

DC plugin SDK (C)

Generated by Doxygen 1.8.4

Fri Jul 26 2013 15:26:26

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	DCUI Struct Reference	5
3.1.1	Detailed Description	6
3.1.2	Member Function Documentation	6
3.1.2.1	void	6
3.1.2.2	void	6
3.1.2.3	void	6
3.1.2.4	void	6
3.1.3	Member Data Documentation	6
3.1.3.1	apiVersion	6
3.1.3.2	command	6
3.1.3.3	icon	6
3.1.3.4	message	6
3.1.3.5	name	6
3.2	tagCommandData Struct Reference	6
3.2.1	Detailed Description	7
3.2.2	Member Data Documentation	7
3.2.2.1	command	7
3.2.2.2	params	7
3.3	tagConfigBool Struct Reference	7
3.3.1	Detailed Description	8
3.3.2	Member Data Documentation	8
3.3.2.1	type	8
3.3.2.2	value	8
3.4	tagConfigInt Struct Reference	8
3.4.1	Detailed Description	9

3.4.2	Member Data Documentation	9
3.4.2.1	type	9
3.4.2.2	value	9
3.5	tagConfigInt64 Struct Reference	9
3.5.1	Detailed Description	10
3.5.2	Member Data Documentation	10
3.5.2.1	type	10
3.5.2.2	value	10
3.6	tagConfigStr Struct Reference	10
3.6.1	Detailed Description	11
3.6.2	Member Data Documentation	11
3.6.2.1	type	11
3.6.2.2	value	11
3.7	tagConfigValue Struct Reference	11
3.7.1	Detailed Description	12
3.7.2	Member Data Documentation	12
3.7.2.1	type	12
3.8	tagConnectionData Struct Reference	12
3.8.1	Detailed Description	13
3.8.2	Member Data Documentation	13
3.8.2.1	ip	13
3.8.2.2	isManaged	13
3.8.2.3	isOp	13
3.8.2.4	isSecure	13
3.8.2.5	object	13
3.8.2.6	port	13
3.8.2.7	protocol	13
3.9	tagDCConfig Struct Reference	13
3.9.1	Detailed Description	14
3.9.2	Member Function Documentation	14
3.9.2.1	ConfigStrPtr	14
3.9.2.2	ConfigStrPtr	14
3.9.2.3	ConfigStrPtr	14
3.9.2.4	ConfigValuePtr	14
3.9.2.5	ConfigValuePtr	15
3.9.2.6	void	15
3.9.2.7	void	15
3.9.3	Member Data Documentation	15
3.9.3.1	apiVersion	15
3.9.3.2	setting	15

3.9.3.3	type	15
3.9.3.4	val	15
3.10	tagDCConnection Struct Reference	15
3.10.1	Detailed Description	16
3.10.2	Member Function Documentation	16
3.10.2.1	UserDataPtr	16
3.10.2.2	void	16
3.10.2.3	void	16
3.10.2.4	void	16
3.10.3	Member Data Documentation	16
3.10.3.1	apiVersion	16
3.10.3.2	cmd	16
3.10.3.3	data	16
3.10.3.4	graceless	16
3.10.3.5	n	16
3.10.3.6	port	16
3.11	tagDCCore Struct Reference	17
3.11.1	Detailed Description	17
3.11.2	Member Function Documentation	17
3.11.2.1	Bool	17
3.11.2.2	Bool	17
3.11.2.3	DCInterfacePtr	17
3.11.2.4	host_name	17
3.11.2.5	intfHandle	17
3.11.3	Member Data Documentation	17
3.11.3.1	apiVersion	18
3.11.3.2	pInterface	18
3.11.3.3	version	18
3.12	tagDCHooks Struct Reference	18
3.12.1	Detailed Description	19
3.12.2	Member Function Documentation	19
3.12.2.1	Bool	19
3.12.2.2	Bool	19
3.12.2.3	hookHandle	19
3.12.2.4	size_t	19
3.12.2.5	subsHandle	19
3.12.3	Member Data Documentation	19
3.12.3.1	apiVersion	19
3.12.3.2	defProc	19
3.12.3.3	hookProc	19

3.12.3.4	pCommon	19
3.12.3.5	pData	19
3.12.3.6	pObject	19
3.13	tagDCHub Struct Reference	19
3.13.1	Detailed Description	20
3.13.2	Member Function Documentation	20
3.13.2.1	Bool	20
3.13.2.2	HubDataPtr	20
3.13.2.3	HubDataPtr	20
3.13.2.4	HubDataPtr	21
3.13.2.5	UserDataPtr	21
3.13.2.6	UserDataPtr	21
3.13.2.7	void	21
3.13.2.8	void	21
3.13.2.9	void	21
3.13.2.10	void	21
3.13.2.11	void	21
3.13.2.12	void	21
3.13.2.13	void	21
3.13.3	Member Data Documentation	21
3.13.3.1	apiVersion	21
3.13.3.2	cmd	21
3.13.3.3	hubUrl	21
3.13.3.4	msg	21
3.13.3.5	nick	21
3.13.3.6	password	21
3.13.3.7	thirdPerson	21
3.13.3.8	type	22
3.14	tagDCInterface Struct Reference	22
3.14.1	Detailed Description	22
3.14.2	Member Data Documentation	22
3.14.2.1	apiVersion	22
3.15	tagDCLog Struct Reference	22
3.15.1	Detailed Description	23
3.15.2	Member Function Documentation	23
3.15.2.1	void	23
3.15.3	Member Data Documentation	23
3.15.3.1	apiVersion	23
3.16	tagDCQueue Struct Reference	23
3.16.1	Detailed Description	24

3.16.2	Member Function Documentation	24
3.16.2.1	Bool	24
3.16.2.2	QueueDataPtr	24
3.16.2.3	QueueDataPtr	24
3.16.2.4	QueueDataPtr	24
3.16.2.5	QueueDataPtr	24
3.16.2.6	void	25
3.16.2.7	void	25
3.16.2.8	void	25
3.16.3	Member Data Documentation	25
3.16.3.1	apiVersion	25
3.16.3.2	priority	25
3.16.3.3	silent	25
3.16.3.4	size	25
3.16.3.5	target	25
3.17	tagDCTagger Struct Reference	25
3.17.1	Detailed Description	26
3.17.2	Member Function Documentation	26
3.17.2.1	get_text	26
3.17.2.2	void	26
3.17.2.3	void	26
3.17.3	Member Data Documentation	27
3.17.3.1	apiVersion	27
3.17.3.2	attributes	27
3.17.3.3	end	27
3.17.3.4	id	27
3.17.3.5	replacement	27
3.17.3.6	start	27
3.18	tagDCUtils Struct Reference	27
3.18.1	Detailed Description	28
3.18.2	Member Function Documentation	29
3.18.2.1	size_t	29
3.18.2.2	size_t	29
3.18.2.3	size_t	29
3.18.2.4	size_t	29
3.18.2.5	size_t	29
3.18.2.6	size_t	29
3.18.3	Member Data Documentation	29
3.18.3.1	apiVersion	29
3.18.3.2	n	29

3.18.3.3	n	29
3.18.3.4	n	29
3.18.3.5	src	29
3.18.3.6	src	29
3.18.3.7	src	29
3.19	tagHubData Struct Reference	29
3.19.1	Detailed Description	30
3.19.2	Member Data Documentation	30
3.19.2.1	ip	30
3.19.2.2	isManaged	30
3.19.2.3	isOp	30
3.19.2.4	isSecure	30
3.19.2.5	object	30
3.19.2.6	port	31
3.19.2.7	protocol	31
3.19.2.8	url	31
3.20	tagMetaData Struct Reference	31
3.20.1	Detailed Description	31
3.20.2	Member Data Documentation	32
3.20.2.1	apiVersion	32
3.20.2.2	author	32
3.20.2.3	dependencies	32
3.20.2.4	description	32
3.20.2.5	guid	32
3.20.2.6	name	32
3.20.2.7	numDependencies	32
3.20.2.8	version	32
3.20.2.9	web	32
3.21	tagQueueData Struct Reference	33
3.21.1	Detailed Description	33
3.21.2	Member Data Documentation	33
3.21.2.1	hash	33
3.21.2.2	isFileList	33
3.21.2.3	isManaged	33
3.21.2.4	location	34
3.21.2.5	object	34
3.21.2.6	size	34
3.21.2.7	target	34
3.22	tagStringData Struct Reference	34
3.22.1	Detailed Description	34

3.22.2	Member Data Documentation	34
3.22.2.1	in	34
3.22.2.2	out	35
3.23	tagTagData Struct Reference	35
3.23.1	Detailed Description	35
3.23.2	Member Data Documentation	35
3.23.2.1	isManaged	35
3.23.2.2	object	35
3.24	tagUDPData Struct Reference	35
3.24.1	Detailed Description	36
3.24.2	Member Data Documentation	36
3.24.2.1	ip	36
3.24.2.2	port	36
3.25	tagUserData Struct Reference	36
3.25.1	Detailed Description	37
3.25.2	Member Data Documentation	37
3.25.2.1	cid	37
3.25.2.2	hubHint	37
3.25.2.3	isManaged	37
3.25.2.4	isOp	37
3.25.2.5	nick	37
3.25.2.6	object	38
3.25.2.7	protocol	38
3.25.2.8	sid	38
4	File Documentation	39
4.1	Config.c File Reference	39
4.1.1	Function Documentation	40
4.1.1.1	free_core_cfg	40
4.1.1.2	get_cfg	40
4.1.1.3	get_cfg_int	40
4.1.1.4	get_cfg_int64	40
4.1.1.5	get_cfg_path	40
4.1.1.6	get_core_cfg	40
4.1.1.7	get_install_path	40
4.1.1.8	init_cfg	40
4.1.1.9	set_cfg	41
4.1.1.10	set_cfg_int	41
4.1.1.11	set_cfg_int64	41
4.1.1.12	uninit_cfg	41

4.1.2	Variable Documentation	41
4.1.2.1	config	41
4.1.2.2	guid	42
4.2	Config.h File Reference	42
4.2.1	Function Documentation	43
4.2.1.1	free_core_cfg	43
4.2.1.2	get_cfg	43
4.2.1.3	get_cfg_int	43
4.2.1.4	get_cfg_int64	43
4.2.1.5	get_cfg_path	43
4.2.1.6	get_core_cfg	43
4.2.1.7	get_install_path	43
4.2.1.8	init_cfg	43
4.2.1.9	set_cfg	44
4.2.1.10	set_cfg_int	44
4.2.1.11	set_cfg_int64	44
4.2.1.12	uninit_cfg	44
4.3	main.c File Reference	44
4.3.1	Function Documentation	45
4.3.1.1	pluginInit	45
4.4	Plugin.c File Reference	45
4.4.1	Macro Definition Documentation	46
4.4.1.1	HOOKS_SUBSCRIBED	46
4.4.2	Function Documentation	46
4.4.2.1	onLoad	46
4.4.2.2	onUnload	47
4.4.2.3	pluginMain	48
4.4.3	Variable Documentation	48
4.4.3.1	dcpp	48
4.4.3.2	hookFuncs	48
4.4.3.3	hookGuids	48
4.4.3.4	hooks	49
4.4.3.5	logging	49
4.4.3.6	subs	49
4.5	Plugin.h File Reference	49
4.5.1	Function Documentation	49
4.5.1.1	pluginMain	49
4.6	PluginDefs.h File Reference	50
4.6.1	Macro Definition Documentation	54
4.6.1.1	Bool	54

4.6.1.2	DCAPI	54
4.6.1.3	DCAPI_CORE_VER	54
4.6.1.4	DCEXP	54
4.6.1.5	DCIMP	54
4.6.1.6	DCINTF_CONFIG	54
4.6.1.7	DCINTF_CONFIG_VER	54
4.6.1.8	DCINTF_DCPP_CONNECTIONS	55
4.6.1.9	DCINTF_DCPP_CONNECTIONS_VER	55
4.6.1.10	DCINTF_DCPP_HUBS	55
4.6.1.11	DCINTF_DCPP_HUBS_VER	55
4.6.1.12	DCINTF_DCPP_QUEUE	55
4.6.1.13	DCINTF_DCPP_QUEUE_VER	55
4.6.1.14	DCINTF_DCPP_TAGGER	55
4.6.1.15	DCINTF_DCPP_TAGGER_VER	55
4.6.1.16	DCINTF_DCPP_UI	55
4.6.1.17	DCINTF_DCPP_UI_VER	55
4.6.1.18	DCINTF_DCPP_UTILS	55
4.6.1.19	DCINTF_DCPP_UTILS_VER	55
4.6.1.20	DCINTF_HOOKS	56
4.6.1.21	DCINTF_HOOKS_VER	56
4.6.1.22	DCINTF_LOGGING	56
4.6.1.23	DCINTF_LOGGING_VER	56
4.6.1.24	False	56
4.6.1.25	HOOK_CHAT_IN	56
4.6.1.26	HOOK_CHAT_OUT	56
4.6.1.27	HOOK_CHAT_PM_IN	56
4.6.1.28	HOOK_CHAT_PM_OUT	56
4.6.1.29	HOOK_HUB_OFFLINE	56
4.6.1.30	HOOK_HUB_ONLINE	56
4.6.1.31	HOOK_NETWORK_CONN_IN	57
4.6.1.32	HOOK_NETWORK_CONN_OUT	57
4.6.1.33	HOOK_NETWORK_HUB_IN	57
4.6.1.34	HOOK_NETWORK_HUB_OUT	57
4.6.1.35	HOOK_NETWORK_UDP_IN	57
4.6.1.36	HOOK_NETWORK_UDP_OUT	57
4.6.1.37	HOOK_QUEUE_ADDED	57
4.6.1.38	HOOK_QUEUE_FINISHED	57
4.6.1.39	HOOK_QUEUE_MOVED	57
4.6.1.40	HOOK_QUEUE_REMOVED	57
4.6.1.41	HOOK_TIMER_MINUTE	57

4.6.1.42	HOOK_TIMER_SECOND	58
4.6.1.43	HOOK_UI_CHAT_COMMAND	58
4.6.1.44	HOOK_UI_CHAT_COMMAND_PM	58
4.6.1.45	HOOK_UI_CHAT_DISPLAY	58
4.6.1.46	HOOK_UI_CHAT_TAGS	58
4.6.1.47	HOOK_UI_CREATED	58
4.6.1.48	HOOK_USER_OFFLINE	58
4.6.1.49	HOOK_USER_ONLINE	58
4.6.1.50	True	58
4.6.2	Typedef Documentation	58
4.6.2.1	bBreak	58
4.6.2.2	CommandData	59
4.6.2.3	CommandDataPtr	59
4.6.2.4	ConfigBool	59
4.6.2.5	ConfigBoolPtr	59
4.6.2.6	ConfigInt	59
4.6.2.7	ConfigInt64	59
4.6.2.8	ConfigInt64Ptr	59
4.6.2.9	ConfigIntPtr	59
4.6.2.10	ConfigStr	59
4.6.2.11	ConfigStrPtr	59
4.6.2.12	ConfigType	59
4.6.2.13	ConfigValue	59
4.6.2.14	ConfigValuePtr	59
4.6.2.15	ConnectionData	59
4.6.2.16	ConnectionDataPtr	59
4.6.2.17	core	59
4.6.2.18	dcBool	59
4.6.2.19	DCCConfig	59
4.6.2.20	DCCConfigPtr	59
4.6.2.21	DCCConnection	59
4.6.2.22	DCCConnectionPtr	59
4.6.2.23	DCCore	59
4.6.2.24	DCCorePtr	59
4.6.2.25	DCHooks	59
4.6.2.26	DCHooksPtr	59
4.6.2.27	DCHub	59
4.6.2.28	DCHubPtr	59
4.6.2.29	DCInterface	60
4.6.2.30	DCInterfacePtr	60

4.6.2.31	DCLog	60
4.6.2.32	DCLogPtr	60
4.6.2.33	dcptr_t	60
4.6.2.34	DCQueue	60
4.6.2.35	DCQueuePtr	60
4.6.2.36	DCTagger	60
4.6.2.37	DCTaggerPtr	60
4.6.2.38	dctime_t	60
4.6.2.39	DCUI	60
4.6.2.40	DCUIPtr	60
4.6.2.41	DCUtils	60
4.6.2.42	DCUtilsPtr	60
4.6.2.43	hookHandle	60
4.6.2.44	HubData	60
4.6.2.45	HubDataPtr	60
4.6.2.46	intfHandle	60
4.6.2.47	MetaData	60
4.6.2.48	MetaDataPtr	60
4.6.2.49	MsgType	60
4.6.2.50	PathType	60
4.6.2.51	pCommon	61
4.6.2.52	pData	61
4.6.2.53	PluginState	61
4.6.2.54	ProtocolType	61
4.6.2.55	QueueData	61
4.6.2.56	QueueDataPtr	61
4.6.2.57	QueuePrio	61
4.6.2.58	StringData	61
4.6.2.59	StringDataPtr	61
4.6.2.60	subsHandle	61
4.6.2.61	TagData	61
4.6.2.62	TagDataPtr	61
4.6.2.63	UDPData	61
4.6.2.64	UDPDataPtr	61
4.6.2.65	UserData	61
4.6.2.66	UserDataPtr	61
4.6.3	Enumeration Type Documentation	61
4.6.3.1	tagConfigType	61
4.6.3.2	tagDCBool	62
4.6.3.3	tagMsgType	62

