# Zend_Ical - Ben Scholzen

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

## Zend Framework: Zend_Ical Component Proposal

| | |
|---|---|
| **Proposed Component Name** | Zend_Ical |
| **Developer Notes** | http://framework.zend.com/wiki/display/ZFDEV/Zend_Ical |
| **Proposers** | • Ben Scholzen<br>• Alexander Veremyev<br>  • (Zend-liaison) |
| **Revision** | 1.1 - 22 September 2007: Initial proposal<br>1.2 - 03 March 2008: Proposal rewrite<br>1.3 - 19 May 2008: Proposal update (wiki revision: 28) |

## Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
Open a ICS calendar file
Fetching specific events
Iterating through all events
9. Class Skeletons

## 1. Overview

Zend_Ical is a component to manage events, alarms and todos in calendar objects.

## 2. References

- iCal Wikipedia Entry
- LALR parser Wikipedia Entry
- RFC 2445

## 3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** handle ICS files from different data sources (Filesystem, FTP and HTTP (webDAV required for writing)).
- This component **will** support the entire ICS 2.0 specification
- This component **will** be able to create/write new ICS data
- This component **will** allow searching and iteration through calendar data
- This component **must** validate any input before writing them to the source

## 4. Dependencies on Other Framework Components

- Zend_Exception
- Zend_Date
- Zend_Uri

## 5. Theory of Operation

Zend_Ical will take an URI (String or Zend_Uri) as argument for the constructor method. Zend_Ical will then parse the ICS file through a LALR parsing mode. All time values within the ICS data will be converted to the local time zone.

After the parsing you can call different methods to gather data from the calendar or add additional data to it. You can also iterate through the calendar object to get all data available.

At the time of writing, Zend_Ical will only support the iCalender 2.0 format.

## 6. Milestones / Tasks

- Milestone 1: [done] Proposal finished
- Milestone 2: [done] Development of prototype started and checked into http://zend.svn.dasprids.de/
- Milestone 3: Proposal approved
- Milestone 4: Working prototype checked into the incubator.
- Milestone 5: Unit tests exist, work, and are checked into SVN.
- Milestone 6: Documentation exists.

## 7. Class Index

- Zend_Ical
- Zend_Ical_Exception
- Zend_Ical_Parser
- Zend_Ical_Source_Exception
- Zend_Ical_Source_Abstract
- Zend_Ical_Source_Filesystem
- Zend_Ical_Source_Ftp
- Zend_Ical_Source_Http

## 8. Use Cases

**UC-01**

### Open a ICS calendar file

```
$calendar = new Zend_Ical('http://user:password@www.zend.com/mycalendar.ics');
```

**UC-02**

### Fetching specific events

```
$calendar->getEvents('Feb 31, 2007');
```

**UC-03**

## Iterating through all events

```
foreach ($calendar->events as $event) {
    // Do something
}
```

## 9. Class Skeletons

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>