

H5P_GET_VIRTUAL_FILENAME

[Expand all](#) [Collapse all](#)

- [Jump to ...](#)
- [Summary](#)
- [Description](#)
- [Example](#)
- [Switch language ...](#)
- [C](#)
- [C++](#)
- [FORTRAN](#)
- [JAVA](#)

[Summary](#)
[Description](#)
[Example](#)
[JAVA](#)
[FORTRAN](#)
[C++](#)
[C](#)

H5P_GET_VIRTUAL_FILENAME

Gets the filename of a source dataset used in the mapping

Procedure:

H5P_GET_VIRTUAL_FILENAME (dcpl_id, index, name, size)

Signature:

```
ssize_t H5Pget_virtual_filename(  
    hid_t dcpl_id,  
    size_t index,  
    char *name,  
    size_t size  
)
```

Fortran Interface: `h5pget_virtual_filename_f`

Signature:

```
SUBROUTINE h5pget_virtual_filename_f(dcpl_id, index, name, hdferr, name_len)
  INTEGER(HID_T) , INTENT(IN)  :: dcpl_id
  INTEGER(SIZE_T) , INTENT(IN)  :: index
  CHARACTER(LEN=*) , INTENT(OUT) :: name
  INTEGER, INTENT(OUT)          :: hdferr
  INTEGER(SIZE_T), OPTIONAL     :: name_len
```

Optional parameters:

`name_len` - The size of name needed to hold the filename. (OUT)

Inputs:

`dcpl_id` - The identifier of the virtual dataset creation property list.
`index` - Mapping index.
The value of `index` is 0 (zero) or greater and less than `count` ($0 < \text{index} < \text{count}$), where `count` is the number of mappings returned by `h5pget_virtual_count`.

Outputs:

`name` - A buffer containing the name of the file containing the source dataset.
`hdferr` - Returns 0 if successful and -1 if fails.

Parameters:

<code>hid_t dcpl_id</code>	IN: The identifier of the virtual dataset creation property list
<code>size_t index</code>	IN: Mapping index The value of <code>index</code> is 0 (zero) or greater and less than <code>count</code> ($0 < \text{index} < \text{count}$), where <code>count</code> is the number of mappings returned by <code>H5P_GET_VIRTUAL_COUNT</code> .
<code>char *name</code>	OUT: A buffer containing the name of the file containing the source dataset
<code>size_t size</code>	IN: The size, in bytes, of the name buffer. Must be the size of the filename in bytes plus 1 for a NULL terminator

Description:

`H5P_GET_VIRTUAL_FILENAME` takes the dataset creation property list for the virtual dataset, `dcpl_id`, the mapping index, `index`, the size of the filename for a source dataset, `size`, and retrieves the name of the file for a source dataset used in the mapping.

Up to `size` characters of the filename are returned in `name`; additional characters, if any, are not returned to the user application.

If the length of the filename, which determines the required value of `size`, is unknown, a preliminary call to `H5P_GET_VIRTUAL_FILENAME` with the last two parameters set to `NULL` and zero respectively can be made. The return value of this call will be the size in bytes of the filename. That value, plus 1 for a `NULL` terminator, must then be assigned to `size` for a second `H5P_GET_VIRTUAL_FILENAME` call, which will retrieve the actual filename.

See Also:

[Virtual Dataset Overview](#)

Supporting Functions:

- `H5P_SET_LAYOUT`
- `H5P_GET_LAYOUT`
- `H5S_IS_REGULAR_HYPERSLAB`
- `H5S_GET_REGULAR_HYPERSLAB`
- `H5S_SELECT_HYPERSLAB`

VDS Functions:

- H5P_GET_VIRTUAL_COUNT
- H5P_GET_VIRTUAL_DSETNAME
- H5P_GET_VIRTUAL_FILENAME
- H5P_GET_VIRTUAL_PREFIX
- H5P_GET_VIRTUAL_PRINTF_GAP
- H5P_GET_VIRTUAL_SRCSPACE
- H5P_GET_VIRTUAL_VIEW
- H5P_GET_VIRTUAL_VSPACE
- H5P_SET_VIRTUAL
- H5P_SET_VIRTUAL_PREFIX
- H5P_SET_VIRTUAL_PRINTF_GAP
- H5P_SET_VIRTUAL_VIEW

Returns:

Returns the length of the filename if successful; otherwise returns a negative value.

Example:

examples / h5_vds.c [213:217]

1.10/master

HDF5V/hdf5

```
/* Get source file name. */
len = H5Pget_virtual_filename (dcpl, (size_t)i, NULL, 0);
filename = (char *)malloc((size_t)len*sizeof(char)+1);
H5Pget_virtual_filename (dcpl, (size_t)i, filename, len+1);
printf("          Source filename %s\n", filename);
```

History:

Release	Change
1.10.0	C function introduced with this release.