4.6.3.4	tagPathType	62
4.6.3.5	tagPluginState	62
4.6.3.6	tagProtocolType	62
4.6.3.7	tagQueuePrio	63
4.6.4	Function Documentation	63
4.6.4.1	Bool	63
4.6.4.2	void	63
4.7	resource.h File Reference	63
4.8	stdafx.c File Reference	63
4.9	stdafx.h File Reference	63
4.9.1	Macro Definition Documentation	64
4.9.1.1	PLUGIN_VERSION	64
4.10	version.h File Reference	65
4.10.1	Macro Definition Documentation	65
4.10.1.1	PLUGIN_AUTHOR	65
4.10.1.2	PLUGIN_DESC	65
4.10.1.3	PLUGIN_GUID	65
4.10.1.4	PLUGIN_NAME	66
4.10.1.5	PLUGIN_VERSION_MAJOR	66
4.10.1.6	PLUGIN_VERSION_MINOR	66
4.10.1.7	PLUGIN_WEB	66

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

DCUI	5
tagCommandData	6
tagConfigBool	7
tagConfigInt	8
tagConfigInt64	9
tagConfigStr	10
tagConfigValue	11
tagConnectionData	12
tagDCConfig	13
tagDCConnection	15
tagDCCore	17
tagDCHooks	18
tagDCHub	19
tagDCInterface	22
tagDCLog	22
tagDCQueue	23
tagDCTagger	25
tagDCUtils	27
tagHubData	29
tagMetaData	31
tagQueueData	33
tagStringData	34
tagTagData	35
tagUDPData	35
tagUserData	36

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Config.c	39
Config.h	42
main.c	44
Plugin.c	45
Plugin.h	49
PluginDefs.h	50
resource.h	63
stdafx.c	63
stdafx.h	63
version.h	65

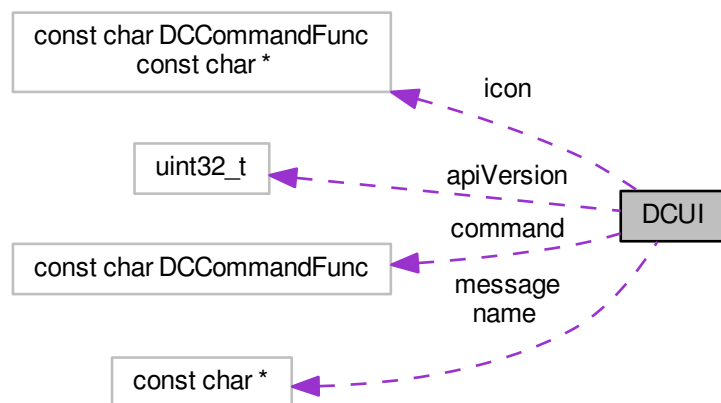
Chapter 3

Class Documentation

3.1 DCUI Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for DCUI:



Public Member Functions

- `void (DCAPI *add_command)(const char *guid`
- `void (DCAPI *remove_command)(const char *guid`
- `void (DCAPI *play_sound)(const char *path)`
- `void (DCAPI *notify)(const char *title`

Public Attributes

- `uint32_t apiVersion`
- `const char DCCommandFunc command`
- `const char DCCommandFunc const char * icon`

- const char * [message](#)
- const char * [name](#)

3.1.1 Detailed Description

Definition at line 448 of file PluginDefs.h.

3.1.2 Member Function Documentation

3.1.2.1 DCUI::void (DCAPI * *add_command*) const

3.1.2.2 DCUI::void (DCAPI * *remove_command*) const

3.1.2.3 DCUI::void (DCAPI * *play_sound*) const

3.1.2.4 DCUI::void (DCAPI * *notify*) const

3.1.3 Member Data Documentation

3.1.3.1 uint32_t DCUI::apiVersion

Definition at line 450 of file PluginDefs.h.

3.1.3.2 const char DCCCommandFunc DCUI::command

Definition at line 454 of file PluginDefs.h.

3.1.3.3 const char DCCCommandFunc const char* DCUI::icon

Definition at line 454 of file PluginDefs.h.

3.1.3.4 const char* DCUI::message

Definition at line 459 of file PluginDefs.h.

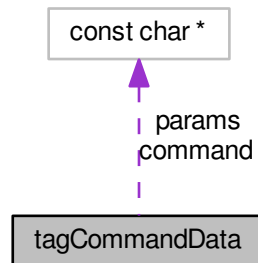
3.1.3.5 const char * DCUI::name

Definition at line 454 of file PluginDefs.h.

3.2 tagCommandData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagCommandData:



Public Attributes

- const char * [command](#)
- const char * [params](#)

3.2.1 Detailed Description

Definition at line 212 of file PluginDefs.h.

3.2.2 Member Data Documentation

3.2.2.1 const char* tagCommandData::command

Definition at line 213 of file PluginDefs.h.

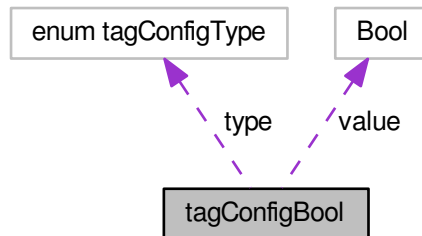
3.2.2.2 const char* tagCommandData::params

Definition at line 214 of file PluginDefs.h.

3.3 tagConfigBool Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagConfigBool:



Public Attributes

- [ConfigType type](#)
- [Bool value](#)

3.3.1 Detailed Description

Definition at line 194 of file PluginDefs.h.

3.3.2 Member Data Documentation

3.3.2.1 ConfigType tagConfigBool::type

Definition at line 195 of file PluginDefs.h.

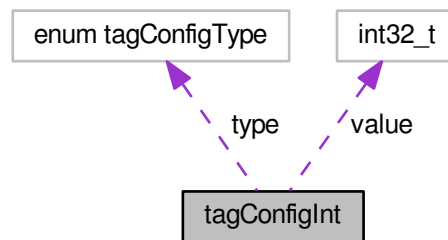
3.3.2.2 Bool tagConfigBool::value

Definition at line 196 of file PluginDefs.h.

3.4 tagConfigInt Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagConfigInt:



Public Attributes

- [ConfigType type](#)
- [int32_t value](#)

3.4.1 Detailed Description

Definition at line 188 of file `PluginDefs.h`.

3.4.2 Member Data Documentation

3.4.2.1 ConfigType tagConfigInt::type

Definition at line 189 of file `PluginDefs.h`.

Referenced by `set_cfg_int()`.

3.4.2.2 int32_t tagConfigInt::value

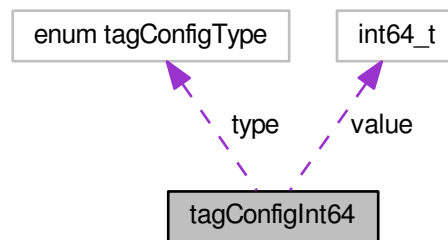
Definition at line 190 of file `PluginDefs.h`.

Referenced by `get_cfg_int()`, and `set_cfg_int()`.

3.5 tagConfigInt64 Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagConfigInt64:



Public Attributes

- [ConfigType type](#)
- [int64_t value](#)

3.5.1 Detailed Description

Definition at line 200 of file PluginDefs.h.

3.5.2 Member Data Documentation

3.5.2.1 ConfigType tagConfigInt64::type

Definition at line 201 of file PluginDefs.h.

Referenced by `set_cfg_int64()`.

3.5.2.2 int64_t tagConfigInt64::value

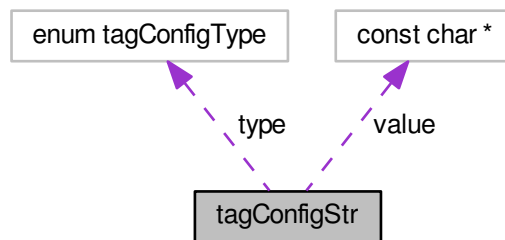
Definition at line 202 of file PluginDefs.h.

Referenced by `get_cfg_int64()`, and `set_cfg_int64()`.

3.6 tagConfigStr Struct Reference

```
#include <PluginDefs.h>
```


Collaboration diagram for tagConfigStr:



Public Attributes

- [ConfigType](#) type
- `const char *` [value](#)

3.6.1 Detailed Description

Definition at line 182 of file `PluginDefs.h`.

3.6.2 Member Data Documentation

3.6.2.1 ConfigType tagConfigStr::type

Definition at line 183 of file `PluginDefs.h`.

Referenced by `set_cfg()`.

3.6.2.2 const char* tagConfigStr::value

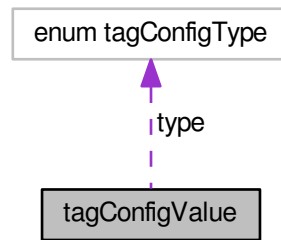
Definition at line 184 of file `PluginDefs.h`.

Referenced by `get_cfg()`, `get_cfg_path()`, `get_install_path()`, and `set_cfg()`.

3.7 tagConfigValue Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagConfigValue:



Public Attributes

- [ConfigType type](#)

3.7.1 Detailed Description

Definition at line 177 of file PluginDefs.h.

3.7.2 Member Data Documentation

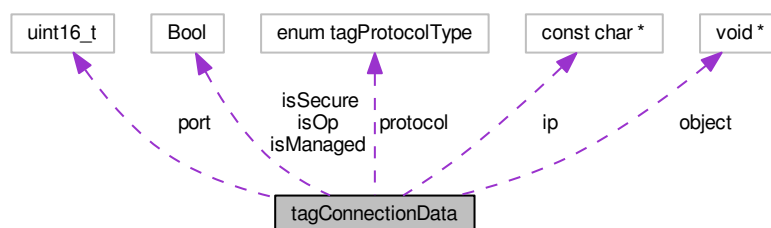
3.7.2.1 ConfigType tagConfigValue::type

Definition at line 178 of file PluginDefs.h.

3.8 tagConnectionData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagConnectionData:



Public Attributes

- `const char * ip`
- `Bool isManaged`
- `Bool isOp`
- `Bool isSecure`
- `dcptr_t object`
- `uint16_t port`
- `ProtocolType protocol`

3.8.1 Detailed Description

Definition at line 242 of file PluginDefs.h.

3.8.2 Member Data Documentation

3.8.2.1 `const char* tagConnectionData::ip`

Definition at line 243 of file PluginDefs.h.

3.8.2.2 `Bool tagConnectionData::isManaged`

Definition at line 249 of file PluginDefs.h.

3.8.2.3 `Bool tagConnectionData::isOp`

Definition at line 247 of file PluginDefs.h.

3.8.2.4 `Bool tagConnectionData::isSecure`

Definition at line 248 of file PluginDefs.h.

3.8.2.5 `dcptr_t tagConnectionData::object`

Definition at line 244 of file PluginDefs.h.

3.8.2.6 `uint16_t tagConnectionData::port`

Definition at line 245 of file PluginDefs.h.

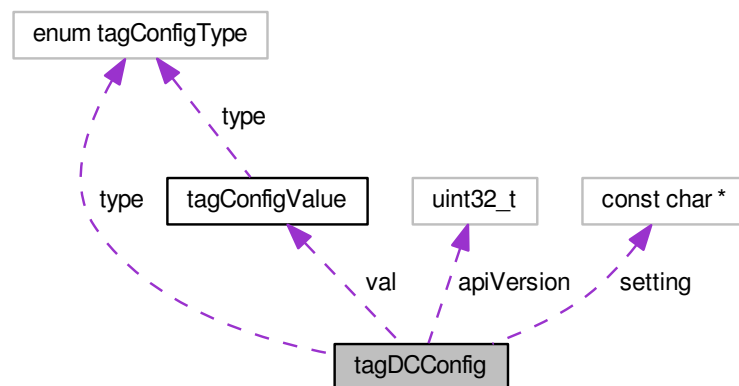
3.8.2.7 `ProtocolType tagConnectionData::protocol`

Definition at line 246 of file PluginDefs.h.

3.9 tagDCConfig Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCConfig:



Public Member Functions

- [ConfigStrPtr](#) (DCAPI *[get_path](#))([PathType](#) [type](#))
- [ConfigStrPtr](#) (DCAPI *[get_install_path](#))(const char *[guid](#))
- [ConfigStrPtr](#) (DCAPI *[get_language](#)())
- [ConfigValuePtr](#) (DCAPI *[get_cfg](#))(const char *[guid](#))
- [ConfigValuePtr](#) (DCAPI *[copy](#))(const [ConfigValuePtr](#) [val](#))
- void (DCAPI *[set_cfg](#))(const char *[guid](#))
- void (DCAPI *[release](#))([ConfigValuePtr](#) [val](#))

Public Attributes

- uint32_t [apiVersion](#)
- const char * [setting](#)
- const char [ConfigType](#) [type](#)
- const char [ConfigValuePtr](#) [val](#)

3.9.1 Detailed Description

Definition at line 337 of file `PluginDefs.h`.

3.9.2 Member Function Documentation

3.9.2.1 `tagDCConfig::ConfigStrPtr (DCAPI * get_path)`

3.9.2.2 `tagDCConfig::ConfigStrPtr (DCAPI * get_install_path) const`

3.9.2.3 `tagDCConfig::ConfigStrPtr (DCAPI * get_language)`

3.9.2.4 `tagDCConfig::ConfigValuePtr (DCAPI * get_cfg) const`

3.9.2.5 `tagDCConfig::ConfigValuePtr (DCAPI * copy) const`

3.9.2.6 `tagDCConfig::void (DCAPI * set_cfg) const`

3.9.2.7 `tagDCConfig::void (DCAPI * release)`

3.9.3 Member Data Documentation

3.9.3.1 `uint32_t tagDCConfig::apiVersion`

Definition at line 339 of file PluginDefs.h.

3.9.3.2 `const char * tagDCConfig::setting`

Definition at line 344 of file PluginDefs.h.

3.9.3.3 `const char ConfigType tagDCConfig::type`

Definition at line 345 of file PluginDefs.h.

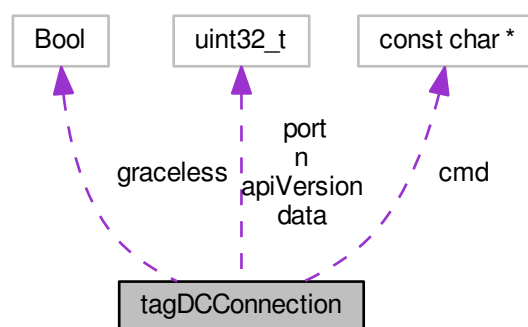
3.9.3.4 `const char ConfigValuePtr tagDCConfig::val`

Definition at line 344 of file PluginDefs.h.

3.10 tagDCConnection Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCConnection:



Public Member Functions

- `UserDataPtr (DCAPI *get_user)(ConnectionDataPtr hConn)`
- `void (DCAPI *send_udp_data)(const char *ip`
- `void (DCAPI *send_protocol_cmd)(ConnectionDataPtr hConn`

- `void (DCAPI *terminate_conn)(ConnectionDataPtr hConn`

Public Attributes

- `uint32_t apiVersion`
- `const char * cmd`
- `uint32_t dcptr_t data`
- `Bool graceless`
- `uint32_t dcptr_t size_t n`
- `uint32_t port`

3.10.1 Detailed Description

Definition at line 365 of file PluginDefs.h.

3.10.2 Member Function Documentation

3.10.2.1 `tagDCConnection::UserDataPtr (DCAPI * get_user)`

3.10.2.2 `tagDCConnection::void (DCAPI * send_udp_data) const`

3.10.2.3 `tagDCConnection::void (DCAPI * send_protocol_cmd)`

3.10.2.4 `tagDCConnection::void (DCAPI * terminate_conn)`

3.10.3 Member Data Documentation

3.10.3.1 `uint32_t tagDCConnection::apiVersion`

Definition at line 367 of file PluginDefs.h.

3.10.3.2 `const char* tagDCConnection::cmd`

Definition at line 370 of file PluginDefs.h.

3.10.3.3 `uint32_t dcptr_t tagDCConnection::data`

Definition at line 369 of file PluginDefs.h.

3.10.3.4 `Bool tagDCConnection::graceless`

Definition at line 371 of file PluginDefs.h.

3.10.3.5 `uint32_t dcptr_t size_t tagDCConnection::n`

Definition at line 369 of file PluginDefs.h.

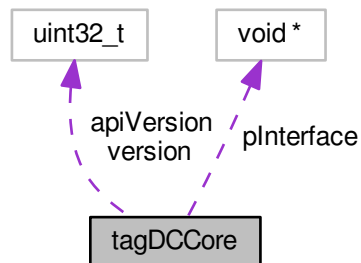
3.10.3.6 `uint32_t tagDCConnection::port`

Definition at line 369 of file PluginDefs.h.

3.11 tagDCCore Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCCore:



Public Member Functions

- `Bool (DCAPI *release_interface)(intfHandle hInterface)`
- `Bool (DCAPI *has_plugin)(const char *guid)`
- `DCInterfacePtr (DCAPI *query_interface)(const char *guid)`
- `const char *DCAPI * host_name ()`
- `intfHandle (DCAPI *register_interface)(const char *guid)`

Public Attributes

- `uint32_t apiVersion`
- `dcptr_t plInterface`
- `uint32_t version`

3.11.1 Detailed Description

Definition at line 295 of file `PluginDefs.h`.

