

# **The Kolab Storage Format**

**2.0rc2**

**Bo Thorsen, bo@sonofthor.dk**  
**David Faure, dfaure@klaralvdalens-datakonsult.se**  
**Joon Radley, joon@radleys.co.za**  
**Martin Konold, martin.konold@erfrakon.de**  
**Stephan Buys, s.buys@codefusion.co.za**  
**Stuart Binge, s.binge@codefusion.co.za**  
**[www.kolab.org](http://www.kolab.org)**

## The Kolab Storage Format: 2.0rc2

by

Bo Thorsen, bo@sonofthor.dk

David Faure, dfaure@klaralvdalens-datakonsult.se

Joon Radley, joon@radleys.co.za

Martin Konold, martin.konold@erfrakon.de

Stephan Buys, s.buys@codefusion.co.za

Stuart Binge, s.binge@codefusion.co.za

www.kolab.org

Published May 16th, 2005

This documentation was written in SGML using the DocBook DTD. HTML and Postscript output is generated automatically and depends on the tools used.

Windows XP®, Microsoft Outlook® are registered trademarks of Microsoft Corporation Inc. Toltec Connector™ K Desktop Environment™ and KDE™ are trademarks of the KDE e.V.

All other herein mentioned trademarks belong to their respective owners. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Finally, the authors of this book are not liable for any errors found as well as anything that may cause a fault. However, if that does occur, please notify the authors so corrections can be made. Furthermore, the reader must also agree to use the information in this book at his/her own risk and relinquish the authors, from any mistakes due to this book. If not, please stop reading now.

BECAUSE THE CONTENT IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE CONTENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW. THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE CONTENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK OF USE OF THE CONTENT IS WITH YOU. SHOULD THE CONTENT PROVE FAULTY, INACCURATE, OR OTHERWISE UNACCEPTABLE YOU ASSUME THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MIRROR AND/OR REDISTRIBUTE THE CONTENT AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE CONTENT, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### Revision History

Revision 0.0.1            June 9th 2004

Revision cvs20040813 August 13th, 2004

Revision cvs20040903 September 3rd 2004

Changed tag "email" to "smtp-address"; added tag "product-id"; added tag "color-label" description.

Revision cvs20041007 October 7th 2004

Updated start-date, attachments. Removed invitation-sent. Added creator.

Revision 2.0rc1            May 3rd, 2005

Added version attribute, "yearly weekday" recurrence, scheduling-id; Changed date to daynumber.

Revision 2.0rc2            May 12th, 2005

Removed scheduling-id.

# Table of Contents

<b>1. Kolab Storage Format Overview</b> .....	<b>1</b>
1.1. Mail structure .....	1
1.2. XML format description .....	1
1.2.1. Types Used .....	2
<b>2. Per Folder requirements</b> .....	<b>4</b>
2.1. IMAP requirements.....	4
2.2. Mail contents.....	4
<b>3. Mail Folders</b> .....	<b>6</b>
<b>4. Common Fields</b> .....	<b>7</b>
4.1. Common In All Types.....	7
4.2. Common In Tasks And Events.....	7
4.2.1. Recurrence .....	8
4.2.2. Recurrence examples.....	9
4.2.3. Attendees .....	11
<b>5. Format Of Notes</b> .....	<b>12</b>
<b>6. Format Of Contacts</b> .....	<b>13</b>
<b>7. Format Of Distribution Lists</b> .....	<b>15</b>
<b>8. Format Of Journals</b> .....	<b>16</b>
<b>9. Format Of Events</b> .....	<b>17</b>
<b>10. Format Of Tasks</b> .....	<b>19</b>

# Chapter 1. Kolab Storage Format Overview

This document describes what a Kolab client can store in the IMAP folders on a Kolab server.

The Kolab storage format is designed to provide a unified storage format for several Kolab client implementations.

