

Improving Interoperability by Clustering of Heterogeneous Information Networks

Kanudan Rohadiya¹, Dr. Hiren Joshi²

¹Web Technology, Master of Computer Science Engineering Department, Gujarat University/Rollwala Institute of computer Studies, Ahmadabad, Gujarat, India

²Professor, Gujarat University/ Rollwala Institute of computer Studies, Ahemedabad, Gujarat, India

ABSTRACT

An interoperable web service is one which can work across platforms, languages, applications and with web services from different vendors, at present or future, in either implementation or access, without any restrictions. The paper represents a faster approach for social media event finding in order to support fast access to information that users find relevant. The main focus of research includes combining multiple types of data from social media in a heterogeneous network. The research introduces an idea to create Calendar Service Agent as a software agent in order to exchange, modify and synchronize information about events from different calendar platforms. For example, clustering relevant topics together allows business users to go directly to the cluster of business related events.

Keywords : Interpretability, Heterogeneous, iCalendar, iMIP, CCS, iTAP, event calendar, Facebook event

I. INTRODUCTION

The notions of interoperability will become ever more important in the environment where services will be shared internally and externally in ever more inter-dependent extended enterprises because The use of calendaring and scheduling has grown considerably in the last decade. Enterprise and inter-enterprise business has become dependent on rapid scheduling of events and actions using this information technology. Defining interoperability in a clear unambiguous manner at several levels (business/service, information, and technical) is a useful architecture planning tool. Implementing interoperability requires the creation, management, acceptance, and enforcement of realistic standards. Clear measures of interoperability are key to success. The paper introduces an idea to create Calendar Service Agent as a software agent in order to exchange, modify and synchronize information about events from different calendar platforms. The solution

is explained by protocols and patterns, as well as the structure and architecture, and platforms included.

The contents of the paper are organized in following manner. Firstly, it represents the Requirement of interoperability explaining present approach, and then protocols concerning calendars, and then it describes the architectural and structural overview of the new proposed solution which not only enhanced the compatibility of information but adds ease to reusability of information over an internet.

II. REQUIRMENT OF INTEROPERABILITY

The initial motivation in this research was to exploit the calendar interoperability and to create a stand-alone software agent which joins all the popular calendar services at one place and offers the users a possibility to join all their calendar and scheduling events at one place, otherwise as shown in Figure 1, case where multiple calendar scenario is used, which does not include most of today's' popular calendar services.

In this paper, it gives an illustration of possibilities to exchange and collecting different information from various information providers from internet Calendaring and Scheduling Core Object Specification.

Figure 1 describes the scenario in which without common format, there will be interoperability problems between dissimilar applications that are not supporting the same format for defining the calendar information. Different organizations and commercial vendors develop their own calendaring and scheduling model and structures. If the calendar representation format is not interoperable, calendar applications cannot communicate with each other.

III. EARLIER APPROACH LIMITATION

No solution includes today's Popular Calendar and Event services like Facebook Events, Eventbrite.com, Meetup.com and Google Calendar. User are forced to user different calendar for different systems for mobile, desktop, web and so User need to check different GUI for events which is time consuming as no common Data available as all of these has own API for Event & Calendar Data

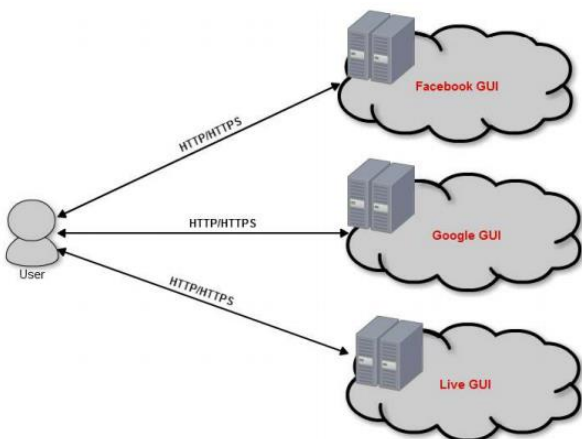


Fig 1: User connection to multiple calendar providers via their GUI.

Proposed Research:

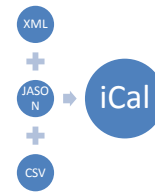


Fig 2: Unique solution based on Web Service which offer more adaptability and Support data transfer using both XML, JASON, CSV.