3.11.2 Member Function Documentation

3.11.2.1 `tagDCCore::Bool (DCAPI * release_interface)`

3.11.2.2 `tagDCCore::Bool (DCAPI * has_plugin) const`

3.11.2.3 `tagDCCore::DCInterfacePtr (DCAPI * query_interface) const`

3.11.2.4 `const char* DCAPI* tagDCCore::host_name ()`

3.11.2.5 `tagDCCore::intfHandle (DCAPI * register_interface) const`

3.11.3 Member Data Documentation

3.11.3.1 uint32_t tagDCCore::apiVersion

Definition at line 297 of file PluginDefs.h.

3.11.3.2 dcptr_t tagDCCore::pInterface

Definition at line 300 of file PluginDefs.h.

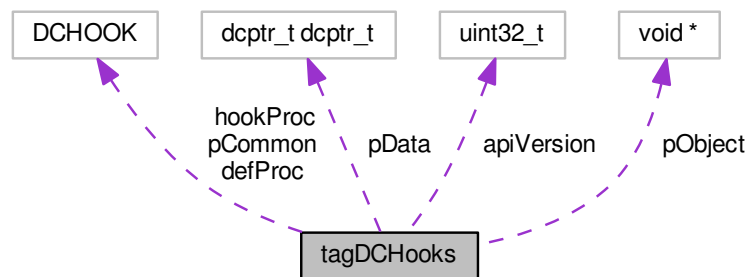
3.11.3.3 uint32_t tagDCCore::version

Definition at line 301 of file PluginDefs.h.

3.12 tagDCHooks Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCHooks:



Public Member Functions

- [Bool](#) ([DCAPI](#) *destroy_hook)([hookHandle](#) hHook)
- [Bool](#) ([DCAPI](#) *run_hook)([hookHandle](#) hHook)
- [hookHandle](#) ([DCAPI](#) *create_hook)(const char *guid)
- [size_t](#) ([DCAPI](#) *release_hook)([subsHandle](#) hHook)
- [subsHandle](#) ([DCAPI](#) *bind_hook)(const char *guid)

Public Attributes

- [uint32_t](#) [apiVersion](#)
- [DCHOOK](#) [defProc](#)
- [DCHOOK](#) [hookProc](#)
- [DCHOOK](#) [void *](#) [pCommon](#)
- [dcptr_t](#) [dcptr_t](#) [pData](#)
- [dcptr_t](#) [pObject](#)

3.12.1 Detailed Description

Definition at line 320 of file PluginDefs.h.

3.12.2 Member Function Documentation

3.12.2.1 tagDCHooks::Bool (DCAPI * *destroy_hook*)

3.12.2.2 tagDCHooks::Bool (DCAPI * *run_hook*)

3.12.2.3 tagDCHooks::hookHandle (DCAPI * *create_hook*) const

3.12.2.4 tagDCHooks::size_t (DCAPI * *release_hook*)

3.12.2.5 tagDCHooks::subsHandle (DCAPI * *bind_hook*) const

3.12.3 Member Data Documentation

3.12.3.1 uint32_t tagDCHooks::apiVersion

Definition at line 322 of file PluginDefs.h.

3.12.3.2 DCHOOK tagDCHooks::defProc

Definition at line 325 of file PluginDefs.h.

3.12.3.3 DCHOOK tagDCHooks::hookProc

Definition at line 329 of file PluginDefs.h.

3.12.3.4 DCHOOK void* tagDCHooks::pCommon

Definition at line 329 of file PluginDefs.h.

3.12.3.5 dcptr_t dcptr_t tagDCHooks::pData

Definition at line 330 of file PluginDefs.h.

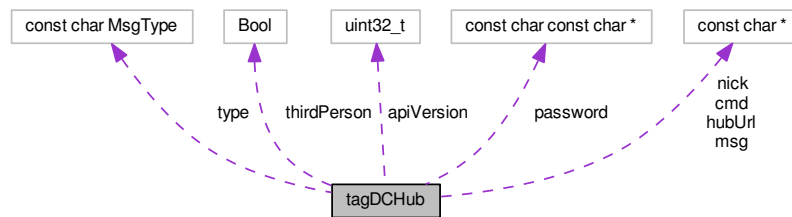
3.12.3.6 dcptr_t tagDCHooks::pObject

Definition at line 330 of file PluginDefs.h.

3.13 tagDCHub Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCHub:



Public Member Functions

- `Bool (DCAPI *send_private_message)(UserDataPtr hUser`
- `HubDataPtr (DCAPI *add_hub)(const char *url`
- `HubDataPtr (DCAPI *find_hub)(const char *url)`
- `HubDataPtr (DCAPI *copy)(const HubDataPtr hHub)`
- `UserDataPtr (DCAPI *find_user)(const char *cid`
- `UserDataPtr (DCAPI *copy_user)(const UserDataPtr hUser)`
- `void (DCAPI *remove_hub)(HubDataPtr hHub)`
- `void (DCAPI *emulate_protocol_cmd)(HubDataPtr hHub`
- `void (DCAPI *send_protocol_cmd)(HubDataPtr hHub`
- `void (DCAPI *send_message)(HubDataPtr hHub`
- `void (DCAPI *local_message)(HubDataPtr hHub`
- `void (DCAPI *release_user)(UserDataPtr hCopy)`
- `void (DCAPI *release)(HubDataPtr hCopy)`

Public Attributes

- `uint32_t apiVersion`
- `const char * cmd`
- `const char * hubUrl`
- `const char * msg`
- `const char * nick`
- `const char const char * password`
- `const char Bool thirdPerson`
- `const char MsgType type`

3.13.1 Detailed Description

Definition at line 377 of file PluginDefs.h.

3.13.2 Member Function Documentation

3.13.2.1 `tagDCHub::Bool (DCAPI * send_private_message)`

3.13.2.2 `tagDCHub::HubDataPtr (DCAPI * add_hub) const`

3.13.2.3 `tagDCHub::HubDataPtr (DCAPI * find_hub) const`

3.13.2.4 tagDCHub::HubDataPtr (DCAPI * *copy*) const

3.13.2.5 tagDCHub::UserDataPtr (DCAPI * *find_user*) const

3.13.2.6 tagDCHub::UserDataPtr (DCAPI * *copy_user*) const

3.13.2.7 tagDCHub::void (DCAPI * *remove_hub*)

3.13.2.8 tagDCHub::void (DCAPI * *emulate_protocol_cmd*)

3.13.2.9 tagDCHub::void (DCAPI * *send_protocol_cmd*)

3.13.2.10 tagDCHub::void (DCAPI * *send_message*)

3.13.2.11 tagDCHub::void (DCAPI * *local_message*)

3.13.2.12 tagDCHub::void (DCAPI * *release_user*)

3.13.2.13 tagDCHub::void (DCAPI * *release*)

3.13.3 Member Data Documentation

3.13.3.1 uint32_t tagDCHub::apiVersion

Definition at line 379 of file PluginDefs.h.

3.13.3.2 const char * tagDCHub::cmd

Definition at line 385 of file PluginDefs.h.

3.13.3.3 const char* tagDCHub::hubUrl

Definition at line 392 of file PluginDefs.h.

3.13.3.4 const char * tagDCHub::msg

Definition at line 388 of file PluginDefs.h.

3.13.3.5 const char* tagDCHub::nick

Definition at line 381 of file PluginDefs.h.

3.13.3.6 const char const char* tagDCHub::password

Definition at line 381 of file PluginDefs.h.

3.13.3.7 const char Bool tagDCHub::thirdPerson

Definition at line 388 of file PluginDefs.h.

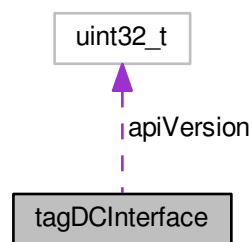
3.13.3.8 const char **MsgType** tagDCHub::type

Definition at line 389 of file PluginDefs.h.

3.14 tagDCInterface Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCInterface:



Public Attributes

- uint32_t [apiVersion](#)

3.14.1 Detailed Description

Definition at line 289 of file PluginDefs.h.

3.14.2 Member Data Documentation

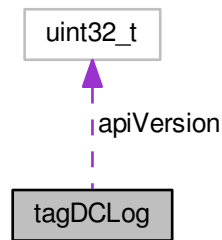
3.14.2.1 uint32_t tagDCInterface::apiVersion

Definition at line 291 of file PluginDefs.h.

3.15 tagDCLog Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCLog:



Public Member Functions

- `void (DCAPI *log)(const char *msg)`

Public Attributes

- `uint32_t apiVersion`

3.15.1 Detailed Description

Definition at line 355 of file PluginDefs.h.

3.15.2 Member Function Documentation

3.15.2.1 `tagDCLog::void (DCAPI * log) const`

3.15.3 Member Data Documentation

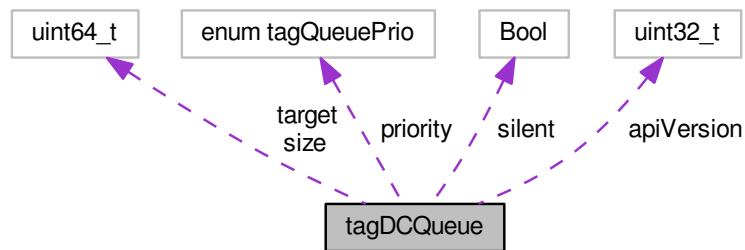
3.15.3.1 `uint32_t tagDCLog::apiVersion`

Definition at line 357 of file PluginDefs.h.

3.16 tagDCQueue Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCQueue:



Public Member Functions

- `Bool (DCAPI *pause)(QueueDataPtr hItem)`
- `QueueDataPtr (DCAPI *add_list)(UserDataPtr hUser`
- `QueueDataPtr (DCAPI *add_download)(const char *hash`
- `QueueDataPtr (DCAPI *find_download)(const char *target)`
- `QueueDataPtr (DCAPI *copy)(const QueueDataPtr hItem)`
- `void (DCAPI *remove_download)(QueueDataPtr hItem)`
- `void (DCAPI *set_priority)(QueueDataPtr hItem)`
- `void (DCAPI *release)(QueueDataPtr hCopy)`

Public Attributes

- `uint32_t apiVersion`
- `QueuePrio priority`
- `Bool silent`
- `uint64_t size`
- `uint64_t const char * target`

3.16.1 Detailed Description

Definition at line 401 of file PluginDefs.h.

3.16.2 Member Function Documentation

3.16.2.1 `tagDCQueue::Bool (DCAPI * pause)`

3.16.2.2 `tagDCQueue::QueueDataPtr (DCAPI * add_list)`

3.16.2.3 `tagDCQueue::QueueDataPtr (DCAPI * add_download) const`

3.16.2.4 `tagDCQueue::QueueDataPtr (DCAPI * find_download) const`

3.16.2.5 `tagDCQueue::QueueDataPtr (DCAPI * copy) const`

3.16.2.6 tagDCQueue::void (DCAPI * *remove_download*)

3.16.2.7 tagDCQueue::void (DCAPI * *set_priority*)

3.16.2.8 tagDCQueue::void (DCAPI * *release*)

3.16.3 Member Data Documentation

3.16.3.1 uint32_t tagDCQueue::apiVersion

Definition at line 403 of file PluginDefs.h.

3.16.3.2 QueuePrio tagDCQueue::priority

Definition at line 410 of file PluginDefs.h.

3.16.3.3 Bool tagDCQueue::silent

Definition at line 405 of file PluginDefs.h.

3.16.3.4 uint64_t tagDCQueue::size

Definition at line 406 of file PluginDefs.h.

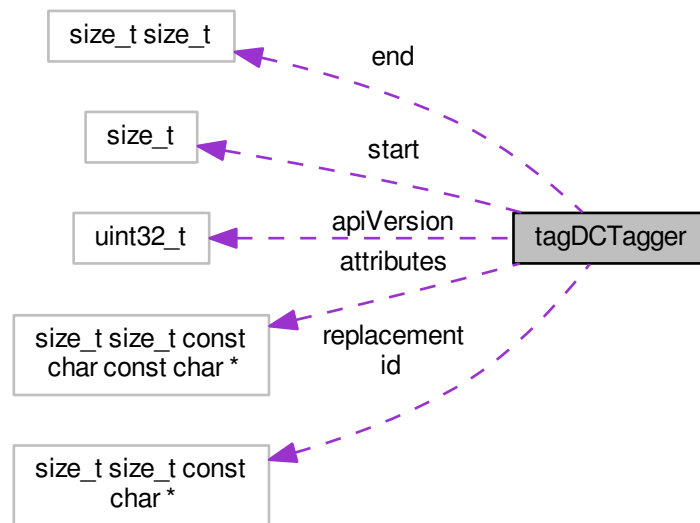
3.16.3.5 uint64_t const char* tagDCQueue::target

Definition at line 406 of file PluginDefs.h.

3.17 tagDCTagger Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCTagger:



Public Member Functions

- `const char *DCAPI * get_text (TagDataPtr hTags)`
- `void (DCAPI *add_tag)(TagDataPtr hTags)`
- `void (DCAPI *replace_text)(TagDataPtr hTags)`

Public Attributes

- `uint32_t apiVersion`
- `size_t size_t const char const char * attributes`
- `size_t size_t end`
- `size_t size_t const char * id`
- `size_t size_t const char * replacement`
- `size_t start`

3.17.1 Detailed Description

Definition at line 435 of file PluginDefs.h.

3.17.2 Member Function Documentation

3.17.2.1 `const char* DCAPI* tagDCTagger::get_text (TagDataPtr hTags)`

3.17.2.2 `tagDCTagger::void (DCAPI * add_tag)`

3.17.2.3 `tagDCTagger::void (DCAPI * replace_text)`

3.17.3 Member Data Documentation

3.17.3.1 uint32_t tagDCTagger::apiVersion

Definition at line 437 of file PluginDefs.h.

3.17.3.2 size_t size_t const char const char* tagDCTagger::attributes

Definition at line 441 of file PluginDefs.h.

3.17.3.3 size_t size_t tagDCTagger::end

Definition at line 441 of file PluginDefs.h.

3.17.3.4 size_t size_t const char* tagDCTagger::id

Definition at line 441 of file PluginDefs.h.

3.17.3.5 size_t size_t const char* tagDCTagger::replacement

Definition at line 442 of file PluginDefs.h.

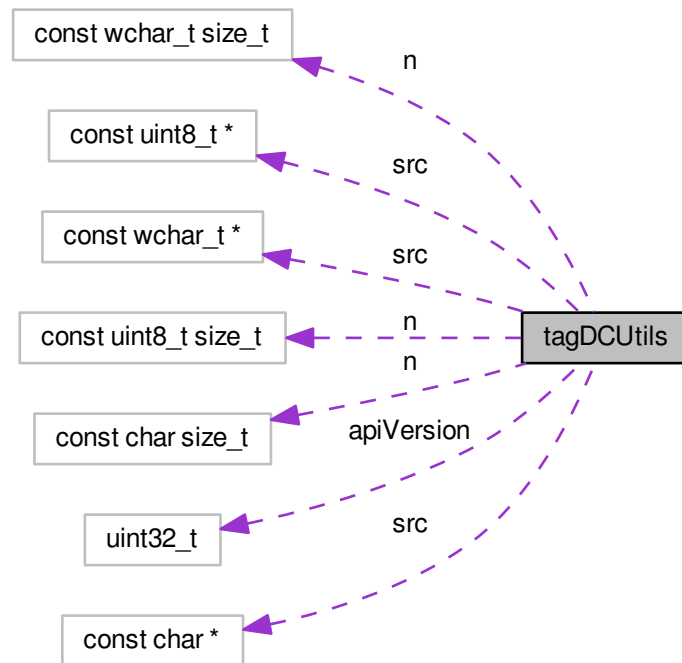
3.17.3.6 size_t tagDCTagger::start

Definition at line 441 of file PluginDefs.h.

3.18 tagDCUtils Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagDCUtils:



Public Member Functions

- [size_t](#) (DCAPI *to_utf8)(char *dst
- [size_t](#) (DCAPI *from_utf8)(char *dst
- [size_t](#) (DCAPI *utf8_to_wcs)(wchar_t *dst
- [size_t](#) (DCAPI *wcs_to_utf8)(char *dst
- [size_t](#) (DCAPI *to_base32)(char *dst
- [size_t](#) (DCAPI *from_base32)(uint8_t *dst

Public Attributes

- `uint32_t` [apiVersion](#)
- `const char` [size_t n](#)
- `const wchar_t` [size_t n](#)
- `const uint8_t` [size_t n](#)
- `const char *` [src](#)
- `const wchar_t *` [src](#)
- `const uint8_t *` [src](#)

3.18.1 Detailed Description

Definition at line 418 of file PluginDefs.h.

3.18.2 Member Function Documentation

3.18.2.1 `tagDCUtils::size_t (DCAPI * to_utf8)`

3.18.2.2 `tagDCUtils::size_t (DCAPI * from_utf8)`

3.18.2.3 `tagDCUtils::size_t (DCAPI * utf8_to_wcs)`

3.18.2.4 `tagDCUtils::size_t (DCAPI * wcs_to_utf8)`

3.18.2.5 `tagDCUtils::size_t (DCAPI * to_base32)`

3.18.2.6 `tagDCUtils::size_t (DCAPI * from_base32)`

3.18.3 Member Data Documentation

3.18.3.1 `uint32_t tagDCUtils::apiVersion`

Definition at line 420 of file PluginDefs.h.

