", DateValue(Now), DateValue(Now))

' Get an array with all resources
Dim r() As Hipergate.CalendarRoom = c.GetRooms()

' Get an array with all available resources
Dim d() As Hipergate.CalendarRoom = c.GetAvailableRooms(DateValue(Now),
DateValue(Now))

' List activities between 1/1/2000 and 12/31/2020
Dim m() As Hipergate.CalendarMeeting = c.GetMeetings(New Date(2000, 1, 1, 0,
0, 0), New Date(2020, 12, 31, 23, 59, 59))

' Create a new activity at room NEWTON
Dim e As New Hipergate.CalendarMeeting
Dim f As Hipergate.CalendarMeeting
e.title = "Meeting title up to 100 characters"
e.description = "Meeting description up to 1000 characters"
e.startdate = Now ' Start date
e.enddate = DateAdd(DateInterval.Hour, 2, e.startdate)
e.AddRoom("NEWTON") ' Call AddRoom once for each resource
e.AddAttendant("administrator@hipergate-test.com")
f = c.StoreMeeting(e)

' Get stored meeting
Dim g As Hipergate.CalendarMeeting = c.GetMeeting(f.id)
' Move the start date ten minutes
g.startdate = DateAdd(DateInterval.Minute, 10, g.startdate)
c.StoreMeeting(g)

' Close session
c.Disconnect()
```

Java client library

The client side Java API for remotely accesing the calendar may be found at package `com.knowgate.addrbook.client`

The main class for accesing the calendar is `CalendarServices`.

Example of usage of Java client library

```
CalendarServices oCal = new CalendarServices();

// Connect to service
oCal.connect("http://localhost/hipergate/servlet/HttpCalendarServlet","admini
strator@hipergate-test.com","TEST");
```

```

// Ceheck if resource NEWTON is available right now
boolean a = oCal.isAvailableRoom("NEWTON", new Date(),new Date ());

// Obtener un array con todos los recursos
ArrayList<CalendarRoom> r = oCal.getRooms();

// Get an array with all available resources
ArrayList<CalendarRoom> d = oCal.getAvailableRooms(new Date (),new Date ());

// Get an array with activities between 1/1/2000 and 12/31/2020
ArrayList<CalendarMeeting> m = oCal.getMeetings(new Date (100,0,1), new Date
(120,11,31));

// Store a new activity at room NEWTON
CalendarMeeting e = new CalendarMeeting();
e.setTitle("Test Activity Title");
e.setDescription("Test Activity Extended Description");
e.setStartDate(new Date());
e.setEndDate(new Date(e.getStartDate().getTime()+36000001));
e.addRoom("NEWTON");
e.addAttendant("administrator@hipergate-test.com");
oCal.storeMeeting(e);

```