The main components of the research include following:

Internet Calendaring and Scheduling Core Object Specification (iCalendar)

iCalendar is a computer file formats which allows Internet users to send meeting requests and tasks to other Internet users by sharing or sending files in this format through various methods. iCalendar was first created in 1998 by the Calendaring and Scheduling Working Group of the Internet Engineering Task Force. iCalendar files typically have the file extension ".ical" ".ics" ".ifb" or ".icalendar" with a MIME type of "text/calendar". The top-level element in iCalendar is the Calendaring and Scheduling Core Object, a collection of calendar and scheduling information. Typically, this information will consist of a single iCalendar object. However, multiple iCalendar objects can be grouped together.

```
BEGIN:VCALENDAR
VERSION:2.0
PRODID:-//hacksw/handcal//NONSGML v1.0//EN
BEGIN:VEVENT
UID:uid1@example.com
DTSTAMP:19970714T170000Z
ORGANIZER;CN=John
Doe:MAILTO:john.doe@example.com
DTSTART:19970714T170000Z
DTEND:19970715T035959Z
SUMMARY:Bastille Day Party
END:VEVENT
END:VCALENDAR
```

The first line must be BEGIN:VCALENDAR, and the last line must be END:VCALENDAR; the contents between these lines is called the "icalbody". The second line VERSION:2.0 indicates that the data is in iCalendar format. VERSION:1.0 was used to specify that data is in the old vCalendar format. The body of the iCalendar object (the icalbody) is made up of a list of calendar properties and one or more calendar components. The calendar properties apply to the entire calendar. The calendar components are several calendar properties which create a calendar schematic, design.

iCalendar Message-Based Interoperability Protocol (iMIP):

It specifies binding from the iCalendar Transport-independent Interoperability Protocol (iTIP) to Internet email-based transports. This binding document provides the transport specific information necessary convey iCalendar Transport-independent Interoperability Protocol (iTIP) over MIME

Compatible Collaborative Calendar-Server (CCS):

CCS is implemented based on the iCalendar specification as a collection of Web Services. The bridge module allows the CCS to communicate with different Calendaring and Scheduling applications, which are based on the iCalendar specification and use http methods for interactions.

iCalendar Transport-Independent Interoperability Protocol (iTIP)

A protocol that uses the iCalendar objects specification to provide scheduling interoperability between different calendaring systems. This is done without reference to a specific transport protocol so as to allow multiple methods of communication between systems. Subsequent documents will define profiles of this protocol that use specific,

interoperable methods of communication between system.

Our next step is to apply this approach to iCal Event Output which can explore the result across web, mobile and desktop application.

IV. REFERENCES

- [1]. W. W. Ahmet Fatih Mustacoglu and G. Fox. Internet calendaring and scheduling core object specification (icalendar) compatible collaborative calendar-server (ccs) web services. In International Symposium on Collaborative Technologies and Systems, pages 12–17, 2006.
- [2]. IETF <https://www.ietf.org/>
- [3]. CalSch-IETFgroup <https://datatracker.ietf.org/wg/calsch/about/>
- [4]. iTIP RFC 2446 <https://tools.ietf.org/html/rfc2446>
- [5]. iMIP RFC 2447 <https://tools.ietf.org/html/rfc2447>
- [6]. D. Crow. Apple's ical and ical/vcal format
- [7]. F. Dawson. Emerging calendaring and scheduling standards. *Computer*, 30(12):126–128, Dec. 1997.
- [8]. Google. Import events from icalendar or csv files.
- [9]. Microsoft. View and subscribe to internet calendars.
- [10]. IBM. Ibm lotus notes 8.5 icalendar: Interoperability, implementation, and application.
- [11]. I. E. T. Force. Internet calendaring and scheduling core object specification (icalendar), Sept. 2009.
- [12]. iCalendar RFC 5545 <https://tools.ietf.org/html/rfc2447>
- [13]. CalDev RFC 4791 <https://tools.ietf.org/html/rfc4791>
- [14]. Facebook Events <https://events.fb.com/#why-facebook-events>

- [15]. EventBrite eventbrite.com
- [16]. Meetup Events <http://www.Meetup.com>
- [17]. Goran Velkoski, Marjan Gusev, Krste Bozinoski, Sasko Ristov, Marjan Gusev “Service Agents For Calendar Exchange” IBM 2006
- [18]. C. Daboo (Apple) “Calendaring Extensions to WebDAV (CalDAV): Time Zones by Reference” – 2016
- [19]. Daboo C., Desruisseaux B., Dusseault L. Links Calendaring Extensions to WebDAV (CalDAV)
- [20]. Kewisch P., Daboo C., Douglass M., “jCal: The JSON Format for iCalendar”, RFC 7265, 2014
- [21]. xCal Daboo C., Douglass M., Lees S. Links, “xCal: The XML Format for iCalendar” RFC 6321, 2011
- [22]. Mustacoglu, Ahmet Fatih Fox, Geoffrey, “AJAX Integration Approach for Collaborative Calendar-Server Web Services”, 2012