3.18.3.2 `const char size_t tagDCUtils::n`

Definition at line 424 of file PluginDefs.h.

3.18.3.3 `const wchar_t size_t tagDCUtils::n`

Definition at line 428 of file PluginDefs.h.

3.18.3.4 `const uint8_t size_t tagDCUtils::n`

Definition at line 430 of file PluginDefs.h.

3.18.3.5 `const char * tagDCUtils::src`

Definition at line 424 of file PluginDefs.h.

3.18.3.6 `const wchar_t* tagDCUtils::src`

Definition at line 428 of file PluginDefs.h.

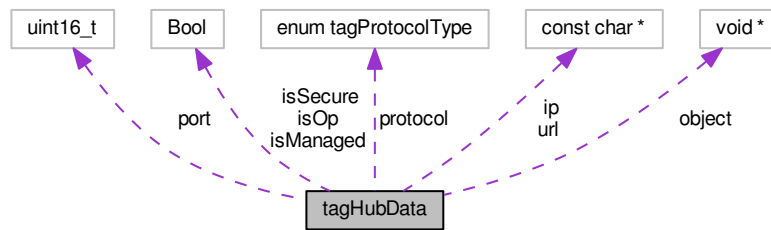
3.18.3.7 `const uint8_t* tagDCUtils::src`

Definition at line 430 of file PluginDefs.h.

3.19 tagHubData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagHubData:



Public Attributes

- `const char * ip`
- `Bool isManaged`
- `Bool isOp`
- `Bool isSecure`
- `dcptr_t object`
- `uint16_t port`
- `ProtocolType protocol`
- `const char * url`

3.19.1 Detailed Description

Definition at line 230 of file `PluginDefs.h`.

3.19.2 Member Data Documentation

3.19.2.1 `const char* tagHubData::ip`

Definition at line 232 of file `PluginDefs.h`.

3.19.2.2 `Bool tagHubData::isManaged`

Definition at line 238 of file `PluginDefs.h`.

3.19.2.3 `Bool tagHubData::isOp`

Definition at line 236 of file `PluginDefs.h`.

3.19.2.4 `Bool tagHubData::isSecure`

Definition at line 237 of file `PluginDefs.h`.

3.19.2.5 `dcptr_t tagHubData::object`

Definition at line 233 of file `PluginDefs.h`.

3.19.2.6 uint16_t tagHubData::port

Definition at line 234 of file PluginDefs.h.

3.19.2.7 ProtocolType tagHubData::protocol

Definition at line 235 of file PluginDefs.h.

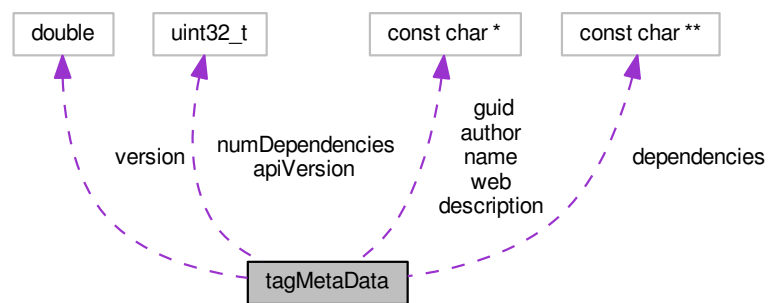
3.19.2.8 const char* tagHubData::url

Definition at line 231 of file PluginDefs.h.

3.20 tagMetaData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagMetaData:



Public Attributes

- uint32_t [apiVersion](#)
- const char * [author](#)
- const char ** [dependencies](#)
- const char * [description](#)
- const char * [guid](#)
- const char * [name](#)
- uint32_t [numDependencies](#)
- double [version](#)
- const char * [web](#)

3.20.1 Detailed Description

Definition at line 276 of file PluginDefs.h.

3.20.2 Member Data Documentation

3.20.2.1 `uint32_t tagMetaData::apiVersion`

Definition at line 284 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.2 `const char* tagMetaData::author`

Definition at line 278 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.3 `const char** tagMetaData::dependencies`

Definition at line 282 of file PluginDefs.h.

3.20.2.4 `const char* tagMetaData::description`

Definition at line 279 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.5 `const char* tagMetaData::guid`

Definition at line 281 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.6 `const char* tagMetaData::name`

Definition at line 277 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.7 `uint32_t tagMetaData::numDependencies`

Definition at line 283 of file PluginDefs.h.

3.20.2.8 `double tagMetaData::version`

Definition at line 285 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.20.2.9 `const char* tagMetaData::web`

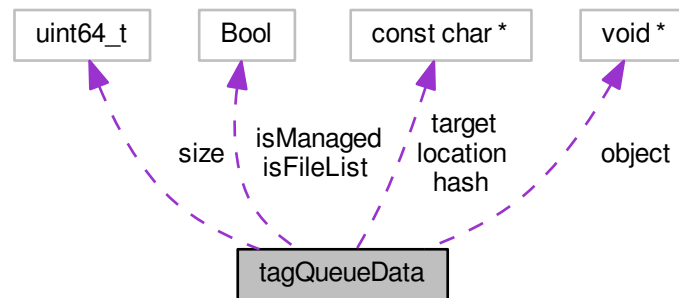
Definition at line 280 of file PluginDefs.h.

Referenced by `pluginInit()`.

3.21 tagQueueData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagQueueData:



Public Attributes

- `const char *` [hash](#)
- `Bool` [isFileList](#)
- `Bool` [isManaged](#)
- `const char *` [location](#)
- `dcptr_t` [object](#)
- `uint64_t` [size](#)
- `const char *` [target](#)

3.21.1 Detailed Description

Definition at line 259 of file `PluginDefs.h`.

3.21.2 Member Data Documentation

3.21.2.1 `const char*` `tagQueueData::hash`

Definition at line 262 of file `PluginDefs.h`.

3.21.2.2 `Bool` `tagQueueData::isFileList`

Definition at line 265 of file `PluginDefs.h`.

3.21.2.3 `Bool` `tagQueueData::isManaged`

Definition at line 266 of file `PluginDefs.h`.

3.21.2.4 `const char* tagQueueData::location`

Definition at line 261 of file PluginDefs.h.

3.21.2.5 `dcptr_t tagQueueData::object`

Definition at line 263 of file PluginDefs.h.

3.21.2.6 `uint64_t tagQueueData::size`

Definition at line 264 of file PluginDefs.h.

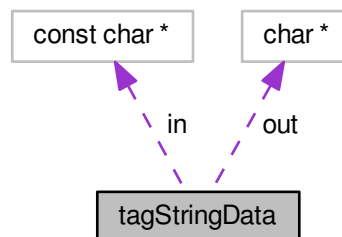
3.21.2.7 `const char* tagQueueData::target`

Definition at line 260 of file PluginDefs.h.

3.22 tagStringData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagStringData:



Public Attributes

- `const char *` [in](#)
- `char *` [out](#)

3.22.1 Detailed Description

Definition at line 206 of file PluginDefs.h.

3.22.2 Member Data Documentation

3.22.2.1 `const char* tagStringData::in`

Definition at line 207 of file PluginDefs.h.

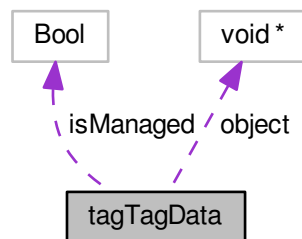
3.22.2.2 char* tagStringData::out

Definition at line 208 of file PluginDefs.h.

3.23 tagTagData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagTagData:



Public Attributes

- [Bool isManaged](#)
- [dcptr_t object](#)

3.23.1 Detailed Description

Definition at line 270 of file PluginDefs.h.

3.23.2 Member Data Documentation

3.23.2.1 Bool tagTagData::isManaged

Definition at line 272 of file PluginDefs.h.

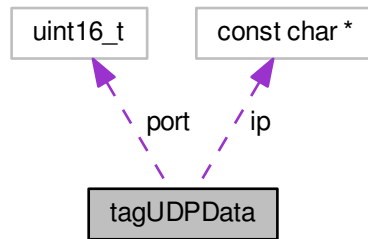
3.23.2.2 dcptr_t tagTagData::object

Definition at line 271 of file PluginDefs.h.

3.24 tagUDPData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagUDPData:



Public Attributes

- `const char *` [ip](#)
- `uint16_t` [port](#)

3.24.1 Detailed Description

Definition at line 253 of file PluginDefs.h.

3.24.2 Member Data Documentation

3.24.2.1 `const char*` `tagUDPData::ip`

Definition at line 254 of file PluginDefs.h.

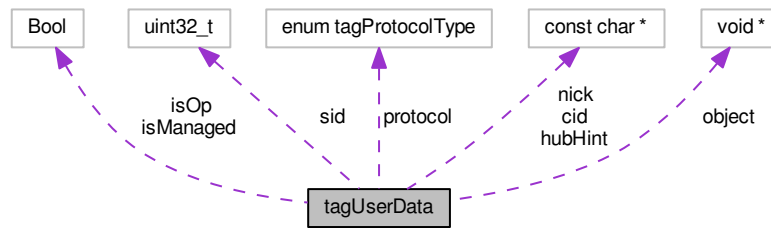
3.24.2.2 `uint16_t` `tagUDPData::port`

Definition at line 255 of file PluginDefs.h.

3.25 tagUserData Struct Reference

```
#include <PluginDefs.h>
```

Collaboration diagram for tagUserData:



Public Attributes

- `const char * cid`
- `const char * hubHint`
- `Bool isManaged`
- `Bool isOp`
- `const char * nick`
- `dcptr_t object`
- `ProtocolType protocol`
- `uint32_t sid`

3.25.1 Detailed Description

Definition at line 218 of file `PluginDefs.h`.

3.25.2 Member Data Documentation

3.25.2.1 `const char* tagUserData::cid`

Definition at line 221 of file `PluginDefs.h`.

3.25.2.2 `const char* tagUserData::hubHint`

Definition at line 220 of file `PluginDefs.h`.

3.25.2.3 `Bool tagUserData::isManaged`

Definition at line 226 of file `PluginDefs.h`.

3.25.2.4 `Bool tagUserData::isOp`

Definition at line 225 of file `PluginDefs.h`.

3.25.2.5 `const char* tagUserData::nick`

Definition at line 219 of file `PluginDefs.h`.

3.25.2.6 `dcptr_t` `tagUserData::object`

Definition at line 222 of file `PluginDefs.h`.

3.25.2.7 `ProtocolType` `tagUserData::protocol`

Definition at line 224 of file `PluginDefs.h`.

3.25.2.8 `uint32_t` `tagUserData::sid`

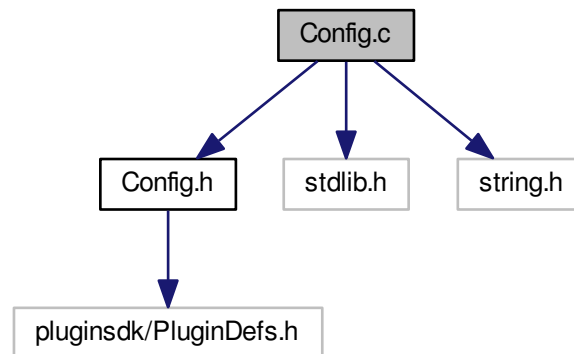
Definition at line 223 of file `PluginDefs.h`.

Chapter 4

File Documentation

4.1 Config.c File Reference

```
#include "Config.h"  
#include <stdlib.h>  
#include <string.h>  
Include dependency graph for Config.c:
```



Functions

- [void DCAPI free_core_cfg \(ConfigValuePtr val\)](#)
- [char *DCAPI get_cfg \(const char *name\)](#)
- [int32_t DCAPI get_cfg_int \(const char *name\)](#)
- [int64_t DCAPI get_cfg_int64 \(const char *name\)](#)
- [char *DCAPI get_cfg_path \(PathType path\)](#)
- [ConfigValuePtr DCAPI get_core_cfg \(const char *name\)](#)
- [char *DCAPI get_install_path \(\)](#)
- [Bool DCAPI init_cfg \(DCCorePtr core, const char *pluginGuid\)](#)
- [void DCAPI set_cfg \(const char *name, const char *value\)](#)
- [void DCAPI set_cfg_int \(const char *name, int32_t value\)](#)
- [void DCAPI set_cfg_int64 \(const char *name, int64_t value\)](#)
- [void DCAPI uninit_cfg \(\)](#)

Variables

- [DCConfigPtr config](#)
- `char * guid`

4.1.1 Function Documentation

4.1.1.1 `void DCAPI free_core_cfg (ConfigValuePtr val)`

Definition at line 100 of file Config.c.

4.1.1.2 `char* DCAPI get_cfg (const char * name)`

Definition at line 43 of file Config.c.

References CFG_TYPE_STRING, [guid](#), and [tagConfigStr::value](#).

4.1.1.3 `int32_t DCAPI get_cfg_int (const char * name)`

Definition at line 54 of file Config.c.

References CFG_TYPE_INT, [guid](#), and [tagConfigInt::value](#).

4.1.1.4 `int64_t DCAPI get_cfg_int64 (const char * name)`

Definition at line 63 of file Config.c.

References CFG_TYPE_INT64, [guid](#), and [tagConfigInt64::value](#).

4.1.1.5 `char* DCAPI get_cfg_path (PathType path)`

Definition at line 102 of file Config.c.

References [tagConfigStr::value](#).

4.1.1.6 `ConfigValuePtr DCAPI get_core_cfg (const char * name)`

Definition at line 99 of file Config.c.

References CFG_TYPE_UNKNOWN.

4.1.1.7 `char* DCAPI get_install_path ()`

Definition at line 113 of file Config.c.

References [guid](#), and [tagConfigStr::value](#).

4.1.1.8 `Bool DCAPI init_cfg (DCCorePtr core, const char * pluginGuid)`

Definition at line 29 of file Config.c.

References DCINTF_CONFIG, DCINTF_CONFIG_VER, `False`, [guid](#), and `True`.

Referenced by [onLoad\(\)](#).

Here is the caller graph for this function:



4.1.1.9 void DCAPI set_cfg (const char * *name*, const char * *value*)

Definition at line 72 of file Config.c.

References `CFG_TYPE_STRING`, `guid`, `tagConfigStr::type`, and `tagConfigStr::value`.

4.1.1.10 void DCAPI set_cfg_int (const char * *name*, int32_t *value*)

Definition at line 81 of file Config.c.

References `CFG_TYPE_INT`, `guid`, `tagConfigInt::type`, and `tagConfigInt::value`.

4.1.1.11 void DCAPI set_cfg_int64 (const char * *name*, int64_t *value*)

Definition at line 90 of file Config.c.

References `CFG_TYPE_INT64`, `guid`, `tagConfigInt64::type`, and `tagConfigInt64::value`.

4.1.1.12 void DCAPI uninit_cfg ()

Definition at line 39 of file Config.c.

References `guid`.

Referenced by `onUnload()`.

Here is the caller graph for this function:



4.1.2 Variable Documentation

4.1.2.1 DCConfigPtr config

Definition at line 26 of file Config.c.

4.1.2.2 char* guid

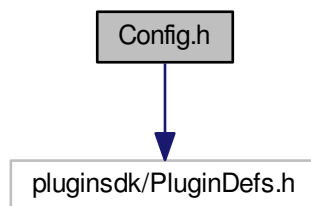
Definition at line 27 of file Config.c.

Referenced by `get_cfg()`, `get_cfg_int()`, `get_cfg_int64()`, `get_install_path()`, `init_cfg()`, `set_cfg()`, `set_cfg_int()`, `set_cfg_int64()`, and `uninit_cfg()`.

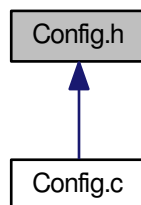
4.2 Config.h File Reference

```
#include <pluginsdk/PluginDefs.h>
```

Include dependency graph for Config.h:



This graph shows which files directly or indirectly include this file:



Functions

- [void DCAPI free_core_cfg](#) ([ConfigValuePtr](#) val)
- [char *DCAPI get_cfg](#) (const char *name)
- [int32_t DCAPI get_cfg_int](#) (const char *name)
- [int64_t DCAPI get_cfg_int64](#) (const char *name)
- [char *DCAPI get_cfg_path](#) ([PathType](#) path)
- [ConfigValuePtr DCAPI get_core_cfg](#) (const char *name)
- [char *DCAPI get_install_path](#) ()
- [Bool DCAPI init_cfg](#) ([DCCorePtr](#) core, const char *pluginGuid)
- [void DCAPI set_cfg](#) (const char *name, const char *value)

- [void DCAPI set_cfg_int](#) (const char *name, int32_t value)
- [void DCAPI set_cfg_int64](#) (const char *name, int64_t value)
- [void DCAPI uninit_cfg](#) ()

4.2.1 Function Documentation

4.2.1.1 void DCAPI free_core_cfg (ConfigValuePtr val)

Definition at line 100 of file Config.c.