## 1.1. Mail structure

Each Kolab object (note, event, contact etc.) is stored in its own email, in the appropriate folder for this type of object.

The email uses a multipart/mixed structure, with the following parts:

- A text body part with a fixed text telling about Kolab. Content-Type: text/plain
- The XML describing the object. Content-Type: application/x-vnd.kolab.\*
- Other attachments, like a contact's picture or the attachments associated with e.g. a note or event.

The following requirements apply to the mail headers:

- The subject of the mail must be set to the UID of the object. (\*)
- An additional header named X-Kolab-Type, must be set to the mimetype of the object, application/x-vnd.kolab.\*

(\*) This can be used by some clients for searching but it is not recommended to implement clients that way; searching on the IMAP ID is much faster when it's possible.

## 1.2. XML format description

The storage format specifies the format of the XML to be saved as an attachment in the mail.

The fields in the XML file are written like this:

```
<field>(type, default)</field>
```

If a field entry is the same as the default, then it's optional whether to write it or not - exception from this is the creation and last modification dates. If there are attributes with a limited amount of options, these are specified after the text.

All tag names, attribute names, and attribute values are case sensitive.

If a client sees a tag it does not understand, this tag must be preserved and saved back to the file. This allows for client specific tags. Outlook writes its client specific tags directly in a tnef file that is saved as an unreferenced attachment.

Each type of folder has its own subclass of the specification. They all share this same and very simple XML header:

```
<?xml version="1.0" encoding="UTF-8"?>
```

After this, we have a single node with the content of the file. One file can only hold one instance of the events, contacts, ... This looks like:

```
<type version="1.0">  
  Entry fields go here  
</type>
```

where `<type>` can be note, event, task, journal, or contact. If a client has a new type, it is allowed to implement this, but then other clients will not be able to handle the folder entry.

The version attribute denotes the version of this specification. Until a stable version is released, the number "1.0" is used.

Here is a short example of a note:

```
<?xml version="1.0" encoding="UTF-8"?>  
<note version="1.0">  
  Note stuff goes here  
</note>
```

### 1.2.1. Types Used

The types that each entry can be are:

- String (utf8 encoded, can be multiple lines)
- Number (actually an int)
- Date YYYY-MM-DD
- Datetime YYYY-MM-DDThh:mm:ssZ. The T is the literal char, and always needs to be between the date and the time. The Z is also the literal char, and shows that the datetime is UTC. The format does not allow for other time settings than UTC. An example could be 2004-05-04T15:00:00Z.
- Color #RRGGBB -- The rgb values are case insensitive, so for example #00aaFF is allowed.
- Bool (true or false)

# Chapter 2. Per Folder requirements

This chapter describes all necessary settings on the IMAP folders in the Kolab storage format.

## 2.1. IMAP requirements

We will be using the ANNOTATEMORE proposal. If changes are made to this in the standardization process, we will incorporate these. Also, we must support IMAP namespaces. All relevant RFCs will be followed (i.e. folder name encoding is by RFC 3501).

The INBOX is the default inbox of the user. The user can not change this default. The IMAP resource folders (one for events, one for contacts etc. ) are subfolders of the INBOX, and this location can not be set by the user.

We will annotate all folders with an entry /vendor/kolab/folder-type containing the attribute value.shared set to: <type>[.<subtype>]

The <type> can be: mail, event, journal, task, note, or contact.

The <subtype> for a mail folder can be inbox, drafts, sentitems, or junkemail (this one holds spam mails). For the other <type>s, it can only be default, or not set.

For other types of folders supported by the clients, these should be prefixed with "k-" for KMail, "h-" for Horde and "o-" for Outlook, and look like for example "kolab.o-voicemail".

We will use "application/x-vnd.kolab.<type>" as the mimetype for the emails used for storage.

The annotation must be set on creation, and can not be changed. For folders created with other IMAP clients, annotations must be preserved. If a folder has no annotation, Outlook and the Toltec connector will ignore it completely. This is not an option for Kontact and Horde, so we will by default consider them to hold mail.

