

# Directory Description with Namaste Tags

## Abstract

Namaste (NAME AS TExt) is a file naming convention to support primitive directory-level metadata tags exposed directly via filenames. As such, Namaste tags greet visitors who request a directory listing (e.g., Linux 'ls') with a glimpse of what the directory holds. An important use is to declare a directory's "type", somewhat like a file's "magic number". A Namaste tag,  $D=tvalue$ , is usually a tag value preceded by a single-digit name, designed so that ordinary directory listings will tend to display tags, if present, in a block near the top. Tag names (digits) are currently reserved for directory type (0) and the four DC Kernel elements (1-4). Restrictions on filename characters and lengths may result in a tag value that is a lossy representation of the complete metadata value, assumed to be the content of tag file.

---

## 1. What kind of directory is this?

Identifying the type of a file is generally much easier than identifying the type of a directory (or folder). A file's type is often carried in-band as a unique multi-byte sequence occurring at or near the beginning of the file, sometimes called its "magic number".

A directory, however, has no equivalent typing mechanism, despite its being a natural container for any digital object complex enough to span a file group or file hierarchy. Namaste tags address this with a kind of in-band directory magic number that appears at or near the beginning of a typical directory listing. Effectively, they establish a well-known location within a directory for software and human users to discover what kind of directory one is dealing with.

---

## 2. Namaste tag basics

A Namaste ("name as text") tag is a metadata element that represents an element name and value directly in the name of a file. The typical form of the filename is designed so that elements should appear as a group near the top of a typical directory listing (e.g., using Linux 'ls'). There they greet the visitor, serving as labels that are quickly noticeable to users and easy to find with software. Here's a sample directory listing, the first entry of which is a Namaste tag declaring this directory's "type" to be "BagIt Version 0.96".

```
0=bagit_0.96    bag-info.txt    fetch.txt
bagit.txt       data/           manifest-md5.txt
```

This specification defines the form and meaning of Namaste tags but an application may otherwise determine how to will use them. For example, a tool that processes "widget 1.3" directories might require the presence of a "0=widget\_1.3" file. A tag's form, as a filename, is

$D=tvalue$

where  $D$  is usually a single decimal digit representing the tag name and  $tvalue$  is a string representing the tag value. What's inside the tag file named " $D=tvalue$ " is the full value string from which  $tvalue$  is derived. For example, on a Linux system,

```
% cat 0=widget_1.3
widget 1.3
```

In general,  $tvalue$  is the result of a transformation that may remap filesystem-unsafe characters and shorten the full value string. Here's another example illustrating one approach to this transformation.

```
% ls
0=dflat_1.8    admin/          splash.txt      v005/
1=Twain,_Mark  annotations/    v001/          v006/
2=Huckleberry.. data/           v002/          v007/
3=1898         enrichment/     v003/          v008/
4=12345678901.. manifest.txt    v004/
% cat ?=*
dflat 1.8
Twain, Mark
Huckleberry Finn
1898
12345678901123456
```

The purpose of Namaste tags is to help a human being get a glimpse of what the containing directory is about. Should it be needed, a tag file's

content provides a complete element value without further parsing. There is no other machine-readability requirement.

---

### 3. Basic and extended tag names

The basic Namaste tag name is a single-digit. The tag name that specifies the directory type is 0, and tag names 1-4 correspond to Dublin Core (DC) **[RFC5013]** Kernel Metadata **[Kernel]** elements h1-h4. The currently defined tag names are summarized below.

- 0=*type* — directory type string ("magic number")
- 1=*who* — who created, published, or contributed to it
- 2=*what* — what the expression was called (DC Title)
- 3=*when* — when it was expressed (DC Date)
- 4=*where* — where to find the expression (DC Identifier)

These tag names were conceived with default sorting order in mind so that directory listings would tend to display tags, if present, in a block near the top. Note that sorting is typically locale-sensitive, sometimes with results that are not immediately obvious when a directory contains other filenames that begin with digits.

An *extended* Namaste tag name is an arbitrary multi-character string of letters, digits, and underscores ('\_') that starts with a letter, underscore, or period ('.'). Extended tag names will tend not to have the same display and grouping features as single-digit tag names.

This specification does not currently define any extended names. Applications that wish to define names that won't conflict with future defined Namaste tag names should begin theirs with "x\_".

Namaste tags have no formal relationship to filesystem-supported key/value metadata, such as XFS extended attributes.

