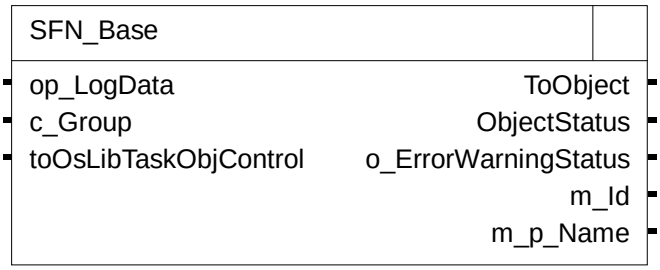
Name: SFN_Base

Revision: 0.28

Ladder: false
Sfc: false
Complex: true

Graphic Schema:



Comment: SafanBase
Basic definition used by each (Safan) Object.
Note:
SafanBase has always a background task enabled but default cyclic time is set to off so it is'nt used.
Reason for this:
If a base class does not have a background task enabled a child class can't have it also.
For cyclic and realtime tasks this restriction is not applicable.

Properties:

	Filename:	C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_Base\SFN_Base.st
	Sigmatek:	false
	OSInterface:	false
	SharedCommandTable:	true
	Objectsize:	(720,300)

Clients:

	Name:	op_LogData
	Class:	SFN_Base
	DataType:	gpt_LogData
	Type:	Data Channel
	Required:	false
	Comment:	Output for logging data to SF_Logger
	Name:	c_Group
	Class:	SFN_Base
	DataType:	UDINT
	Type:	Data Channel
	Required:	false
	Comment:	Group identification object is part of.
	Name:	toOsLibTaskObjControl
	Class:	_TaskObjectControl
	DataType:	DINT
	Type:	Object Channel
	Required:	false

Server:

	Name:	ToObject
	GUID:	{9EED831D-1495-49CA-AD11-0E8CDA8941D2}
	Class:	SFN_Base
	Visualized:	false
	DataType:	pVoid
	Type:	Object Channel
	Initialize:	false
	WriteProtected:	true
	Retentive:	false

	Comment:	ToObject: Object channel server. Channel to reach the methodes of this object. In case this object should be Accessible by a pointer the variable ToObject is used to handover the address of this object. This server is never used for anything else!!!
	Name:	ObjectStatus
	GUID:	{A29E2DB3-B7AD-4363-A726-DB93BF808383}
	Visualized:	false
	DataType:	gt_ObjectStatus
	Type:	Data Channel
	Initialize:	false
	WriteProtected:	false
	Retentive:	false
	Comment:	Generic object status interface. Bit 0 - 15 is reserved for generic Safan framework Bit 16 -31 can be used by the application See also type gt_ObjectStatus.
	Name:	o_ErrorWarningStatus
	GUID:	{71D405E1-73D5-4283-95DF-BAB38DCB9D0B}
	Visualized:	false
	DataType:	UDINT
	Type:	Data Channel
	Initialize:	false
	WriteProtected:	true
	Retentive:	false
	Comment:	Actual active Error or Warning status. A lower level message will not overwite an active higher level message. for example a warning will not overwrite this server if it has already an active error Note: see also internal variable FirstError to find the first error detected since last error handling.
	Name:	m_Id
	GUID:	{AAB7581E-3B92-42E8-9287-D9BCBCF7E8ED}
	Visualized:	false
	DataType:	UDINT
	Type:	Data Channel
	Initialize:	true
	WriteProtected:	false
	Retentive:	false
	Comment:	An identification number of this object. Free to use/define by derived classes and application.
	Name:	m_p_Name
	GUID:	{33AD7665-E336-4A23-BACF-BD56E31FDC73}
	Visualized:	false
	DataType:	pChar
	Type:	Data Channel
	Initialize:	false
	WriteProtected:	true
	Retentive:	false
	Comment:	Name of this object. String is closed by an end of sting character (value 0) String is read only!
Methods:	Name:	SFN_Base
	UseOldHiddenInstImpl:	false
	Virtual:	false
	Global access:	false
	AWL implementation:	false
	CDecl:	false
	UseBaseCmd:	false
	Hidden:	false
	Output:	Name: Ret_code

Type: ConfStates
Pointer: false
Register: <undefined>

Name: Init
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false

Name: Background
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Input:: Name: EAX
Type: UDINT
Pointer: false
Register: EAX
Output:: Name: state
Type: UDINT
Pointer: false
Register: EAX

Name: Clone
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Creates a copy of this object containing the same attribute data

the same as SaveAs functionality.

Output:: Name: p_ObjectOut
Type: pVoid
Pointer: false
Register: <undefined>
Comment: Pointer to cloned object.