4.2.1.2 char* DCAPI get_cfg (const char * name)

Definition at line 43 of file Config.c.

References CFG_TYPE_STRING, guid, and tagConfigStr::value.

4.2.1.3 int32_t DCAPI get_cfg_int (const char * name)

Definition at line 54 of file Config.c.

References CFG_TYPE_INT, guid, and tagConfigInt::value.

4.2.1.4 int64_t DCAPI get_cfg_int64 (const char * name)

Definition at line 63 of file Config.c.

References CFG_TYPE_INT64, guid, and tagConfigInt64::value.

4.2.1.5 char* DCAPI get_cfg_path (PathType path)

Definition at line 102 of file Config.c.

References tagConfigStr::value.

4.2.1.6 ConfigValuePtr DCAPI get_core_cfg (const char * name)

Definition at line 99 of file Config.c.

References CFG_TYPE_UNKNOWN.

4.2.1.7 char* DCAPI get_install_path ()

Definition at line 113 of file Config.c.

References guid, and tagConfigStr::value.

4.2.1.8 Bool DCAPI init_cfg (DCCorePtr core, const char * pluginGuid)

Definition at line 29 of file Config.c.

References DCINTF_CONFIG, DCINTF_CONFIG_VER, False, guid, and True.

Referenced by onLoad().

Here is the caller graph for this function:



4.2.1.9 void DCAPI set_cfg (const char * *name*, const char * *value*)

Definition at line 72 of file Config.c.

References CFG_TYPE_STRING, guid, tagConfigStr::type, and tagConfigStr::value.

4.2.1.10 void DCAPI set_cfg_int (const char * *name*, int32_t *value*)

Definition at line 81 of file Config.c.

References CFG_TYPE_INT, guid, tagConfigInt::type, and tagConfigInt::value.

4.2.1.11 void DCAPI set_cfg_int64 (const char * *name*, int64_t *value*)

Definition at line 90 of file Config.c.

References CFG_TYPE_INT64, guid, tagConfigInt64::type, and tagConfigInt64::value.

4.2.1.12 void DCAPI uninit_cfg ()

Definition at line 39 of file Config.c.

References guid.

Referenced by onUnload().

Here is the caller graph for this function:

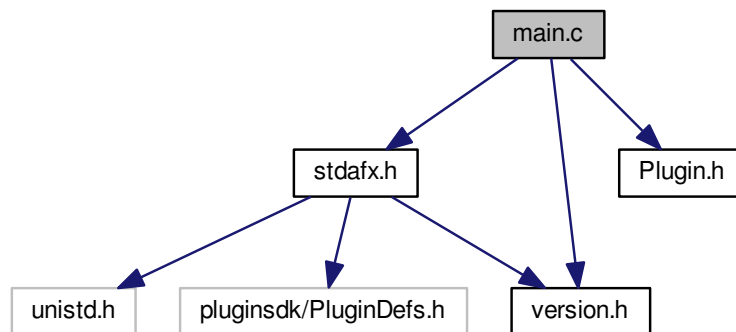


4.3 main.c File Reference

```

#include "stdafx.h"
#include "Plugin.h"
#include "version.h"
  
```

Include dependency graph for main.c:



Functions

- [DCEXP DCMAIN DCAPI pluginInit](#) ([MetaDataPtr](#) info)

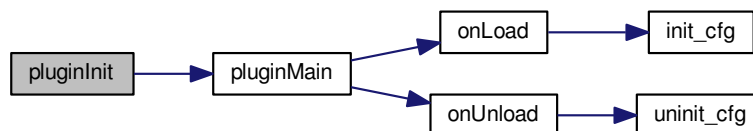
4.3.1 Function Documentation

4.3.1.1 DCEXP DCMAIN DCAPI pluginInit ([MetaDataPtr](#) info)

Definition at line 24 of file main.c.

References [tagMetaData::apiVersion](#), [tagMetaData::author](#), [DCAPI_CORE_VER](#), [tagMetaData::description](#), [tagMetaData::guid](#), [tagMetaData::name](#), [PLUGIN_AUTHOR](#), [PLUGIN_DESC](#), [PLUGIN_GUID](#), [PLUGIN_NAME](#), [PLUGIN_VERSION](#), [PLUGIN_WEB](#), [pluginMain\(\)](#), [tagMetaData::version](#), and [tagMetaData::web](#).

Here is the call graph for this function:

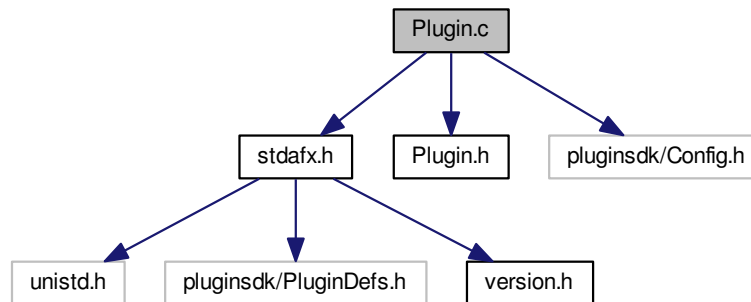


4.4 Plugin.c File Reference

```

#include "stdafx.h"
#include "Plugin.h"
#include <pluginsdk/Config.h>
  
```

Include dependency graph for Plugin.c:



Macros

- `#define HOOKS_SUBSCRIBED 0`

Functions

- `Bool onLoad (DCCorePtr core, Bool install, Bool runtime)`
- `Bool onUnload ()`
- `Bool DCAPI pluginMain (PluginState state, DCCorePtr core, dcptr_t pData)`

Variables

- `DCCorePtr dcpp`
- `DCHOOK hookFuncs [HOOKS_SUBSCRIBED]`
- `const char * hookGuids [HOOKS_SUBSCRIBED]`
- `DCHooksPtr hooks`
- `DCLogPtr logging`
- `subsHandle subs [HOOKS_SUBSCRIBED]`

4.4.1 Macro Definition Documentation

4.4.1.1 `#define HOOKS_SUBSCRIBED 0`

Definition at line 32 of file Plugin.c.

Referenced by `onLoad()`, and `onUnload()`.

4.4.2 Function Documentation

4.4.2.1 `Bool onLoad (DCCorePtr core, Bool install, Bool runtime)`

Definition at line 71 of file Plugin.c.

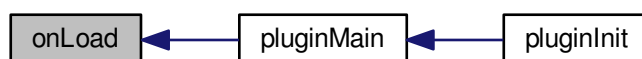
References `DCINTF_HOOKS`, `DCINTF_HOOKS_VER`, `DCINTF_LOGGING`, `DCINTF_LOGGING_VER`, `False`, `hookFuncs`, `hookGuids`, `HOOKS_SUBSCRIBED`, `init_cfg()`, `PLUGIN_GUID`, `subs`, and `True`.

Referenced by pluginMain().

Here is the call graph for this function:



Here is the caller graph for this function:



4.4.2.2 Bool onUnload ()

Definition at line 104 of file Plugin.c.

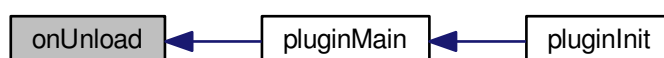
References `HOOKS_SUBSCRIBED`, `subs`, `True`, and `uninit_cfg()`.

Referenced by pluginMain().

Here is the call graph for this function:



Here is the caller graph for this function:



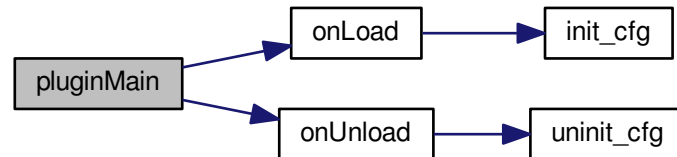
4.4.2.3 Bool DCAPI pluginMain (PluginState state, DCCorePtr core, dcptr_t pData)

Definition at line 49 of file Plugin.c.

References False, ON_INSTALL, ON_LOAD, ON_LOAD_RUNTIME, ON_UNINSTALL, ON_UNLOAD, onLoad(), and onUnload().

Referenced by pluginInit().

Here is the call graph for this function:



Here is the caller graph for this function:



4.4.3 Variable Documentation

4.4.3.1 DCCorePtr dcpp

Definition at line 26 of file Plugin.c.

4.4.3.2 DCHOOK hookFuncs[HOOKS_SUBSCRIBED]

Initial value:

```

= {
}

```

Definition at line 38 of file Plugin.c.

Referenced by onLoad().

4.4.3.3 const char* hookGuids[HOOKS_SUBSCRIBED]

Initial value:

```
= {
}
```

Definition at line 34 of file Plugin.c.

Referenced by onLoad().

4.4.3.4 DCHooksPtr hooks

Definition at line 28 of file Plugin.c.

4.4.3.5 DCLogPtr logging

Definition at line 29 of file Plugin.c.

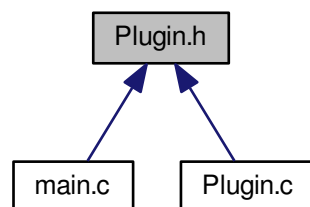
4.4.3.6 subsHandle subs[HOOKS_SUBSCRIBED]

Definition at line 42 of file Plugin.c.

Referenced by onLoad(), and onUnload().

4.5 Plugin.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- [Bool DCAPI pluginMain \(PluginState state, DCCorePtr core, dcptr_t pData\)](#)

4.5.1 Function Documentation

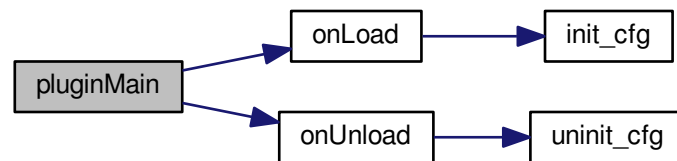
4.5.1.1 Bool DCAPI pluginMain (PluginState state, DCCorePtr core, dcptr_t pData)

Definition at line 49 of file Plugin.c.

References False, ON_INSTALL, ON_LOAD, ON_LOAD_RUNTIME, ON_UNINSTALL, ON_UNLOAD, onLoad(), and onUnload().

Referenced by pluginInit().

Here is the call graph for this function:



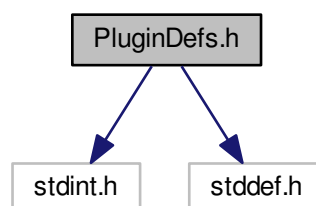
Here is the caller graph for this function:



4.6 PluginDefs.h File Reference

```
#include <stdint.h>
#include <stddef.h>
```

Include dependency graph for `PluginDefs.h`:



Classes

- struct [DCUI](#)
- struct [tagCommandData](#)
- struct [tagConfigBool](#)
- struct [tagConfigInt](#)

- struct [tagConfigInt64](#)
- struct [tagConfigStr](#)
- struct [tagConfigValue](#)
- struct [tagConnectionData](#)
- struct [tagDCConfig](#)
- struct [tagDCConnection](#)
- struct [tagDCCore](#)
- struct [tagDCHooks](#)
- struct [tagDCHub](#)
- struct [tagDCInterface](#)
- struct [tagDCLog](#)
- struct [tagDCQueue](#)
- struct [tagDCTagger](#)
- struct [tagDCUtils](#)
- struct [tagHubData](#)
- struct [tagMetaData](#)
- struct [tagQueueData](#)
- struct [tagStringData](#)
- struct [tagTagData](#)
- struct [tagUDPData](#)
- struct [tagUserData](#)

Macros

- `#define Bool dcBool`
- `#define DCAPI`
- `#define DCAPI_CORE_VER 8`
- `#define DCEXP __attribute__((visibility("default")))`
- `#define DCIMP __attribute__((visibility("default")))`
- `#define DCINTF_CONFIG "generic.plugins.DCConfig" /* Config management */`
- `#define DCINTF_CONFIG_VER 1`
- `#define DCINTF_DCPP_CONNECTIONS "dcpp.network.DCConnection" /* Peer connections */`
- `#define DCINTF_DCPP_CONNECTIONS_VER 1`
- `#define DCINTF_DCPP_HUBS "dcpp.network.DCHub" /* Hubs */`
- `#define DCINTF_DCPP_HUBS_VER 1`
- `#define DCINTF_DCPP_QUEUE "dcpp.queue.DCQueue" /* Download Queue */`
- `#define DCINTF_DCPP_QUEUE_VER 1`
- `#define DCINTF_DCPP_TAGGER "dcpp.xml.DCTagger" /* Manipulation of an XML tagger */`
- `#define DCINTF_DCPP_TAGGER_VER 1`
- `#define DCINTF_DCPP_UI "dcpp.ui.DCUI" /* User interface */`
- `#define DCINTF_DCPP_UI_VER 1`
- `#define DCINTF_DCPP_UTILS "dcpp.utils.DCUtils" /* Utility and convenience functions */`
- `#define DCINTF_DCPP_UTILS_VER 1`
- `#define DCINTF_HOOKS "generic.plugins.DCHooks"`
- `#define DCINTF_HOOKS_VER DCAPI_CORE_VER`
- `#define DCINTF_LOGGING "generic.plugins.DCLog" /* Logging functions */`
- `#define DCINTF_LOGGING_VER 1`
- `#define False dcFalse`
- `#define HOOK_CHAT_IN "dcpp.chat.onIncomingChat" /* Incoming chat from hub (obj: HubData) */`
- `#define HOOK_CHAT_OUT "dcpp.chat.onOutgoingChat" /* Outgoing chat (obj: HubData) */`
- `#define HOOK_CHAT_PM_IN "dcpp.chat.onIncomingPM" /* Incoming private message (obj: UserData) */`
- `#define HOOK_CHAT_PM_OUT "dcpp.chat.onOutgoingPM" /* Outgoing private message (obj: UserData) */`
- `#define HOOK_HUB_OFFLINE "dcpp.hubs.onOffline" /* Hub has just gone offline (obj: HubData) */`
- `#define HOOK_HUB_ONLINE "dcpp.hubs.onOnline" /* (New) hub has just gone online (obj: HubData) */`

- `#define HOOK_NETWORK_CONN_IN "dcpp.network.onClientDataIn" /* Incoming client<->client protocol message (obj: ConnectionData) */`
- `#define HOOK_NETWORK_CONN_OUT "dcpp.network.onClientDataOut" /* Outgoing client<->client protocol message (obj: ConnectionData) */`
- `#define HOOK_NETWORK_HUB_IN "dcpp.network.onHubDataIn" /* Incoming protocol messages from hub (obj: HubData) */`
- `#define HOOK_NETWORK_HUB_OUT "dcpp.network.onHubDataOut" /* Outgoing protocol message to hub (obj: HubData) */`
- `#define HOOK_NETWORK_UDP_IN "dcpp.network.onUDPDataIn" /* Incoming UDP data (obj: UDPData) */`
- `#define HOOK_NETWORK_UDP_OUT "dcpp.network.onUDPDataOut" /* Outgoing UDP data (obj: UDPData) */`
- `#define HOOK_QUEUE_ADDED "dcpp.queue.onAdded" /* (New) item has been added to download queue (obj: QueueData) */`
- `#define HOOK_QUEUE_FINISHED "dcpp.queue.onFinished" /* Item has just finished downloading (obj: QueueData) */`
- `#define HOOK_QUEUE_MOVED "dcpp.queue.onMoved" /* Download queue item has been moved to new location (obj: QueueData) */`
- `#define HOOK_QUEUE_REMOVED "dcpp.queue.onRemoved" /* Item has just been removed from download queue (obj: QueueData) */`
- `#define HOOK_TIMER_MINUTE "dcpp.timer.onMinute" /* Timer event fired once per minute (data: tick value) */`
- `#define HOOK_TIMER_SECOND "dcpp.timer.onSecond" /* Timer event fired once per second (data: tick value) */`
- `#define HOOK_UI_CHAT_COMMAND "dcpp.ui.onChatCommand" /* Client side commands in hub chat (obj: HubData; data: CommandData) */`
- `#define HOOK_UI_CHAT_COMMAND_PM "dcpp.ui.onChatCommandPM" /* Client side commands in private chat (obj: UserData; data: CommandData) */`
- `#define HOOK_UI_CHAT_DISPLAY "dcpp.ui.onChatDisplay" /* Chat messages before they are displayed in chat (obj: UserData; data: StringData) */`
- `#define HOOK_UI_CHAT_TAGS "dcpp.ui.onChatTags" /* Chat message tags before tag merging (obj: UserData; data: TagData) */`
- `#define HOOK_UI_CREATED "dcpp.ui.onCreated" /* Host application UI has been created (obj: if any, impl. dependant) */`
- `#define HOOK_USER_OFFLINE "dcpp.users.onOffline" /* User is offline (obj: UserData) */`
- `#define HOOK_USER_ONLINE "dcpp.users.onOnline" /* User is online (obj: UserData) */`
- `#define True dcTrue`

Typedefs

- `typedef dcptr_t dcptr_t Bool * bBreak`
- `typedef struct tagCommandData CommandData`
- `typedef struct tagCommandData * CommandDataPtr`
- `typedef struct tagConfigBool ConfigBool`
- `typedef struct tagConfigBool * ConfigBoolPtr`
- `typedef struct tagConfigInt ConfigInt`
- `typedef struct tagConfigInt64 ConfigInt64`
- `typedef struct tagConfigInt64 * ConfigInt64Ptr`
- `typedef struct tagConfigInt * ConfigIntPtr`
- `typedef struct tagConfigStr ConfigStr`
- `typedef struct tagConfigStr * ConfigStrPtr`
- `typedef enum tagConfigType ConfigType`
- `typedef struct tagConfigValue ConfigValue`
- `typedef struct tagConfigValue * ConfigValuePtr`
- `typedef struct tagConnectionData ConnectionData`
- `typedef struct tagConnectionData * ConnectionDataPtr`