## 2.2. Mail contents

The IMAP folder contents will of course normally be email. But in the special folders containing notes, events, tasks and so on, the following rule applies.

We store on note/event/task/... in each mail on the IMAP server. The mail we store in will have a so far undefined text as the body (could for example be a link to [www.kolab.org](http://www.kolab.org) and say why you can not view

this as a normally mail) (TODO: Define the body contents or make it be a thing for the client to decide), and any number of attachments. One of these attachments will be the XML file containing the actual entry (the note/event/task/...). The mimetype of this attachment will be "application/x-vnd.kolab.<type>" where type is the content type of the file. Some of these attachments can be referenced in the file, and these are "owned" by the XML file, meaning if the client application deletes the attachment reference, the attachment should also be deleted. Those attachments that are not referenced must be preserved when the mail is saved again.

# Chapter 3. Mail Folders

Mail folders hold mails in the format as described in RFC 822.

# Chapter 4. Common Fields

## 4.1. Common In All Types

There are a number of fields that are present in all non-mail folder types. These are:

- uid (string, no default)
- body (string, default empty)
- categories (string, default empty)
- creation-date (datetime, no default)
- last-modification-date (datetime, no default)
- sensitivity (string, default public)
- inline-attachment (string, no default)
- link-attachment (string, no default)
- product-id (string, default empty)

The categories is a comma separated list. There is no fixed set of categories in this format. The clients are free to choose.

The sensitivity possibilities are private, confidential and public.

There can be any number of attachments. "inline-attachment" references the filename of attachments in the mail where this is stored. Attachments not referenced from the XML should simply not be displayed to the user. This allows to attach application-specific attachments (like TNEF for Outlook) that remain hidden to the user.

"link-attachment" means the attachment is external; the contents of the tag is the URL.

The product id is the name of the last application that created or modified the object. It is mostly used for debugging.

## 4.2. Common In Tasks And Events

These are the fields that are both in tasks and events. This base class can be called incidence.

```
<summary>(string, default empty)</summary>  
<location>(string, default empty)</location>
```

```

<creator>
  <display-name>(string, default empty)</display-name>
  <smtp-address>(string, default empty)</smtp-address>
</creator>
<organizer>
  <display-name>(string, default empty)</display-name>
  <smtp-address>(string, default empty)</smtp-address>
</organizer>
<start-date>(date or datetime, default not present)</start-date>
<alarm>(number, no default)</alarm>
<recurrence cycle="cycletype" [type="extra type"]>
  <interval>(number, default 1)</interval>
  {<day>(string, no default)</day>}
  <daynumber>(number, no default)</daynumber>
  <month>(string, no default)</month>
  <range type="rangetype">(date or number or nothing, no default)</range>
  {<exclusion>(date, no default)</exclusion>}
</recurrence>
{<attendee>
  <display-name>(string, default empty)</display-name>
  <smtp-address>(string, default empty)</smtp-address>
  <status>(string, default none)</status>
  <request-response>(bool, default true)</request-response>
  <role>(string, default required)</role>
</attendee>}

```

The start-date is optional for tasks. For events, it is required. If they are there, they can either have a date or a datetime as the type. Parsing this is just a matter of looking at the length of the date string. In the case of an all day event (floating event) the end-date must also be in date only format

The alarm specifies number of minutes before the incidence that the alarm should fire. In case of incidences with only a start date but no specific time, this means minutes before 0:00 on that day.

