

H5D_CREATE2

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

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

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

H5D_CREATE2

Creates a new dataset and links it into the file

Procedure:

H5D_CREATE2(loc_id, name, dtype_id, space_id, lcpl_id, dcpl_id, dapl_id)

Signature:

```
hid_t H5Dcreate2( hid_t loc_id, const char *name, hid_t dtype_id, hid_t space_id, hid_t lcpl_id, hid_t dcpl_id, hid_t dapl_id )
```

```

SUBROUTINE h5dcreate_f(loc_id, name, type_id, space_id, dset_id, &
    hdferr, dcpl_id, lcpl_id, dapl_id)

    IMPLICIT NONE
    INTEGER(HID_T), INTENT(IN) :: loc_id    ! File or group identifier
    CHARACTER(LEN=*), INTENT(IN) :: name    ! Name of the dataset
    INTEGER(HID_T), INTENT(IN) :: type_id   ! Datatype identifier
    INTEGER(HID_T), INTENT(IN) :: space_id  ! Dataspace identifier
    INTEGER(HID_T), INTENT(OUT) :: dset_id  ! Dataset identifier
    INTEGER, INTENT(OUT) :: hdferr          ! Error code
                                           ! 0 on success and -1 on failure
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: dcpl_id
                                           ! Dataset creation property list
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: lcpl_id
                                           ! Link creation property list
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: dapl_id
                                           ! Dataset access property list
END SUBROUTINE h5dcreate_f

```

Parameters:

<i>hid_t</i> loc_id	IN: Location identifier; may be a file, group, dataset, named datatype, or attribute
<i>const char</i> *name	IN: Dataset name
<i>hid_t</i> dtype_id	IN: Datatype identifier
<i>hid_t</i> space_id	IN: Dataspace identifier
<i>hid_t</i> lcpl_id	IN: Link creation property list
<i>hid_t</i> dcpl_id	IN: Dataset creation property list
<i>hid_t</i> dapl_id	IN: Dataset access property list

Description:

H5D_CREATE2 creates a new dataset named *name* at the location specified by *loc_id*, and associates constant and initial persistent properties with that dataset, including *dtype_id*, the datatype of each data element as stored in the file; *space_id*, the dataspace of the dataset; and other initial properties as defined in the dataset creation property and access property lists, *dcpl_id* and *dapl_id*, respectively. Once created, the dataset is opened for access.

loc_id may be a file, group, dataset, named datatype or attribute. If an attribute, dataset, or named datatype is specified for *loc_id* then the dataset will be created at the location where the attribute, dataset, or named datatype is attached. *name* may be either an absolute path in the file or a relative path from *loc_id* naming the dataset.

If *dtype_id* is either a fixed-length or variable-length string, it is important to set the string length when defining the datatype. String datatypes are derived from H5T_C_S1 (or H5T_FORTRAN_S1 for Fortran codes), which defaults to 1 character in size. See [H5T_SET_SIZE](#) and [Creating variable-length string datatypes](#).

If *dtype_id* is a committed datatype, and if the file location associated with the committed datatype is different from the file location where the dataset will be created, the datatype is copied and converted to a transient type.

The link creation property list, *lcpl_id*, governs creation of the link(s) by which the new dataset is accessed and the creation of any intermediate groups that may be missing.

The datatype and dataspace properties and the dataset creation and access property lists are attached to the dataset, so the caller may derive new datatypes, dataspace, and creation and access properties from the old ones and reuse them in calls to create additional datasets.

Once created, the dataset is ready to receive raw data. Immediately attempting to read raw data from the dataset will probably return the fill value.

To conserve and release resources, the dataset should be closed when access is no longer required.

Returns:

Returns a dataset identifier if successful; otherwise returns a negative value.

Example:

examples / h5_subset.c [64:65]

1.10/master H5FFV/hdf5

```
dataset_id = H5Dcreate2 (file_id, DATASETNAME, H5T_STD_I32BE, dataspace_id,
                        H5P_DEFAULT, H5P_DEFAULT, H5P_DEFAULT);
```

fortran / examples / compound.f90 [25:35]

1.10/master H5FFV/hd

f5

```
PROGRAM COMPOUNDEXAMPLE

  USE HDF5 ! This module contains all necessary modules

  IMPLICIT NONE

  CHARACTER(LEN=11), PARAMETER :: filename = "compound.h5" ! File name
  CHARACTER(LEN=8), PARAMETER :: dsetname = "Compound"      ! Dataset name
  INTEGER, PARAMETER :: dimsize = 6 ! Size of the dataset

  INTEGER(HID_T) :: file_id      ! File identifier
```

History:

Release	Change
1.8.0	C function introduced in this release.