# Specification

## of the physical structure and intra-container naming of VCD Containers

## Purpose

This document specifies the physical structure of a VCD Container as well as the naming of files and folders that are referenced inside a VCDPackage or VCD. In particular, it defines

- the structure of a VCD Container, i.e. a VCD Package delivered as a zip file,
- the folder structure of a VCD Package that is present in a VCD Container,
- folder naming of VCD Package folders and VCD folders,
- file naming of VCD Container, VCD Package(XML), VCDs(XML) and document files,
- referencing VCDs from a VCD Package,
- referencing files from VCD Package meta-data files and VCD meta-data files and
- internal references inside a VCD

## Log of changes

| Version | Change date | Changed by | Summary of change |
|---|---|---|---|
| 0.1 | 13.10.2010 | Daniel Reiser, Wolfgang Groiss | Initial version |
| 0.2 | 14.10.2010 | Daniel Reiser | Included additional related resource [3]; adjusted URI pattern; changed criterion and evidence ID to UUID value; |
| 0.3 | 21.10.2010 | Arianna Brutti, Piero Milani | Editing of common keywords, definitions, comments. |
| 0.4 | 29.11.2010 | Daniel Reiser | Including latest agreements of wiki discussion about cbc:FileName; Included comments of version 0.3 |
| 0.5 | 03.12.2010 | Daniel Reiser | Included latest comments concerning DocumentReference/cbc:Description |
| 0.6 | 16.02.2011 | Daniel Reiser | Update concerning file naming of VCD Containers |

## Resources covered by this specification

| Object | Description |
|---|---|
| VCD (Folder) | The Virtual Company Dossier of a single economic operator, i.e. a folder that contains an XML meta-data file as well as other files. |
| VCD (XML) | The XML meta-data file of a VCD. |
| VCD Package (Folder) | A collection of several VCDs put together into one single folder along with an XML meta-data file. |
| VCD Package (XML) | The XML meta-data file of a VCD Package. |
| VCD Container | A VCD Package (Folder) delivered as a single zip file. |

***Table 1: Resources covered by this specification***

## Related resources

| # | Resource title | URL | Author |
|---|---|---|---|
| [1] | Intra-Container Naming | Wiki page | Wolfgang Groiss |
| [2] | VCD Container Folder Structure Proposal | Document | Daniel Reiser |
| [3] | VCD CodeLists & Identifiers | Mercurial LINK | Arianna Brutti |
| [4] | VCD Container sample files | Zip File (Bidding consortium)<br>Zip file (Single tenderer) | Daniel Reiser |

***Table 2: Related resources***

## Content

## List of tables



## List of figures

## Physical structure of a VCD Container

A VCD Package is delivered as a zip file, the so called VCD Container. A VCD Container consists of:

- A meta-data file named VCDPackage_*.xml, i.e. the VCD Package (XML) file,
- Zero or more files representing additional documents,
- One sub folder for each VCD of an economic operator, i.e. the VCD (Folder), each containing:
    - A meta-data file named VCD_*.xml, i.e. the VCD (XML),
    - Zero or more files representing evidences and additional documents.

An example folder structure is illustrated in Figure 1.[1]

