

H5T_UNREGISTER

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

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

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

H5T_UNREGISTER

Removes a conversion function.

Procedure:

H5T_UNREGISTER(*type*, *name*, *src_id*, *dst_id*, *func*)

Signature:

```
herr_t H5Tunregister( H5T_pers_t type, const char *name, hid_t src_id, hid_t dst_id, H5T_conv_t func )
```

Parameters:

<i>H5T_pers_t</i> <i>type</i>	IN: Conversion function type: H5T_PERS_HARD for hard conversion functions or H5T_PERS_SOFT for soft conversion functions.
<i>const char *</i> <i>name</i>	IN: Name displayed in diagnostic output.
<i>hid_t</i> <i>src_id</i>	IN: Identifier of source datatype.
<i>hid_t</i> <i>dst_id</i>	IN: Identifier of destination datatype.
<i>H5T_conv_t</i> <i>func</i>	IN: Function to convert between source and destination datatypes.

Description:

`H5Tunregister` removes a conversion function matching criteria such as soft or hard conversion, source and destination types, and the conversion function.

If a user is trying to remove a conversion function he registered, all parameters can be used. If he is trying to remove a library's default conversion function, there is no guarantee the `name` and `func` parameters will match the user's chosen values. Passing in some values may cause this function to fail. A good practice is to pass in `NULL` as their values.

All parameters are optional. The missing parameters will be used to generalize the search criteria.

The conversion function pointer type declaration is described in `H5Tregister`.

Returns:

Returns a non-negative value if successful; otherwise returns a negative value.

Example:

Coming Soon!

History:

Release	C
1.6.3	The following change occurred in the <code>H5Tconv_t</code> function: <code>nelmts</code> parameter type changed to <code>size_t</code> .

--- Last Modified: January 24, 2018 | 11:02 AM