### 4.2.1. Recurrence

There can be one <recurrence> tag. This tag has an attribute cycle that is one of "daily", "weekly", "monthly", or "yearly". Depending on the cycle, different subtags are valid:

daily: Interval specifies "every X days".

weekly: Interval specifies "every X weeks". Day can be monday, tuesday wednesday, thursday, friday, saturday, and sunday. There can be 1 to 7 of these days.

monthly: The recurrence tag has a second attribute type, which can be either "daynumber" or "weekday". In both cases, interval specifies "every X months". In the case of "daynumber", tag <daynumber> gives the date in the month this recurs on. For "weekday", tags <daynumber> and <day> must be there.

yearly: The recurrence tag has a second attribute type, which can be either "monthday", "yearday" or "weekday". In both cases, interval specifies "every X years". "monthday" is e.g. "23rd of March", and uses the tags <daynumber> and <month>. "yearday" is e.g. "155th day in the year", as specified by <daynumber>. This subtype isn't supported by Outlook. "weekday" is e.g. "2nd Friday of September", and uses the tags <daynumber>, <day> and <month>.

The range must also be present. This can be "none", which means a never ending recurrence. Or "number", which specifies the number of times this recurrence happens (before exclusions are subtracted). Or "date", which means the recurrence does not happen after this date.

Finally there can be any number of exclusions. These are dates that are removed from the list of recurrences after all other calculations are done. This specifically means if you recur three times but have an exclusion on one of the dates, there will actually only be two recurrences.

#### 4.2.2. Recurrence examples

Neverending incidence every 4 days with no exclusions:

```
<recurrence cycle="daily">
  <interval>4</interval>
  <range type="none"/>
</recurrence>
```

Recurrence weekly on mondays and thursdays, until 5 has happened. No exclusions:

```
<recurrence cycle="weekly">
  <interval>3</interval>
  <day>monday</day>
  <day>thursday</day>
  <range type="number">5</range>
</recurrence>
```

Same one, but this time with one exclusion. Note that the actual ending is the same, meaning in reality only four incidences happened:

```

<recurrence cycle="weekly">
  <interval>3</interval>
  <day>monday</day>
  <day>thursday</day>
  <range type="number">5</range>
  <exclusion>2005-05-04</exclusion>
</recurrence>

```

Monthly recurrence. Happens until June 1st 2006 on the 3rd of every second month with no exclusions:

```

<recurrence cycle="monthly" type="daynumber">
  <interval>2</interval>
  <daynumber>3</daynumber>
  <range type="date">2006-06-01</range>
</recurrence>

```

Monthly infinite recurrence with two exclusions. Happens every second thursday of every sixth month:

```

<recurrence cycle="monthly" type="weekday">
  <interval>6</interval>
  <daynumber>2</daynumber>
  <day>thursday</day>
  <range type="none" />
  <exclusion>2005-12-12</exclusion>
  <exclusion>2006-06-15</exclusion>
</recurrence>

```

Yearly recurrence ending after 2005 with no exclusions. Happens every May 4th:

```

<recurrence cycle="yearly" type="monthday">
  <interval>1</interval>
  <daynumber>4</daynumber>
  <month>june</month>
  <range type="date">2005-12-31</range>
</recurrence>

```

Yearly infinite recurrence with no exclusions. Happens every second year on the 125th day:

```
<recurrence cycle="yearly" type="yearday">  
  <interval>2</interval>  
  <daynumber>125</daynumber>  
  <range type="none" />  
</recurrence>
```

Yearly infinite recurrence with no exclusions. Happens every third year on the 2nd Friday of September:

```
<recurrence cycle="yearly" type="weekday">  
  <interval>3</interval>  
  <daynumber>2</daynumber>  
  <day>friday</day>  
  <month>september</month>  
  <range type="none" />  
</recurrence>
```

### 4.2.3. Attendees

There can be any number of attendees.

The status must be one of none, tentative, accepted, or declined. role is one of required, optional, or resource.