- typedef [DCCorePtr](#) core
- typedef enum [tagDCBool](#) dcBool
- typedef struct [tagDCConfig](#) DCConfig
- typedef struct [tagDCConfig](#) * DCConfigPtr
- typedef struct [tagDCConnection](#) DCConnection
- typedef struct [tagDCConnection](#) * DCConnectionPtr
- typedef struct [tagDCCore](#) DCCore
- typedef struct [tagDCCore](#) * DCCorePtr
- typedef struct [tagDCHooks](#) DCHooks
- typedef struct [tagDCHooks](#) * DCHooksPtr
- typedef struct [tagDCHub](#) DCHub
- typedef struct [tagDCHub](#) * DCHubPtr
- typedef struct [tagDCInterface](#) DCInterface
- typedef struct [tagDCInterface](#) * DCInterfacePtr
- typedef struct [tagDCLog](#) DCLog
- typedef struct [tagDCLog](#) * DCLogPtr
- typedef void * dcptr_t
- typedef struct [tagDCQueue](#) DCQueue
- typedef struct [tagDCQueue](#) * DCQueuePtr
- typedef struct [tagDCTagger](#) DCTagger
- typedef struct [tagDCTagger](#) * DCTaggerPtr
- typedef uint64_t dctime_t
- typedef struct [DCUI](#) DCUI
- typedef struct [DCUI](#) * DCUIPtr
- typedef struct [tagDCUtils](#) DCUtils
- typedef struct [tagDCUtils](#) * DCUtilsPtr
- typedef void * hookHandle
- typedef struct [tagHubData](#) HubData
- typedef struct [tagHubData](#) * HubDataPtr
- typedef void * intfHandle
- typedef struct [tagMetaData](#) MetaData
- typedef struct [tagMetaData](#) * MetaDataPtr
- typedef enum [tagMsgType](#) MsgType
- typedef enum [tagPathType](#) PathType
- typedef [dcptr_t](#) dcptr_t pCommon
- typedef [DCCorePtr](#) dcptr_t pData
- typedef enum [tagPluginState](#) PluginState
- typedef enum [tagProtocolType](#) ProtocolType
- typedef struct [tagQueueData](#) QueueData
- typedef struct [tagQueueData](#) * QueueDataPtr
- typedef enum [tagQueuePrio](#) QueuePrio
- typedef struct [tagStringData](#) StringData
- typedef struct [tagStringData](#) * StringDataPtr
- typedef void * subsHandle
- typedef struct [tagTagData](#) TagData
- typedef struct [tagTagData](#) * TagDataPtr
- typedef struct [tagUDPData](#) UDPData
- typedef struct [tagUDPData](#) * UDPDataPtr
- typedef struct [tagUserData](#) UserData
- typedef struct [tagUserData](#) * UserDataPtr

Enumerations

- enum `tagConfigType` { `CFG_TYPE_UNKNOWN` = -2, `CFG_TYPE_REMOVE`, `CFG_TYPE_STRING`, `CFG_TYPE_INT`, `CFG_TYPE_BOOL`, `CFG_TYPE_INT64` }
- enum `tagDCBool` { `dcFalse` = 0, `dcTrue` }
- enum `tagMsgType` { `MSG_CLIENT` = 0, `MSG_STATUS`, `MSG_SYSTEM` }
- enum `tagPathType` { `PATH_GLOBAL_CONFIG` = 0, `PATH_USER_CONFIG`, `PATH_USER_LOCAL`, `PATH_RESOURCES`, `PATH_LOCALE` }
- enum `tagPluginState` { `ON_INSTALL` = 0, `ON_LOAD`, `ON_LOAD_RUNTIME`, `ON_CONFIGURE`, `ON_UNLOAD`, `ON_UNINSTALL` }
- enum `tagProtocolType` { `PROTOCOL_ADC` = 0, `PROTOCOL_NMDC`, `PROTOCOL_DHT` }
- enum `tagQueuePrio` { `PRIO_DEFAULT` = -1, `PRIO_LOWEST` = 1, `PRIO_LOW`, `PRIO_NORMAL`, `PRIO_HIGH`, `PRIO_HIGHEST` }

Functions

- typedef `Bool` (`DCAPI *DCMAIN`)(`PluginState` pluginState
- typedef `void` (`DCAPI *DCCommandFunc`)(const char *name)

4.6.1 Macro Definition Documentation

4.6.1.1 #define Bool dcBool

Definition at line 172 of file PluginDefs.h.

4.6.1.2 #define DCAPI

Definition at line 49 of file PluginDefs.h.

4.6.1.3 #define DCAPI_CORE_VER 8

Definition at line 35 of file PluginDefs.h.

Referenced by `pluginInit()`.

4.6.1.4 #define DCEXP __attribute__((visibility("default")))

Definition at line 51 of file PluginDefs.h.

4.6.1.5 #define DCIMP __attribute__((visibility("default")))

Definition at line 52 of file PluginDefs.h.

4.6.1.6 #define DCINTF_CONFIG "generic.plugins.DCConfig" /* Config management */

Definition at line 60 of file PluginDefs.h.

Referenced by `init_cfg()`.

4.6.1.7 #define DCINTF_CONFIG_VER 1

Definition at line 61 of file PluginDefs.h.

Referenced by `init_cfg()`.

4.6.1.8 `#define DCINTF_DCPP_CONNECTIONS "dcpp.network.DCConnection" /* Peer connections */`

Definition at line 67 of file PluginDefs.h.

4.6.1.9 `#define DCINTF_DCPP_CONNECTIONS_VER 1`

Definition at line 68 of file PluginDefs.h.

4.6.1.10 `#define DCINTF_DCPP_HUBS "dcpp.network.DCHub" /* Hubs */`

Definition at line 70 of file PluginDefs.h.

4.6.1.11 `#define DCINTF_DCPP_HUBS_VER 1`

Definition at line 71 of file PluginDefs.h.

4.6.1.12 `#define DCINTF_DCPP_QUEUE "dcpp.queue.DCQueue" /* Download Queue */`

Definition at line 73 of file PluginDefs.h.

4.6.1.13 `#define DCINTF_DCPP_QUEUE_VER 1`

Definition at line 74 of file PluginDefs.h.

4.6.1.14 `#define DCINTF_DCPP_TAGGER "dcpp.xml.DCTagger" /* Manipulation of an XML tagger */`

Definition at line 79 of file PluginDefs.h.

4.6.1.15 `#define DCINTF_DCPP_TAGGER_VER 1`

Definition at line 80 of file PluginDefs.h.

4.6.1.16 `#define DCINTF_DCPP_UI "dcpp.ui.DCUI" /* User interface */`

Definition at line 82 of file PluginDefs.h.

4.6.1.17 `#define DCINTF_DCPP_UI_VER 1`

Definition at line 83 of file PluginDefs.h.

4.6.1.18 `#define DCINTF_DCPP_UTILS "dcpp.utils.DCUtils" /* Utility and convenience functions */`

Definition at line 76 of file PluginDefs.h.

4.6.1.19 `#define DCINTF_DCPP_UTILS_VER 1`

Definition at line 77 of file PluginDefs.h.

4.6.1.20 `#define DCINTF_HOOKS "generic.plugins.DCHooks"`

Definition at line 56 of file PluginDefs.h.

Referenced by `onLoad()`.

4.6.1.21 `#define DCINTF_HOOKS_VER DCAPI_CORE_VER`

Definition at line 57 of file PluginDefs.h.

Referenced by `onLoad()`.

4.6.1.22 `#define DCINTF_LOGGING "generic.plugins.DCLog" /* Logging functions */`

Definition at line 63 of file PluginDefs.h.

Referenced by `onLoad()`.

4.6.1.23 `#define DCINTF_LOGGING_VER 1`

Definition at line 64 of file PluginDefs.h.

Referenced by `onLoad()`.

4.6.1.24 `#define False dcFalse`

Definition at line 174 of file PluginDefs.h.

Referenced by `init_cfg()`, `onLoad()`, and `pluginMain()`.

4.6.1.25 `#define HOOK_CHAT_IN "dcpp.chat.onIncomingChat" /* Incoming chat from hub (obj: HubData) */`

Definition at line 86 of file PluginDefs.h.

4.6.1.26 `#define HOOK_CHAT_OUT "dcpp.chat.onOutgoingChat" /* Outgoing chat (obj: HubData) */`

Definition at line 87 of file PluginDefs.h.

4.6.1.27 `#define HOOK_CHAT_PM_IN "dcpp.chat.onIncomingPM" /* Incoming private message (obj: UserData) */`

Definition at line 88 of file PluginDefs.h.

4.6.1.28 `#define HOOK_CHAT_PM_OUT "dcpp.chat.onOutgoingPM" /* Outgoing private message (obj: UserData) */`

Definition at line 89 of file PluginDefs.h.

4.6.1.29 `#define HOOK_HUB_OFFLINE "dcpp.hubs.onOffline" /* Hub has just gone offline (obj: HubData) */`

Definition at line 95 of file PluginDefs.h.

4.6.1.30 `#define HOOK_HUB_ONLINE "dcpp.hubs.onOnline" /* (New) hub has just gone online (obj: HubData) */`

Definition at line 94 of file PluginDefs.h.

4.6.1.31 `#define HOOK_NETWORK_CONN_IN "dcpp.network.onClientDataIn" /* Incoming client<->client protocol message (obj: ConnectionData) */`

Definition at line 101 of file PluginDefs.h.

4.6.1.32 `#define HOOK_NETWORK_CONN_OUT "dcpp.network.onClientDataOut" /* Outgoing client<->client protocol message (obj: ConnectionData) */`

Definition at line 102 of file PluginDefs.h.

4.6.1.33 `#define HOOK_NETWORK_HUB_IN "dcpp.network.onHubDataIn" /* Incoming protocol messages from hub (obj: HubData) */`

Definition at line 99 of file PluginDefs.h.

4.6.1.34 `#define HOOK_NETWORK_HUB_OUT "dcpp.network.onHubDataOut" /* Outgoing protocol message to hub (obj: HubData) */`

Definition at line 100 of file PluginDefs.h.

4.6.1.35 `#define HOOK_NETWORK_UDP_IN "dcpp.network.onUDPDataIn" /* Incoming UDP data (obj: UDPData) */`

Definition at line 103 of file PluginDefs.h.

4.6.1.36 `#define HOOK_NETWORK_UDP_OUT "dcpp.network.onUDPDataOut" /* Outgoing UDP data (obj: UDPData) */`

Definition at line 104 of file PluginDefs.h.

4.6.1.37 `#define HOOK_QUEUE_ADDED "dcpp.queue.onAdded" /* (New) item has been added to download queue (obj: QueueData) */`

Definition at line 106 of file PluginDefs.h.

4.6.1.38 `#define HOOK_QUEUE_FINISHED "dcpp.queue.onFinished" /* Item has just finished downloading (obj: QueueData) */`

Definition at line 109 of file PluginDefs.h.

4.6.1.39 `#define HOOK_QUEUE_MOVED "dcpp.queue.onMoved" /* Download queue item has been moved to new location (obj: QueueData) */`

Definition at line 107 of file PluginDefs.h.

4.6.1.40 `#define HOOK_QUEUE_REMOVED "dcpp.queue.onRemoved" /* Item has just been removed from download queue (obj: QueueData) */`

Definition at line 108 of file PluginDefs.h.

4.6.1.41 `#define HOOK_TIMER_MINUTE "dcpp.timer.onMinute" /* Timer event fired once per minute (data: tick value) */`

Definition at line 92 of file PluginDefs.h.

4.6.1.42 **#define HOOK_TIMER_SECOND** "dcpp.timer.onSecond" /* Timer event fired once per second (data: tick value) */

Definition at line 91 of file PluginDefs.h.

4.6.1.43 **#define HOOK_UI_CHAT_COMMAND** "dcpp.ui.onChatCommand" /* Client side commands in hub chat (obj: **HubData**; data: **CommandData**) */

Definition at line 114 of file PluginDefs.h.

4.6.1.44 **#define HOOK_UI_CHAT_COMMAND_PM** "dcpp.ui.onChatCommandPM" /* Client side commands in private chat (obj: **UserData**; data: **CommandData**) */

Definition at line 115 of file PluginDefs.h.

4.6.1.45 **#define HOOK_UI_CHAT_DISPLAY** "dcpp.ui.onChatDisplay" /* Chat messages before they are displayed in chat (obj: **UserData**; data: **StringData**) */

Definition at line 113 of file PluginDefs.h.

4.6.1.46 **#define HOOK_UI_CHAT_TAGS** "dcpp.ui.onChatTags" /* Chat message tags before tag merging (obj: **UserData**; data: **TagData**) */

Definition at line 112 of file PluginDefs.h.

4.6.1.47 **#define HOOK_UI_CREATED** "dcpp.ui.onCreated" /* Host application UI has been created (obj: if any, impl. dependant) */

Definition at line 111 of file PluginDefs.h.

4.6.1.48 **#define HOOK_USER_OFFLINE** "dcpp.users.onOffline" /* User is offline (obj: **UserData**) */

Definition at line 97 of file PluginDefs.h.

4.6.1.49 **#define HOOK_USER_ONLINE** "dcpp.users.onOnline" /* User is online (obj: **UserData**) */

Definition at line 96 of file PluginDefs.h.

4.6.1.50 **#define True** **dcTrue**

Definition at line 173 of file PluginDefs.h.

Referenced by `init_cfg()`, `onLoad()`, and `onUnload()`.

4.6.2 Typedef Documentation

4.6.2.1 **typedef** **dcptr_t** **dcptr_t** **Bool*** **bBreak**

Definition at line 317 of file PluginDefs.h.

- 4.6.2.2 `typedef struct tagCommandData CommandData`
- 4.6.2.3 `typedef struct tagCommandData * CommandDataPtr`
- 4.6.2.4 `typedef struct tagConfigBool ConfigBool`
- 4.6.2.5 `typedef struct tagConfigBool * ConfigBoolPtr`
- 4.6.2.6 `typedef struct tagConfigInt ConfigInt`
- 4.6.2.7 `typedef struct tagConfigInt64 ConfigInt64`
- 4.6.2.8 `typedef struct tagConfigInt64 * ConfigInt64Ptr`
- 4.6.2.9 `typedef struct tagConfigInt * ConfigIntPtr`
- 4.6.2.10 `typedef struct tagConfigStr ConfigStr`
- 4.6.2.11 `typedef struct tagConfigStr * ConfigStrPtr`
- 4.6.2.12 `typedef enum tagConfigType ConfigType`
- 4.6.2.13 `typedef struct tagConfigValue ConfigValue`
- 4.6.2.14 `typedef struct tagConfigValue * ConfigValuePtr`
- 4.6.2.15 `typedef struct tagConnectionData ConnectionData`
- 4.6.2.16 `typedef struct tagConnectionData * ConnectionDataPtr`
- 4.6.2.17 `typedef DCCorePtr core`

Definition at line 312 of file PluginDefs.h.

- 4.6.2.18 `typedef enum tagDCBool dcBool`
- 4.6.2.19 `typedef struct tagDCConfig DCConfig`
- 4.6.2.20 `typedef struct tagDCConfig * DCConfigPtr`
- 4.6.2.21 `typedef struct tagDCConnection DCConnection`
- 4.6.2.22 `typedef struct tagDCConnection * DCConnectionPtr`
- 4.6.2.23 `typedef struct tagDCCore DCCore`
- 4.6.2.24 `typedef struct tagDCCore * DCCorePtr`
- 4.6.2.25 `typedef struct tagDCHooks DCHooks`
- 4.6.2.26 `typedef struct tagDCHooks * DCHooksPtr`
- 4.6.2.27 `typedef struct tagDCHub DCHub`
- 4.6.2.28 `typedef struct tagDCHub * DCHubPtr`

4.6.2.29 `typedef struct tagDCInterface DCInterface`

4.6.2.30 `typedef struct tagDCInterface * DCInterfacePtr`

4.6.2.31 `typedef struct tagDCLog DCLog`

4.6.2.32 `typedef struct tagDCLog * DCLogPtr`

4.6.2.33 `typedef void * dcptr_t`

Definition at line 167 of file PluginDefs.h.

4.6.2.34 `typedef struct tagDCQueue DCQueue`

4.6.2.35 `typedef struct tagDCQueue * DCQueuePtr`

4.6.2.36 `typedef struct tagDCTagger DCTagger`

4.6.2.37 `typedef struct tagDCTagger * DCTaggerPtr`

4.6.2.38 `typedef uint64_t dctime_t`

Definition at line 169 of file PluginDefs.h.

4.6.2.39 `typedef struct DCUI DCUI`

4.6.2.40 `typedef struct DCUI * DCUIPtr`

4.6.2.41 `typedef struct tagDCUtils DCUtils`

4.6.2.42 `typedef struct tagDCUtils * DCUtilsPtr`

4.6.2.43 `typedef void* hookHandle`

Definition at line 167 of file PluginDefs.h.