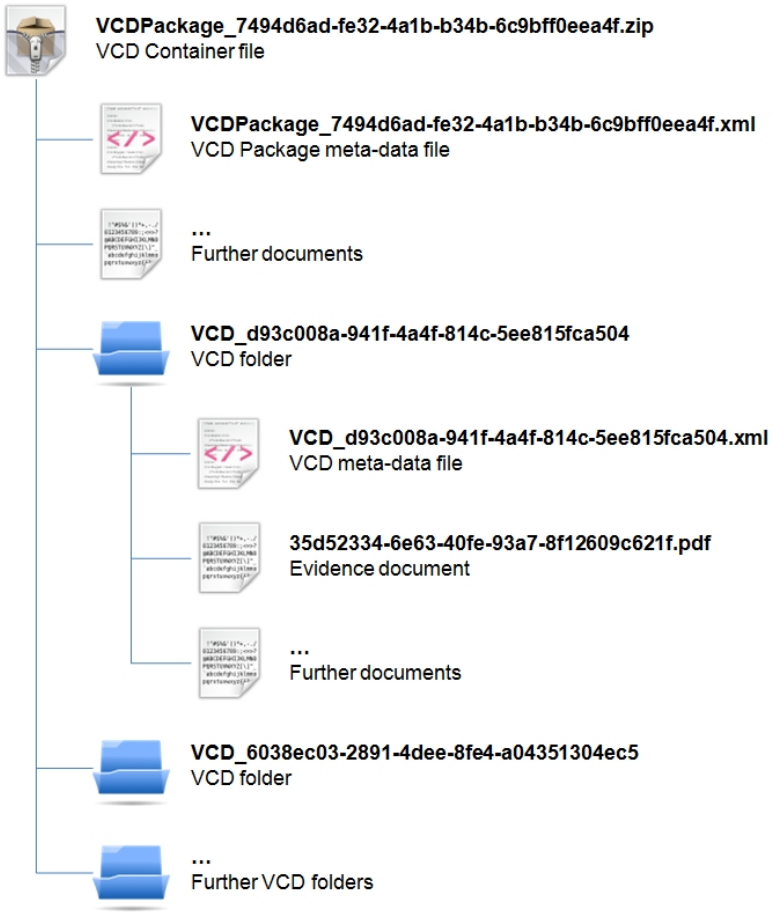


***Figure 1: Example folder structure of a VCD Container***

The important thing to notice here is that there is no VCD Package folder in a VCD Container zip file. The figure above can be transferred to an extracted container (i.e. a VCD package folder) with the root element being a folder with the same name as the VCD Container.

[1] Note that the name of the VCD Container Zip file may be different; it can be freely chosen by the user.

## Naming patterns and conventions

Table 3 defines the naming patterns of all covered resources.

| Item | Naming pattern |
|---|---|
| **VCD Container**[2] | Pattern: VCD Container_<packageUuid>.zip<br>Example: VCD Container_7494d6ad-fe32-4a1b-b34b-6c9bff0eea4f.zip |
| **VCD Package (Folder)**[3] | Pattern: VCDPackage_<packageUuid><br>Example: VCDPackage_7494d6ad-fe32-4a1b-b34b-6c9bff0eea4f |
| **VCD Package (XML)** | Pattern: VCDPackage_<packageUuid>.xml<br>Example: VCDPackage_7494d6ad-fe32-4a1b-b34b-6c9bff0eea4f.xml |
| **VCD (Folder)** | Pattern: VCD_<vcdUuid><br>Example: VCD_d93c008a-941f-4a4f-814c-5ee815fca504 |
| **VCD (XML)** | Pattern: VCD_<vcdUuid>.xml<br>Example: VCD_d93c008a-941f-4a4f-814c-5ee815fca504.xml |
| **Physical file names** | Pattern: <fileUuid>.<fileExtension><br>Example: 35d52334-6e63-40fe-93a7-8f12609c621f.pdf |
| **URI** | Pattern: urn:eu:peppol:vcd:document:<fileUuid>.<fileExtension><br>Example: urn:eu:peppol:vcd:document: 35d52334-6e63-40fe-93a7-8f12609c621f.pdf |

***Table 3: Naming patterns***

**Kommentar [DR1]:** Or "urn:x-eu:peppol:vcd:document…" because it is an unqualified uri?

As defined in [3], UUIDs are generated according to http://tools.ietf.org/html/rfc4122 using UUID version 4 (random generated UUID).

[2] It should be possible to allow the user to rename the zip file, for example when storing it on his local computer or sending it by e-mail. This rather cryptic file name of a container zip file seems inappropriate for such purposes. VCD Software applications should be able to treat any name of the zip file.

[3] According to [1)] and the name of the container zip file, the name of the folder is the same as the container zip file. However, changes to this name are rather unlikely, because a container zip file is usually extracted only by VCD software applications, and not by users. The VCD Package folder only occurs when an application is building or extracting a VCD Package.