## Chapter 5. Format Of Notes

The mimetype for notes is "application/x-vnd.kolab.note". This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<note version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Note specific fields -->
  <summary>(string, default empty)</summary>
  <background-color>(color, default #000000)</background-color>
  <foreground-color>(color, default #ffff00)</foreground-color>
</note>
```

See Section 4.1> for the description of the common fields

## Chapter 6. Format Of Contacts

The mimetype for contacts is "application/x-vnd.kolab.contact". This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<contact version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Contact specific fields -->
  <name>
    <given-name>(string, default empty)</given-name>
    <middle-names>(string, default empty)</middle-names>
    <last-name>(string, default empty)</last-name>
    <full-name>(string, default empty)</full-name>
    <initials>(string, default empty)</initials>
    <prefix>(string, default empty)</prefix>
    <suffix>(string, default empty)</suffix>
  </name>
  <free-busy-url>(string, default empty)</free-busy-url>
  <organization>(string, default empty)</organization>
  <web-page>(string, default empty)</web-page>
  <im-address>(string, default empty)</im-address>
  <department>(string, default empty)</department>
  <office-location>(string, default empty)</office-location>
  <profession>(string, default empty)</profession>
  <job-title>(string, default empty)</job-title>
  <manager-name>(string, default empty)</manager-name>
  <assistant>(string, default empty)</assistant>
  <nick-name>(string, default empty)</nick-name>
  <spouse-name>(string, default empty)</spouse-name>
  <birthday>(date, no default)</birthday>
  <anniversary>(date, no default)</anniversary>
  <picture>(string(attachment filename), default empty)</picture>
  <children>(string, default empty)</children>
  <gender>(string, default empty)</gender>
  <language>(string, default empty)</language>
  {<phone>
    <type>(string, no default)</type>
    <number>(string, default empty)</number>
```

```

</phone>}
{<email>
  <display-name>(string, default empty)</display-name>
  <smtp-address>(string, default empty)</smtp-address>
</email>}
{<address>
  <type>(string, default home)</type>
  <street>(string, default empty)</street>
  <locality>(string, default empty)</locality>
  <region>(string, default empty)</region>
  <postal-code>(string, default empty)</postal-code>
  <country>(string, default empty)</country>
</address>}
<preferred-address>(string, default none)</preferred-address>
<latitude>(float, no default)</latitude>
<longitude>(float, no default)</longitude>
</contact>

```

See Section 4.1> for the description of the common fields

A contact has a field for free text for it. This is usually called a note on the contact. This note is stored in the body tag.

You can have one phone number of each type. The types are: business1, business2, businessfax, callback, car, company, home1, home2, homefax, isdn, mobile, pager, primary, radio, telex, ttyddd, assistant, and other.

The Kontact and Horde address books have the notion of a preferred email address. This could be done by a bool on the emails or a separate tag like for preferred address.

The address type can be home, business, and other. The preferred address can be set to one of these or none.

In an address, locality is the city or village name, the meaning of region depends on the country (e.g. it's the state in federal states like the USA), and postal-code is what is also known as "zip" in some countries.

# Chapter 7. Format Of Distribution Lists

A distribution list is a special contact. This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<distribution-list version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Distribution-list specific fields -->
  <display-name>(string)</display-name>
  {<member>
    <display-name>(string)</display-name>
    <smtp-address>(string)</smtp-address>
  </member>}
</distribution-list>
```

See Section 4.1> for the description of the common fields

A distribution list is a list of members, where each member is specified by its name and email address.

# Chapter 8. Format Of Journals

The mimetype for journals is "application/x-vnd.kolab.journal". This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<journal version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Journal specific fields -->
  <summary>(string, default empty)</summary>
  <start-date>(datetime, default not present)</start-date>
  <end-date>(datetime, default not present)</end-date>
  {<contact>
    <display-name>(string, default empty)</display-name>
    <smtp-address>(string, default empty)</smtp-address>
  </contact>}
</journal>
```

See Section 4.1> for the description of the common fields

There can be any number of contacts, which is why these are in { }.