Name: GetActuelErrorWarning
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get actual warning or error status.
Only the highest and latest level status will
be returned
Output:: Name: status
Type: UDINT
Pointer: false
Register: <undefined>

Name: GetChannelName
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
Hidden: false

Comment: Get name of a server or client.

Note 1:
Object name is included in name string

Note 2:
p_string should point to 255 byte allocated memory
to store channelName string.
If allocated size is less then 255
you have to be sure string will not exceed that size.

return string examples
"ObjectName.ServerName"
"ObjectName.ClientName"

Input:: Name: p_NameString
Type: ^CHAR
Pointer: true
Register: <undefined>
Comment: Pointer to a 255 bytes string location
to store channel name

Input:: Name: p_Channel
Type: ^void
Pointer: true
Register: <undefined>
Comment: Pointer to a server or client

Input:: Name: p_this
Type: ^void
Pointer: true
Register: <undefined>
Comment: Pointer to object of requested server or client name

Name: GetConnectedObjectName
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get name of object connected to the specified client.

Input:: Name: p_ClientChannel
Type: ^void
Pointer: true
Register: <undefined>
Comment: Pointer to client

Output:: Name: p_ObjName
Type: ^CHAR
Pointer: true
Register: <undefined>
Comment: Pointer to object name string. (read only)

Name: GetConnectedServerName
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get name of server connected to the specified client.

Note: excluding object name
Note: For some reason does not work for object channels !!!!!

Input:: Name: p_ClientChannel
Type: ^void
Pointer: true
Register: <undefined>

Output:: Name: p_SrvName
Type: ^CHAR

Name: GetFirstError
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get the actual error or warning status.
Output::
  Name: status
  Type: UDINT
  Pointer: false
  Register: <undefined>
  Comment: Actual Error or warning status

Name: GetGroupId
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get group Id object is part of
Output::
  Name: group
  Type: UDINT
  Pointer: false
  Register: <undefined>

Name: GetNextSfnObject
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get next object pointer for framework only.

This methode is used to go through all safan class objects
in linked list.

Output::
  Name: p_Object
  Type: ^SFN_Base
  Pointer: true
  Register: <undefined>

Name: GetObjectData
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Take a copy of objectdata and put it into a destination object
Input::
  Name: p_DestObj
  Type: pVoid
  Pointer: false
  Register: <undefined>
  Comment: Pointer to Destination object where to store objectdata
Input::
  Name: DestSize
  Type: UDINT
  Pointer: false
  Register: <undefined>
  Comment: Size of object data to copy.

  can be found by doing sizeof(class)

Register: <undefined>
Comment: Result of this methode

Passed if succesfull
Failed if not

Name: GetObjectStatus
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Gets object status
Output:: Name: status
Type: gt_ObjectStatus
Pointer: false
Register: <undefined>
Comment: Object status

Name: GetThisPtr
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Gets object pointer (this pointer).

Output:: Name: p_ThisPtr
Type: pVoid
Pointer: false
Register: <undefined>
Comment: Object Pointer

Name: ModeEnable
UseOldHiddenInstImpl: false
Virtual: true
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Purpose to enable temporary object
Typically used by internal methodes

See also ExtModeEnable / ExtModeDisable
For fixed setting see ConfigEnable/ConfigDisable

Name: ModeDisable
UseOldHiddenInstImpl: false
Virtual: true
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Disable object
Purpose to disable temporary object
Typically used by internal methodes

See also ExtModeEnable / ExtModeDisable
For fixed setting see ConfigEnable/ConfigDisable

Name: ConfigEnable
UseOldHiddenInstImpl: false
Virtual: true

UseBaseCmd: false
Hidden: false
Comment: Configuration Enable object

Purpose Enable configuration setting is for fixed settings
For temporary setting see ModeEnable/ModeDisable and
ExtModeEnable/ExtModeDisable

Name: ConfigDisable
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Configuration Disable object

Purpose Disable configuration setting is for fixed settings
For temporary setting see ModeEnable/ModeDisable and
ExtModeEnable/ExtModeDisable

Name: ExtModeEnable
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Enable object from external instances

Purpose to enable temporary object
Typically used by External methodes

See also ModeEnable / ModeDisable
For fixed setting see ConfigEnable/ConfigDisable

Output:: Name: ResultStatus
Type: BOOL
Pointer: false
Register: <undefined>

Name: ExtModeDisable
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Disable object from an external object.
like HMI or a machine setting file.