4.6.2.44 `typedef struct tagHubData HubData`

4.6.2.45 `typedef struct tagHubData * HubDataPtr`

4.6.2.46 `typedef void * intfHandle`

Definition at line 167 of file PluginDefs.h.

4.6.2.47 `typedef struct tagMetaData MetaData`

4.6.2.48 `typedef struct tagMetaData * MetaDataPtr`

4.6.2.49 `typedef enum tagMsgType MsgType`

4.6.2.50 `typedef enum tagPathType PathType`

4.6.2.51 `typedef dcptr_t dcptr_t pCommon`

Definition at line 317 of file PluginDefs.h.

4.6.2.52 `typedef dcptr_t pData`

Definition at line 312 of file PluginDefs.h.

4.6.2.53 `typedef enum tagPluginState PluginState`

4.6.2.54 `typedef enum tagProtocolType ProtocolType`

4.6.2.55 `typedef struct tagQueueData QueueData`

4.6.2.56 `typedef struct tagQueueData * QueueDataPtr`

4.6.2.57 `typedef enum tagQueuePrio QueuePrio`

4.6.2.58 `typedef struct tagStringData StringData`

4.6.2.59 `typedef struct tagStringData * StringDataPtr`

4.6.2.60 `typedef void * subsHandle`

Definition at line 167 of file PluginDefs.h.

4.6.2.61 `typedef struct tagTagData TagData`

4.6.2.62 `typedef struct tagTagData * TagDataPtr`

4.6.2.63 `typedef struct tagUDPData UDPData`

4.6.2.64 `typedef struct tagUDPData * UDPDataPtr`

4.6.2.65 `typedef struct tagUserData UserData`

4.6.2.66 `typedef struct tagUserData * UserDataPtr`

4.6.3 Enumeration Type Documentation

4.6.3.1 `enum tagConfigType`

Enumerator

CFG_TYPE_UNKNOWN

CFG_TYPE_REMOVE

CFG_TYPE_STRING

CFG_TYPE_INT

CFG_TYPE_BOOL

CFG_TYPE_INT64

Definition at line 128 of file PluginDefs.h.

4.6.3.2 enum tagDCBool

Enumerator

dcFalse

dcTrue

Definition at line 168 of file PluginDefs.h.

4.6.3.3 enum tagMsgType

Enumerator

MSG_CLIENT

MSG_STATUS

MSG_SYSTEM

Definition at line 151 of file PluginDefs.h.

4.6.3.4 enum tagPathType

Enumerator

PATH_GLOBAL_CONFIG

PATH_USER_CONFIG

PATH_USER_LOCAL

PATH_RESOURCES

PATH_LOCALE

Definition at line 143 of file PluginDefs.h.

4.6.3.5 enum tagPluginState

Enumerator

ON_INSTALL

ON_LOAD

ON_LOAD_RUNTIME

ON_CONFIGURE

ON_UNLOAD

ON_UNINSTALL

Definition at line 118 of file PluginDefs.h.

4.6.3.6 enum tagProtocolType

Enumerator

PROTOCOL_ADC

PROTOCOL_NMDC

PROTOCOL_DHT

Definition at line 137 of file PluginDefs.h.

4.6.3.7 enum tagQueuePrio

Enumerator

PRIO_DEFAULT
PRIO_LOWEST
PRIO_LOW
PRIO_NORMAL
PRIO_HIGH
PRIO_HIGHEST

Definition at line 157 of file PluginDefs.h.

4.6.4 Function Documentation

4.6.4.1 typedef Bool (DCAPI * DCMAIN)

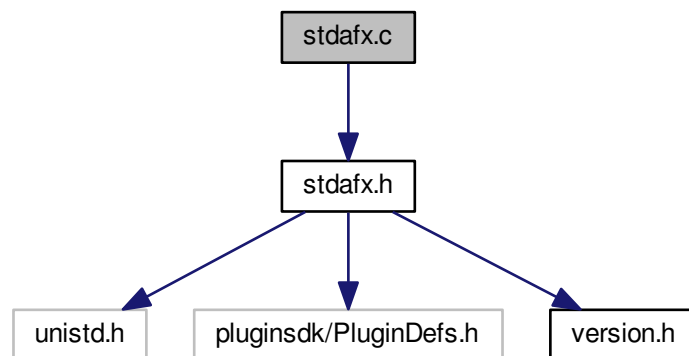
4.6.4.2 typedef void (DCAPI * DCCommandFunc) const

4.7 resource.h File Reference

4.8 stdafx.c File Reference

```
#include "stdafx.h"
```

Include dependency graph for stdafx.c:



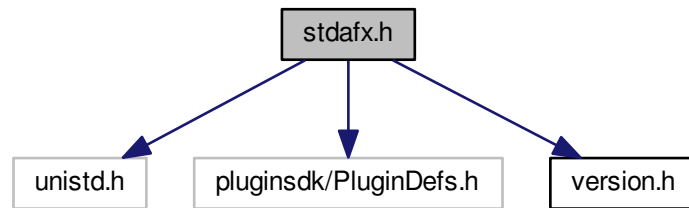
4.9 stdafx.h File Reference

```

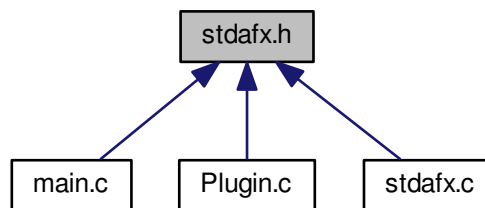
#include <unistd.h>
#include <pluginsdk/PluginDefs.h>
#include "version.h"

```

Include dependency graph for stdafx.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define` [PLUGIN_VERSION](#)

4.9.1 Macro Definition Documentation

4.9.1.1 `#define` [PLUGIN_VERSION](#)

Value:

```

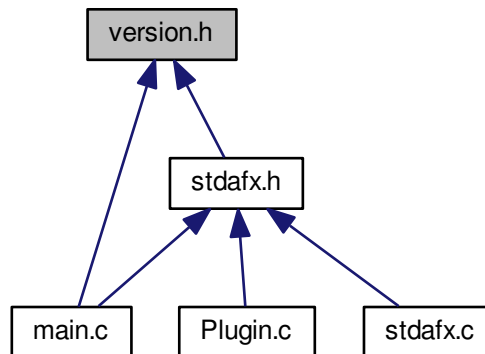
(PLUGIN_VERSION_MAJOR + (PLUGIN_VERSION_MINOR >= 10 ?
  PLUGIN_VERSION_MINOR >= 100 ? \
  PLUGIN_VERSION_MINOR / 1000. : PLUGIN_VERSION_MINOR / 100. :
  PLUGIN_VERSION_MINOR / 10.))
  
```

Definition at line 37 of file stdafx.h.

Referenced by `pluginInit()`.

4.10 version.h File Reference

This graph shows which files directly or indirectly include this file:



Macros

- `#define PLUGIN_AUTHOR "..."`
- `#define PLUGIN_DESC "..."`
- `#define PLUGIN_GUID "..."`
- `#define PLUGIN_NAME "..."`
- `#define PLUGIN_VERSION_MAJOR 1`
- `#define PLUGIN_VERSION_MINOR 0`
- `#define PLUGIN_WEB "N/A"`

4.10.1 Macro Definition Documentation

4.10.1.1 `#define PLUGIN_AUTHOR "..."`

Definition at line 33 of file version.h.

Referenced by pluginInit().

4.10.1.2 `#define PLUGIN_DESC "..."`

Definition at line 36 of file version.h.

Referenced by pluginInit().

4.10.1.3 `#define PLUGIN_GUID "..."`

Definition at line 27 of file version.h.

Referenced by onLoad(), and pluginInit().

4.10.1.4 `#define PLUGIN_NAME "..."`

Definition at line 30 of file version.h.

Referenced by pluginInit().

4.10.1.5 `#define PLUGIN_VERSION_MAJOR 1`

Definition at line 39 of file version.h.

4.10.1.6 `#define PLUGIN_VERSION_MINOR 0`

Definition at line 40 of file version.h.

4.10.1.7 `#define PLUGIN_WEB "N/A"`

Definition at line 43 of file version.h.

Referenced by pluginInit().

Index

- apiVersion
 - DCUI, [6](#)
 - tagDCConfig, [15](#)
 - tagDCConnection, [16](#)
 - tagDCCore, [17](#)
 - tagDCHooks, [19](#)
 - tagDCHub, [21](#)
 - tagDCInterface, [22](#)
 - tagDCLog, [23](#)
 - tagDCQueue, [25](#)
 - tagDCTagger, [27](#)
 - tagDCUtils, [29](#)
 - tagMetaData, [32](#)
- attributes
 - tagDCTagger, [27](#)
- author
 - tagMetaData, [32](#)
- bBreak
 - PluginDefs.h, [58](#)
- Bool
 - PluginDefs.h, [54](#), [63](#)
 - tagDCCore, [17](#)
 - tagDCHooks, [19](#)
 - tagDCHub, [20](#)
 - tagDCQueue, [24](#)
- CFG_TYPE_BOOL
 - PluginDefs.h, [61](#)
- CFG_TYPE_INT
 - PluginDefs.h, [61](#)
- CFG_TYPE_INT64
 - PluginDefs.h, [61](#)
- CFG_TYPE_REMOVE
 - PluginDefs.h, [61](#)
- CFG_TYPE_STRING
 - PluginDefs.h, [61](#)
- CFG_TYPE_UNKNOWN
 - PluginDefs.h, [61](#)
- cid
 - tagUserData, [37](#)
- cmd
 - tagDCConnection, [16](#)
 - tagDCHub, [21](#)
- command
 - DCUI, [6](#)
 - tagCommandData, [7](#)
- CommandData
 - PluginDefs.h, [58](#)
- CommandDataPtr
 - PluginDefs.h, [59](#)
- config
 - Config.c, [41](#)
- Config.c, [39](#)
 - config, [41](#)
 - free_core_cfg, [40](#)
 - get_cfg, [40](#)
 - get_cfg_int, [40](#)
 - get_cfg_int64, [40](#)
 - get_cfg_path, [40](#)
 - get_core_cfg, [40](#)
 - get_install_path, [40](#)
 - guid, [41](#)
 - init_cfg, [40](#)
 - set_cfg, [41](#)
 - set_cfg_int, [41](#)
 - set_cfg_int64, [41](#)
 - uninit_cfg, [41](#)
- Config.h, [42](#)
 - free_core_cfg, [43](#)
 - get_cfg, [43](#)
 - get_cfg_int, [43](#)
 - get_cfg_int64, [43](#)
 - get_cfg_path, [43](#)
 - get_core_cfg, [43](#)
 - get_install_path, [43](#)
 - init_cfg, [43](#)
 - set_cfg, [44](#)
 - set_cfg_int, [44](#)
 - set_cfg_int64, [44](#)
 - uninit_cfg, [44](#)
- ConfigBool
 - PluginDefs.h, [59](#)
- ConfigBoolPtr
 - PluginDefs.h, [59](#)
- ConfigInt
 - PluginDefs.h, [59](#)
- ConfigInt64
 - PluginDefs.h, [59](#)
- ConfigInt64Ptr
 - PluginDefs.h, [59](#)
- ConfigIntPtr
 - PluginDefs.h, [59](#)
- ConfigStr
 - PluginDefs.h, [59](#)
- ConfigStrPtr
 - PluginDefs.h, [59](#)
 - tagDCConfig, [14](#)
- ConfigType

- PluginDefs.h, [59](#)
- ConfigValue
 - PluginDefs.h, [59](#)
- ConfigValuePtr
 - PluginDefs.h, [59](#)
 - tagDCCConfig, [14](#)
- ConnectionData
 - PluginDefs.h, [59](#)
- ConnectionDataPtr
 - PluginDefs.h, [59](#)
- core
 - PluginDefs.h, [59](#)
- DCAPI
 - PluginDefs.h, [54](#)
- DCAPI_CORE_VER
 - PluginDefs.h, [54](#)
- DCCConfig
 - PluginDefs.h, [59](#)
- DCCConfigPtr
 - PluginDefs.h, [59](#)
- DCCConnection
 - PluginDefs.h, [59](#)
- DCCConnectionPtr
 - PluginDefs.h, [59](#)
- DCCore
 - PluginDefs.h, [59](#)
- DCCorePtr
 - PluginDefs.h, [59](#)
- DCEXP
 - PluginDefs.h, [54](#)
- DCHooks
 - PluginDefs.h, [59](#)
- DCHooksPtr
 - PluginDefs.h, [59](#)
- DCHub
 - PluginDefs.h, [59](#)
- DCHubPtr
 - PluginDefs.h, [59](#)
- DCIMP
 - PluginDefs.h, [54](#)
- DCINTF_CONFIG
 - PluginDefs.h, [54](#)
- DCINTF_CONFIG_VER
 - PluginDefs.h, [54](#)
- DCINTF_DCPP_HUBS
 - PluginDefs.h, [55](#)
- DCINTF_DCPP_QUEUE
 - PluginDefs.h, [55](#)
- DCINTF_DCPP_TAGGER
 - PluginDefs.h, [55](#)
- DCINTF_DCPP_UI
 - PluginDefs.h, [55](#)
- DCINTF_DCPP_UI_VER
 - PluginDefs.h, [55](#)
- DCINTF_DCPP_UTILS
 - PluginDefs.h, [55](#)
- DCINTF_HOOKS
 - PluginDefs.h, [55](#)
- DCINTF_HOOKS_VER
 - PluginDefs.h, [56](#)
- DCINTF_LOGGING
 - PluginDefs.h, [56](#)
- DCINTF_LOGGING_VER
 - PluginDefs.h, [56](#)
- DCInterface
 - PluginDefs.h, [59](#)
- DCInterfacePtr
 - PluginDefs.h, [60](#)
 - tagDCCore, [17](#)
- DCLog
 - PluginDefs.h, [60](#)
- DCLogPtr
 - PluginDefs.h, [60](#)
- DCQueue
 - PluginDefs.h, [60](#)
- DCQueuePtr
 - PluginDefs.h, [60](#)
- DCTagger
 - PluginDefs.h, [60](#)
- DCTaggerPtr
 - PluginDefs.h, [60](#)
- DCUI, [5](#)
 - apiVersion, [6](#)
 - command, [6](#)
 - icon, [6](#)
 - message, [6](#)
 - name, [6](#)
 - PluginDefs.h, [60](#)
 - void, [6](#)
- DCUIPtr
 - PluginDefs.h, [60](#)
- DCUtils
 - PluginDefs.h, [60](#)
- DCUtilsPtr
 - PluginDefs.h, [60](#)
- data
 - tagDCCConnection, [16](#)
- dcFalse
 - PluginDefs.h, [62](#)
- dcTrue
 - PluginDefs.h, [62](#)
- dcBool
 - PluginDefs.h, [59](#)
- dcpp
 - Plugin.c, [48](#)
- dcptr_t
 - PluginDefs.h, [60](#)
- dctime_t
 - PluginDefs.h, [60](#)
- defProc
 - tagDCHooks, [19](#)
- dependencies
 - tagMetaData, [32](#)
- description
 - tagMetaData, [32](#)
- end

- tagDCTagger, [27](#)
- False
 - PluginDefs.h, [56](#)
- free_core_cfg
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_cfg
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_cfg_int
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_cfg_int64
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_cfg_path
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_core_cfg
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_install_path
 - Config.c, [40](#)
 - Config.h, [43](#)
- get_text
 - tagDCTagger, [26](#)
- graceless
 - tagDCConnection, [16](#)
- guid
 - Config.c, [41](#)
 - tagMetaData, [32](#)
- HOOK_CHAT_IN
 - PluginDefs.h, [56](#)
- HOOK_CHAT_OUT
 - PluginDefs.h, [56](#)
- HOOK_CHAT_PM_IN
 - PluginDefs.h, [56](#)
- HOOK_CHAT_PM_OUT
 - PluginDefs.h, [56](#)
- HOOK_HUB_OFFLINE
 - PluginDefs.h, [56](#)
- HOOK_HUB_ONLINE
 - PluginDefs.h, [56](#)
- HOOK_QUEUE_ADDED
 - PluginDefs.h, [57](#)
- HOOK_QUEUE_FINISHED
 - PluginDefs.h, [57](#)
- HOOK_QUEUE_MOVED
 - PluginDefs.h, [57](#)
- HOOK_QUEUE_REMOVED
 - PluginDefs.h, [57](#)
- HOOK_TIMER_MINUTE
 - PluginDefs.h, [57](#)
- HOOK_TIMER_SECOND
 - PluginDefs.h, [57](#)
- HOOK_UI_CHAT_TAGS
 - PluginDefs.h, [58](#)
- HOOK_UI_CREATED
 - PluginDefs.h, [58](#)
- HOOK_USER_OFFLINE
 - PluginDefs.h, [58](#)
- HOOK_USER_ONLINE
 - PluginDefs.h, [58](#)
- HOOKS_SUBSCRIBED
 - Plugin.c, [46](#)
- hash
 - tagQueueData, [33](#)
- hookFuncs
 - Plugin.c, [48](#)
- hookGuids
 - Plugin.c, [48](#)
- hookHandle
 - PluginDefs.h, [60](#)
 - tagDCHooks, [19](#)
- hookProc
 - tagDCHooks, [19](#)
- hooks
 - Plugin.c, [49](#)
- host_name
 - tagDCCore, [17](#)
- HubData
 - PluginDefs.h, [60](#)
- HubDataPtr
 - PluginDefs.h, [60](#)
 - tagDCHub, [20](#)
- hubHint
 - tagUserData, [37](#)
- hubUrl
 - tagDCHub, [21](#)
- icon
 - DCUI, [6](#)
- id
 - tagDCTagger, [27](#)
- in
 - tagStringData, [34](#)
- init_cfg
 - Config.c, [40](#)
 - Config.h, [43](#)
- intfHandle
 - PluginDefs.h, [60](#)
 - tagDCCore, [17](#)
- ip
 - tagConnectionData, [13](#)
 - tagHubData, [30](#)
 - tagUDPData, [36](#)
- isFileList
 - tagQueueData, [33](#)
- isManaged
 - tagConnectionData, [13](#)
 - tagHubData, [30](#)
 - tagQueueData, [33](#)
 - tagTagData, [35](#)
 - tagUserData, [37](#)
- isOp