## Chapter 9. Format Of Events

The mimetype for events is "application/x-vnd.kolab.event". This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<event version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Incidence fields -->
  <summary>(string, default empty)</summary>
  <location>(string, default empty)</location>
  <organizer>
    <display-name>(string, default empty)</display-name>
    <smtp-address>(string, default empty)</smtp-address>
  </organizer>
  <start-date>(date or datetime, default not present)</start-date>
  <alarm>(number, no default)</alarm>
  <recurrence cycle="cycletype" [type="extra type"]>
    <interval>(number, default 1)</interval>
    {<day>(string, no default)</day>}
    <daynumber>(number, no default)</daynumber>
    <date>(number, no default)</date>
    <month>(string, no default)</month>
    <range type="rangetype">(date or number or nothing, no default)</range>
    {<exclusion>(date, no default)</exclusion>}
  </recurrence>
  {<attendee>
    <display-name>(string, default empty)</display-name>
    <smtp-address>(string, default empty)</smtp-address>
    <status>(string, default none)</status>
    <request-response>(bool, default true)</request-response>
    <role>(string, default required)</role>
  </attendee>}

  <!-- Event specific fields -->
  <show-time-as>(string, default busy)</show-time-as>
  <color-label>(string, default none)</color-label>
  <end-date>(date or datetime, default not present)</end-date>
</event>
```

See Section 4.1> for the description of the common fields

See Section 4.2> for the description of the incidence fields

show-time-as is one of free, tentative, busy, or outofoffice.

color-label can be none, important, business, personal, vacation, must-attend, travel-required, needs-preparation, birthday, anniversary, phone-call. Outlook maps those to colors, hence the name.

# Chapter 10. Format Of Tasks

The mimetype for tasks is "application/x-vnd.kolab.task". This is the specification of the body contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<task version="1.0">
  <!-- Common fields -->
  <uid>(string, no default)</uid>
  <body>(string, default empty)</body>
  <categories>(string, default empty)</categories>
  <creation-date>(datetime, no default)</creation-date>
  <last-modification-date>(datetime, no default)</last-modification-date>
  <sensitivity>(string, default public)</sensitivity>
  {<inline-attachment>(string, no default)</inline-attachment>}
  {<link-attachment>(string, no default)</link-attachment>}
  <product-id>(string, default empty)</product-id>

  <!-- Incidence fields -->
  <summary>(string, default empty)</summary>
  <location>(string, default empty)</location>
  <organizer>
    <display-name>(string, default empty)</display-name>
    <smtp-address>(string, default empty)</smtp-address>
  </organizer>
  <start-date>(date or datetime, default not present)</start-date>
  <alarm>(number, no default)</alarm>
  <recurrence cycle="cycletype" [type="extra type"]>
    <interval>(number, default 1)</interval>
    {<day>(string, no default)</day>}
    <daynumber>(number, no default)</daynumber>
    <date>(number, no default)</date>
    <month>(string, no default)</month>
    <range type="rangetype">(date or number or nothing, no default)</range>
    {<exclusion>(date, no default)</exclusion>}
  </recurrence>
  {<attende>
    <display-name>(string, default empty)</display-name>
    <smtp-address>(string, default empty)</smtp-address>
    <status>(string, default none)</status>
    <request-response>(bool, default true)</request-response>
    <role>(string, default required)</role>
  </attende>}

  <!-- Task specific fields -->
  <priority>(number, default 3)</priority>
  <completed>(number, default 0)</completed>
  <status>(string, default not-started)</status>
  <due-date>(date or datetime, default not present)</due-date>
  <parent>(string, default empty)</parent>
</task>
```

See Section 4.1> for the description of the common fields

See Section 4.2> for the description of the incidence fields

status can be one of not-started, in-progress, completed, waiting-on-someone-else, or deferred.

The priority can be a number between 1 and 5, with 1 being the highest priority.

completed is a percentage, so it must be between 0 and 100.

If this is a subtask, the parent is set to the uid of the parent task. If it's not a subtask, then the parent field must be empty.