---

### 4. Transforming metadata values into tag values

As mentioned, the tag value, *tvalue* within the name "*D=tvalue*", is in general the result of a transformation of the full value string found inside the tag file. That transformation, *Tr*, may remap filesystem-unsafe characters and shorten the full value string, *fvalue*, in creating *tvalue*.

$$Tr(\text{ContentOfFileNamed}("D=tvalue")) = Tr(fvalue) = tvalue$$

The transformation process can be very flexible as it is entirely for the benefit of human users. An application that creates Namaste tags is thus free to transform values differently depending on the element and the audience. For example, it might deem element 4 ("where") to be too valuable ever to truncate, regardless of the consequences for display. As with any transformation, some *fvalues* may produce no change (i.e., *tvalue* is the same as *fvalue*); this was the case for the tag `3=1898` in the previous example.

Two common aspects of transformation are character re-mapping and string shortening. Re-mapping is necessary to avoid characters in *tvalue* that would be illegal in a filename, such as '/'. Re-mapping some legal characters may make it easier to manipulate files (e.g., changing spaces to underscores). If platform independence is desired in contemporary filesystems (Unix and Windows), the following characters found in *fvalue* should be avoided:

```
" * / : < > ? \ |
```

Shortening strings may be necessary for convenient display of a multi-column directory listing. For long strings, shortening may also be necessary because of maximum filename length restrictions (e.g., 255 characters in Windows). Shortening may occur anywhere that it is most appropriate. A common technique is to substitute the least significant characters with an ellipsis (".." or "...") to indicate missing characters. Applications may choose to truncate at the right, left, or middle of a string, and vary truncation length depending on the element.

---

## 5. Namaste directory types

This specification defines an extendable register of directory types in **Table 1**. Within this register, `N.M` refers to major and minor specification version numbers.

---

| Directory Type              | Reference         |
|-----------------------------|-------------------|
| <code>0=bagit_N.M</code>    | <b>[BAGIT]</b>    |
| <code>0=can_N.M</code>      | <b>[CAN]</b>      |
| <code>0=dflat_N.M</code>    | <b>[DFLAT]</b>    |
| <code>0=pairtree_N.M</code> | <b>[PAIRTREE]</b> |

Table 1: Namaste Directory Types

---

Additional types will be defined and may be submitted by sending email to the author. Submissions should conform to these guidelines to reduce the need for lossy transformation when creating the *tvalue*: proposed strings (*fvalues*) should not exceed 16 characters in length and should contain only letters, digits, underscores, periods, and hyphens.

---

## 6. Tag file content

Once a directory has been read, its Namaste tags, as filenames, are available without an extra disk read. While this provides efficient access, the filename metadata in *tvalue* is often lossy. To mitigate this situation, the tag file's content, *fvalue*, should be a lossless, newline-terminated, plain text representation, using UTF-8 [\[RFC3629\]](#), of *tvalue*.

The terminal newline (LF hex 0a), or its equivalent (CR hex 0d or CRLF hex 0d0a), is for convenient editing and display of a metadata value in line-oriented systems such as Unix, and should be trimmed by applications that require a strict sense of the *fvalue*.

---

## 7. References

- [BAGIT] Boyko, A., Kunze, J., Littman, J., Madden, L., and B. Vargas, "[BagIt File Package Format](#)," November 2008 ([HTML](#)).
- [CAN] "[Content Access Node \(CAN\)](#)," April 2009 ([PDF](#)).
- [DFLAT] "[Dflat: Simple File-Based Object Storage](#)," April 2009 ([PDF](#)).
- [Kernel] Kunze, J. and A. Turner, "[Kernel Metadata and Electronic Resource Citations \(ERCs\)](#)," October 2007 ([HTML](#)).
- [PAIRTREE] Kunze, J., Haye, M., Hetzner, E., Reyes, M., and C. Snively, "[Pairtrees for Collection Storage](#)," December 2008 ([HTML](#)).
- [REDD] Kunze, J., Abrams, S., Hetzner, E., and D. Loy, "[Reverse Directory Deltas \(ReDD\)](#)," June 2009 ([HTML](#)).
- [RFC3629] Yergeau, F., "[UTF-8, a transformation format of ISO 10646](#)," STD 63, RFC 3629, November 2003 ([TXT](#)).
- [RFC5013] Kunze, J. and T. Baker, "[The Dublin Core Metadata Element Set](#)," RFC 5013, August 2007 ([TXT](#)).

---

## Author's Address

John A. Kunze  
California Digital Library  
415 20th St, 4th Floor  
Oakland, CA 94612  
US

**Email: [jak@ucop.edu](mailto:jak@ucop.edu)**