## Referencing VCDs inside a VCD Package (XML)

Each VCD that is included in a VCD Container is referenced inside the VCD Package meta-data file. The following xml fragment illustrates this:

```
<vcdp:VCDPackage>

      […]

      <cac:BiddingConsortium>

            <cac:LeaderSingleTenderer>
                  <cbc:VCDReferenceID>
                        d93c008a-941f-4a4f-814c-5ee815fca504
                  </cbc:VCDReferenceID>
            </cac:LeaderSingleTenderer>

            <cac:OtherMemberSingleTenderer>
                  <cbc:VCDReferenceID>
                        6038ec03-2891-4dee-8fe4-a04351304ec5
                  </cbc:VCDReferenceID>
            </cac:OtherMemberSingleTenderer>

            […]

      </cac:BiddingConsortium>

      […]

</vcdp:VCDPackage>
```

The names of the VCD folder and the VCD meta-data file can be derived from this information. In the above example, the VCD of the leading tenderer will be stored in the folder “VCD_d93c008a-941f-4a4f-814c-5ee815fca504” that contains the meta-data file “VCD_d93c008a-941f-4a4f-814c-5ee815fca504.xml”. Accordingly, the VCD and the meta-data file of the other tenderer will be stored in the folder named “6038ec03-2891-4dee-8fe4-a04351304ec5”.

## Document references

Files inside a VCD Package (Folder) or VCD (Folder) are referenced by the cac:DocumentReference element in the corresponding meta-data files.

During generation of a VCD and the inclusion of files, the following values must be created and stored:

- A UUID for the file object, e.g. 35d52334-6e63-40fe-93a7-8f12609c621f
- A new physical file name, e.g. 35d52334-6e63-40fe-93a7-8f12609c621f.pdf
- The original file name, e.g. Strafregisterauszug für Max Mustermann.pdf

- A URI for the file object, e.g. urn:eu:peppol:vcd:document:35d52334-6e63-40fe-93a7-8f12609c621f.pdf

The physical file will be stored in the VCD folder using the generated new physical file name (<fileUuid>.<fileExtension>). This (physical) file name, the URI and the original file name will be stored in the VCD meta-data file in the elements cbc:FileName, cbc:URI and cbc:Description. Cbc:URI and cbc:FileName are mandatory elements, cbc:Description is optional in case a more precise file name or description of the attached file is required. An example XML fragment of the meta-data files will look like:

```
<cac:DocumentReference>
      <cbc:ID>35d52334-6e63-40fe-93a7-8f12609c621f</cbc:ID>

      <cac:Attachment>
            <cac:ExternalReference>

                  […]
                  <cbc:URI>
                        urn:eu:peppol:vcd:document:35d52334-6e63-40fe-
                        93a7-8f12609c621f.pdf
                  </cbc:URI>

                  <cbc:FileName>
                        35d52334-6e63-40fe-93a7-8f12609c621f.pdf
                  </cbc:FileName>

                  <cbc:Description languageID=”DE”>
                        Strafregisterauszug für Max Mustermann.pdf
                  </cbc:Description>

                  <cbc:Description languageID=”EN”>
                        Criminal Record for Max Mustermann.pdf
                  </cbc:Description>

            </cac:ExternalReference>
      </cac:Attachment>

</cac:DocumentReference>
```

cbc:Description should be a TextType field with cardinality 0..n in order to support multiple languages (see attribute “languageID”), which is an important point, if this field is needed for presentation or export purposes.

## Internal references

The elements cac:Criterion and cac:Evidence furthermore require internal references inside a single VCD meta-data file. This is needed in order to maintain the inter-connection between a criterion and an evidence document that has been provided by an economic operator to prove the criterion.

This is achieved by the element cac:Criterion/cbc: ProvingEvidenceID that points at cac:Evidence/cbc: DocumentGroupID, i.e. it contains the same value. Vice versa, cac:Evidence/cbc:ProvesCriterionID contains the same value as cac:Criterion/cbc:ID.

As defined in [3], both cac:Criterion/cbc:ID and cac:Evidence/cbc: DocumentGroupID are UUIDs generated according to http://tools.ietf.org/html/rfc4122 using UUID version 4 (random generated UUID). The following xml fragment illustrates the aforementioned cross-reference:

```
<cac:Criterion>
      […]
      <cbc:ID>
            38432acc-093d-45be-b505-a86fa16fb55b
      </cbc:ID>
      <cbc:ProvingEvidenceID>
            bf73bb6e-3aa7-471a-8f1a-009f6aa32ecc
      </cbc:ProvingEvidenceID>
      […]
</cac:Criterion>

<cac:Evidence>
      […]
      <cbc:DocumentGroupID>
            bf73bb6e-3aa7-471a-8f1a-009f6aa32ecc
      </cbc:DocumentGroupID>
      <cbc:ProvesCriterionID>
            38432acc-093d-45be-b505-a86fa16fb55b
      </cbc:ProvesCriterionID>
      […]
</cac:Evidence>
```