- tagConnectionData, 13
 - tagHubData, 30
 - tagUserData, 37
- isSecure
 - tagConnectionData, 13
 - tagHubData, 30
- location
 - tagQueueData, 33
- logging
 - Plugin.c, 49
- MSG_CLIENT
 - PluginDefs.h, 62
- MSG_STATUS
 - PluginDefs.h, 62
- MSG_SYSTEM
 - PluginDefs.h, 62
- main.c, 44
 - pluginInit, 45
- message
 - DCUI, 6
- MetaData
 - PluginDefs.h, 60
- MetaDataPtr
 - PluginDefs.h, 60
- msg
 - tagDCHub, 21
- MsgType
 - PluginDefs.h, 60
- n
 - tagDCConnection, 16
 - tagDCUtils, 29
- name
 - DCUI, 6
 - tagMetaData, 32
- nick
 - tagDCHub, 21
 - tagUserData, 37
- numDependencies
 - tagMetaData, 32
- ON_CONFIGURE
 - PluginDefs.h, 62
- ON_INSTALL
 - PluginDefs.h, 62
- ON_LOAD
 - PluginDefs.h, 62
- ON_LOAD_RUNTIME
 - PluginDefs.h, 62
- ON_UNINSTALL
 - PluginDefs.h, 62
- ON_UNLOAD
 - PluginDefs.h, 62
- object
 - tagConnectionData, 13
 - tagHubData, 30
 - tagQueueData, 34
 - tagTagData, 35
 - tagUserData, 37
- onLoad
 - Plugin.c, 46
- onUnload
 - Plugin.c, 47
- out
 - tagStringData, 34
- PATH_GLOBAL_CONFIG
 - PluginDefs.h, 62
- PATH_LOCALE
 - PluginDefs.h, 62
- PATH_RESOURCES
 - PluginDefs.h, 62
- PATH_USER_CONFIG
 - PluginDefs.h, 62
- PATH_USER_LOCAL
 - PluginDefs.h, 62
- PRIO_DEFAULT
 - PluginDefs.h, 63
- PRIO_HIGH
 - PluginDefs.h, 63
- PRIO_HIGHEST
 - PluginDefs.h, 63
- PRIO_LOW
 - PluginDefs.h, 63
- PRIO_LOWEST
 - PluginDefs.h, 63
- PRIO_NORMAL
 - PluginDefs.h, 63
- PROTOCOL_ADC
 - PluginDefs.h, 62
- PROTOCOL_DHT
 - PluginDefs.h, 62
- PROTOCOL_NMDC
 - PluginDefs.h, 62
- pCommon
 - PluginDefs.h, 60
 - tagDCHooks, 19
- pData
 - PluginDefs.h, 61
 - tagDCHooks, 19
- pInterface
 - tagDCCore, 18
- PLUGIN_AUTHOR
 - version.h, 65
- PLUGIN_DESC
 - version.h, 65
- PLUGIN_GUID
 - version.h, 65
- PLUGIN_NAME
 - version.h, 65
- PLUGIN_VERSION
 - stdafx.h, 64
- PLUGIN_WEB
 - version.h, 66
- pObject
 - tagDCHooks, 19

- params
 - tagCommandData, [7](#)
- password
 - tagDCHub, [21](#)
- PathType
 - PluginDefs.h, [60](#)
- Plugin.c, [45](#)
 - dcpp, [48](#)
 - HOOKS_SUBSCRIBED, [46](#)
 - hookFuncs, [48](#)
 - hookGuids, [48](#)
 - hooks, [49](#)
 - logging, [49](#)
 - onLoad, [46](#)
 - onUnload, [47](#)
 - pluginMain, [48](#)
 - subs, [49](#)
- Plugin.h, [49](#)
 - pluginMain, [49](#)
- PluginDefs.h
 - CFG_TYPE_BOOL, [61](#)
 - CFG_TYPE_INT, [61](#)
 - CFG_TYPE_INT64, [61](#)
 - CFG_TYPE_REMOVE, [61](#)
 - CFG_TYPE_STRING, [61](#)
 - CFG_TYPE_UNKNOWN, [61](#)
 - dcFalse, [62](#)
 - dcTrue, [62](#)
 - MSG_CLIENT, [62](#)
 - MSG_STATUS, [62](#)
 - MSG_SYSTEM, [62](#)
 - ON_CONFIGURE, [62](#)
 - ON_INSTALL, [62](#)
 - ON_LOAD, [62](#)
 - ON_LOAD_RUNTIME, [62](#)
 - ON_UNINSTALL, [62](#)
 - ON_UNLOAD, [62](#)
 - PATH_GLOBAL_CONFIG, [62](#)
 - PATH_LOCALE, [62](#)
 - PATH_RESOURCES, [62](#)
 - PATH_USER_CONFIG, [62](#)
 - PATH_USER_LOCAL, [62](#)
 - PRIO_DEFAULT, [63](#)
 - PRIO_HIGH, [63](#)
 - PRIO_HIGHEST, [63](#)
 - PRIO_LOW, [63](#)
 - PRIO_LOWEST, [63](#)
 - PRIO_NORMAL, [63](#)
 - PROTOCOL_ADC, [62](#)
 - PROTOCOL_DHT, [62](#)
 - PROTOCOL_NMDC, [62](#)
- PluginDefs.h, [50](#)
 - bBreak, [58](#)
 - Bool, [54](#), [63](#)
 - CommandData, [58](#)
 - CommandDataPtr, [59](#)
 - ConfigBool, [59](#)
 - ConfigBoolPtr, [59](#)
 - ConfigInt, [59](#)
 - ConfigInt64, [59](#)
 - ConfigInt64Ptr, [59](#)
 - ConfigIntPtr, [59](#)
 - ConfigStr, [59](#)
 - ConfigStrPtr, [59](#)
 - ConfigType, [59](#)
 - ConfigValue, [59](#)
 - ConfigValuePtr, [59](#)
 - ConnectionData, [59](#)
 - ConnectionDataPtr, [59](#)
 - core, [59](#)
 - DCAPI, [54](#)
 - DCAPI_CORE_VER, [54](#)
 - DCConfig, [59](#)
 - DCConfigPtr, [59](#)
 - DCConnection, [59](#)
 - DCConnectionPtr, [59](#)
 - DCCore, [59](#)
 - DCCorePtr, [59](#)
 - DCEXP, [54](#)
 - DCHooks, [59](#)
 - DCHooksPtr, [59](#)
 - DCHub, [59](#)
 - DCHubPtr, [59](#)
 - DCIMP, [54](#)
 - DCINTF_CONFIG, [54](#)
 - DCINTF_CONFIG_VER, [54](#)
 - DCINTF_DCPP_HUBS, [55](#)
 - DCINTF_DCPP_QUEUE, [55](#)
 - DCINTF_DCPP_TAGGER, [55](#)
 - DCINTF_DCPP_UI, [55](#)
 - DCINTF_DCPP_UTILS, [55](#)
 - DCINTF_HOOKS, [55](#)
 - DCINTF_HOOKS_VER, [56](#)
 - DCINTF_LOGGING, [56](#)
 - DCINTF_LOGGING_VER, [56](#)
 - DCInterface, [59](#)
 - DCInterfacePtr, [60](#)
 - DCLog, [60](#)
 - DCLogPtr, [60](#)
 - DCQueue, [60](#)
 - DCQueuePtr, [60](#)
 - DCTagger, [60](#)
 - DCTaggerPtr, [60](#)
 - DCUI, [60](#)
 - DCUIPtr, [60](#)
 - DCUtils, [60](#)
 - DCUtilsPtr, [60](#)
 - dcBool, [59](#)
 - dcptr_t, [60](#)
 - dctime_t, [60](#)
 - False, [56](#)
 - HOOK_CHAT_IN, [56](#)
 - HOOK_CHAT_OUT, [56](#)
 - HOOK_CHAT_PM_IN, [56](#)
 - HOOK_CHAT_PM_OUT, [56](#)
 - HOOK_HUB_OFFLINE, [56](#)

- HOOK_HUB_ONLINE, 56
- HOOK_QUEUE_ADDED, 57
- HOOK_QUEUE_MOVED, 57
- HOOK_QUEUE_REMOVED, 57
- HOOK_TIMER_MINUTE, 57
- HOOK_TIMER_SECOND, 57
- HOOK_UI_CHAT_TAGS, 58
- HOOK_UI_CREATED, 58
- HOOK_USER_OFFLINE, 58
- HOOK_USER_ONLINE, 58
- hookHandle, 60
- HubData, 60
- HubDataPtr, 60
- intfHandle, 60
- MetaData, 60
- MetaDataPtr, 60
- MsgType, 60
- pCommon, 60
- pData, 61
- PathType, 60
- PluginState, 61
- ProtocolType, 61
- QueueData, 61
- QueueDataPtr, 61
- QueuePrio, 61
- StringData, 61
- StringDataPtr, 61
- subsHandle, 61
- tagConfigType, 61
- tagDCBool, 61
- TagData, 61
- TagDataPtr, 61
- tagMsgType, 62
- tagPathType, 62
- tagPluginState, 62
- tagProtocolType, 62
- tagQueuePrio, 62
- True, 58
- UDPData, 61
- UDPDataPtr, 61
- UserData, 61
- UserDataPtr, 61
- void, 63
- pluginInit
 - main.c, 45
- pluginMain
 - Plugin.c, 48
 - Plugin.h, 49
- PluginState
 - PluginDefs.h, 61
- port
 - tagConnectionData, 13
 - tagDCConnection, 16
 - tagHubData, 30
 - tagUDPData, 36
- priority
 - tagDCQueue, 25
- protocol
 - tagConnectionData, 13
 - tagHubData, 31
 - tagUserData, 38
 - ProtocolType
 - PluginDefs.h, 61
 - QueueData
 - PluginDefs.h, 61
 - QueueDataPtr
 - PluginDefs.h, 61
 - tagDCQueue, 24
 - QueuePrio
 - PluginDefs.h, 61
 - replacement
 - tagDCTagger, 27
 - resource.h, 63
 - set_cfg
 - Config.c, 41
 - Config.h, 44
 - set_cfg_int
 - Config.c, 41
 - Config.h, 44
 - set_cfg_int64
 - Config.c, 41
 - Config.h, 44
 - setting
 - tagDCConfig, 15
 - sid
 - tagUserData, 38
 - silent
 - tagDCQueue, 25
 - size
 - tagDCQueue, 25
 - tagQueueData, 34
 - size_t
 - tagDCHooks, 19
 - tagDCUtils, 29
 - src
 - tagDCUtils, 29
 - start
 - tagDCTagger, 27
 - stdafx.c, 63
 - stdafx.h, 63
 - PLUGIN_VERSION, 64
 - StringData
 - PluginDefs.h, 61
 - StringDataPtr
 - PluginDefs.h, 61
 - subs
 - Plugin.c, 49
 - subsHandle
 - PluginDefs.h, 61
 - tagDCHooks, 19
 - tagCommandData, 6
 - command, 7
 - params, 7

- tagConfigBool, 7
 - type, 8
 - value, 8
- tagConfigInt, 8
 - type, 9
 - value, 9
- tagConfigInt64, 9
 - type, 10
 - value, 10
- tagConfigStr, 10
 - type, 11
 - value, 11
- tagConfigType
 - PluginDefs.h, 61
- tagConfigValue, 11
 - type, 12
- tagConnectionData, 12
 - ip, 13
 - isManaged, 13
 - isOp, 13
 - isSecure, 13
 - object, 13
 - port, 13
 - protocol, 13
- tagDCBool
 - PluginDefs.h, 61
- tagDCConfig, 13
 - apiVersion, 15
 - ConfigStrPtr, 14
 - ConfigValuePtr, 14
 - setting, 15
 - type, 15
 - val, 15
 - void, 15
- tagDCConnection, 15
 - apiVersion, 16
 - cmd, 16
 - data, 16
 - graceless, 16
 - n, 16
 - port, 16
 - UserDataPtr, 16
 - void, 16
- tagDCCore, 17
 - apiVersion, 17
 - Bool, 17
 - DCInterfacePtr, 17
 - host_name, 17
 - intfHandle, 17
 - plInterface, 18
 - version, 18
- tagDCHooks, 18
 - apiVersion, 19
 - Bool, 19
 - defProc, 19
 - hookHandle, 19
 - hookProc, 19
 - pCommon, 19
 - pData, 19
 - pObject, 19
 - size_t, 19
 - subsHandle, 19
- tagDCHub, 19
 - apiVersion, 21
 - Bool, 20
 - cmd, 21
 - HubDataPtr, 20
 - hubUrl, 21
 - msg, 21
 - nick, 21
 - password, 21
 - thirdPerson, 21
 - type, 21
 - UserDataPtr, 21
 - void, 21
- tagDCInterface, 22
 - apiVersion, 22
- tagDCLog, 22
 - apiVersion, 23
 - void, 23
- tagDCQueue, 23
 - apiVersion, 25
 - Bool, 24
 - priority, 25
 - QueueDataPtr, 24
 - silent, 25
 - size, 25
 - target, 25
 - void, 24, 25
- tagDCTagger, 25
 - apiVersion, 27
 - attributes, 27
 - end, 27
 - get_text, 26
 - id, 27
 - replacement, 27
 - start, 27
 - void, 26
- tagDCUtils, 27
 - apiVersion, 29
 - n, 29
 - size_t, 29
 - src, 29
- TagData
 - PluginDefs.h, 61
- TagDataPtr
 - PluginDefs.h, 61
- tagHubData, 29
 - ip, 30
 - isManaged, 30
 - isOp, 30
 - isSecure, 30
 - object, 30
 - port, 30
 - protocol, 31
 - url, 31

- tagMetaData, 31
 - apiVersion, 32
 - author, 32
 - dependencies, 32
 - description, 32
 - guid, 32
 - name, 32
 - numDependencies, 32
 - version, 32
 - web, 32
- tagMsgType
 - PluginDefs.h, 62
- tagPathType
 - PluginDefs.h, 62
- tagPluginState
 - PluginDefs.h, 62
- tagProtocolType
 - PluginDefs.h, 62
- tagQueueData, 33
 - hash, 33
 - isFileList, 33
 - isManaged, 33
 - location, 33
 - object, 34
 - size, 34
 - target, 34
- tagQueuePrio
 - PluginDefs.h, 62
- tagStringData, 34
 - in, 34
 - out, 34
- tagTagData, 35
 - isManaged, 35
 - object, 35
- tagUDPData, 35
 - ip, 36
 - port, 36
- tagUserData, 36
 - cid, 37
 - hubHint, 37
 - isManaged, 37
 - isOp, 37
 - nick, 37
 - object, 37
 - protocol, 38
 - sid, 38
- target
 - tagDCQueue, 25
 - tagQueueData, 34
- thirdPerson
 - tagDCHub, 21
- True
 - PluginDefs.h, 58
- type
 - tagConfigBool, 8
 - tagConfigInt, 9
 - tagConfigInt64, 10
 - tagConfigStr, 11
 - tagConfigValue, 12
 - tagDCConfig, 15
 - tagDCHub, 21
- UDPData
 - PluginDefs.h, 61
- UDPDataPtr
 - PluginDefs.h, 61
- uninit_cfg
 - Config.c, 41
 - Config.h, 44
- url
 - tagHubData, 31
- UserData
 - PluginDefs.h, 61
- UserDataPtr
 - PluginDefs.h, 61
 - tagDCConnection, 16
 - tagDCHub, 21
- val
 - tagDCConfig, 15
- value
 - tagConfigBool, 8
 - tagConfigInt, 9
 - tagConfigInt64, 10
 - tagConfigStr, 11
- version
 - tagDCCore, 18
 - tagMetaData, 32
- version.h, 65
 - PLUGIN_AUTHOR, 65
 - PLUGIN_DESC, 65
 - PLUGIN_GUID, 65
 - PLUGIN_NAME, 65
 - PLUGIN_WEB, 66
- void
 - DCUI, 6
 - PluginDefs.h, 63
 - tagDCConfig, 15
 - tagDCConnection, 16
 - tagDCHub, 21
 - tagDCLog, 23
 - tagDCQueue, 24, 25
 - tagDCTagger, 26
- web
 - tagMetaData, 32