iCalendar
A Language for Manage Events

Mengfei Ren, Yun Feng, Chang Hu, Jiacheng Chen, Yu Kang

{mr3258, yf2294, ch2950, jc3940, yk2540}@columbia.edu

1 Language Description

Every day we have a lot of things to do which make us crazy. That is why many people used to take a little notebook with them to record important things. In the fast-paced modern life, sometimes it is very difficult for people to remember to do everything they need to. From this angle, it is essential to make a kind of electronic calendar with the help of which people could create their own calendar, store and manage events. As a result, we plan to design a new language, the iCalendar Language, specifically for the implementations about calendars. We wish that with iCalendar language, users could use more efficient and user-friendly programming methods to implement operations on calendars.

This manual describes the iCalendar language. The iCalendar programming language is an effective programming language for recording your daily events. Users could build their event models (structure) and traverse the calendar (container) to manage their events.

2 Example Code

The goal of iCalendar is to let users define their own event model, store and manage real event objects.

Users could define their own event models with language like this:

```
Event myEvent
{
    int priority;
    string description;
    string location;
}
```

Now as we have our own event model, we can record many events.

To declare an event, we could use:

```
myEvent e1 = [10, "final exam", "CS 401"];
myEvent e2 = [1, "dinner with Mary", "Ollise"];
```

We could then use a calendar to maintain all our events:

```
Calendar c = [e1, e2];
```

Our language is much like C- or Java language.

So we could do some basic operation to manage event. For example, we could choose which event is more important by using if else clause.

```
if (e1.priority > e2.priority){
    Print(e1.desciprtion)
}
else
{
    Print(e2.priority);
}
```

If time permits, we hope to develop calendar function. For example, in a Calendar, we could use c.sortBy() to sort events by certain attribute. Consider the hard of this part, we might not be possible to finish this part.