Purpose mode setting is for temporary settings from
external objects
See also ModeEnable / ModeDisable
For fixed setting see ConfigEnable/ConfigDisable

Output:: Name: ResultStatus
Type: BOOL
Pointer: false
Register: <undefined>

Name: Verify
UseOldHiddenInstImpl: false
Virtual: true
Global access: true

Hidden: false
Comment: Checks internal consistency of this object.

Note:
This methode checks only.

Output:: Name: Result
Type: BOOL
Pointer: false
Register: <undefined>
Comment: Return status:
status=0 : PASSED
status<>0 : FAILED

Name: UpdateObjectStatusEnable
UseOldHiddenInstImpl: false
Virtual: true
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Update actual object enable status.

Name: Load
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: load object data from remanent memory into temporary memory (RAM).

Name: ResetError
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Reset internal and application errors.

Note:
Critical errors are cleared if application is restarted.
Output:: Name: ErrorStatus
Type: BOOL
Pointer: false
Register: <undefined>
Comment: If no error active PASSED
If error active FAILED

Name: Save
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Save object data into remanent (permanent) memory .

Name: SetApplicationErrorFlag
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false

Hidden: false
Comment: Set / Reset application error flag
Input:: Name: SetValue
Type: BOOL
Pointer: false
Register: <undefined>
Comment: TRUE set flag
FALSE Reset flag

Name: SetCriticalErrorFlag
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Set / Reset critical error flag
Input:: Name: SetValue
Type: BOOL
Pointer: false
Register: <undefined>
Comment: TRUE set flag
FALSE Reset flag

Name: SetNextSfnObject
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Writes next object pointer for framework only.

This methode is only used by SFN_Logger to build
a linked list of all safan objects used in the application.

Input:: Name: p_Object
Type: ^SFN_Base
Pointer: true
Register: <undefined>

Name: GetInternalChannelPtr
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Convert a client pointer as it is used in application
to an internal client pointer (The way it is used internal by LASAL)
Methode used Workaround to solve
mismatch in :
#p_client^.items (accessed via pointer)
client.items

for structure items see CltCh
Input:: Name: p_Channel
Type: ^void
Pointer: true
Register: <undefined>
Output:: Name: p_IntChannel
Type: ^CltCh
Pointer: true
Register: <undefined>

Name: GetObjectName
Virtual: false

Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Get name of a server or client.
Workaround methode for _GetObjectName() to remove "_BASE\" from name string.

Input::
- Name: pO
- Type: ^Obj
- Pointer: true
- Register: <undefined>
- Comment: Pointer to object

Input::
- Name: pCh
- Type: ^void
- Pointer: true
- Register: <undefined>
- Comment: Pointer to server or client

Input::
- Name: pName
- Type: ^CHAR
- Pointer: true
- Register: <undefined>
- Comment: pName should point to memory of 255 bytes to be able to store the biggest string.

Name: SetInternalErrorFlag
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Set / Reset internal error flag

Input::
- Name: SetValue
- Type: BOOL
- Pointer: false
- Register: <undefined>
- Comment: TRUE set flag
  FALSE Reset flag

Name: UpdateObjectStatusError
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Update ObjectStatus error flag with actual error flags

Name: WriteActiveErrorWarning
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Write an active error or warning status
- Convert error or warning to an external id (incl group and level information)
- Write status in o_ErrorWarningStatus
- Store first error
- Log error message
- Set error flag in object status

Note: For a realtime task use the RT_WriteActiveErrorWarning()

Input::
- Name: IntErrorWarningId
- Type: UDINT

Comment: Internal Error or warning identifier
Note: without level and group info

Input:: Name: LogLevel
Type: gt_LogLevel
Pointer: false
Register: <undefined>
Comment: Error type (internal, application or critical)

Input:: Name: p_msg
Type: ^CHAR
Pointer: true
Register: <undefined>
Comment: Pointer to log message string

Name: WriteLogMessage
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Writes a log message.
If Lasal logger is connected message will be written to LasalLogger
If Lasal logger is not connected message will be written to screen

Input:: Name: msgId
Type: UDINT
Pointer: false
Register: <undefined>

Input:: Name: LogLevel
Type: gt_LogLevel
Pointer: false
Register: <undefined>
Comment: Loglevel

Input:: Name: p_msg
Type: ^CHAR
Pointer: true
Register: <undefined>
Comment: Pointer to message (read only)

Input:: Name: rtCall
Type: BOOL
Pointer: false
Register: <undefined>
Comment: Indicates methode is called from a Realtime Task.

Name: WriteLogRtData
UseOldHiddenInstImpl: false
Virtual: false
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Writes realtime log data to LasalLogger
2 values are logged: Identifier + value

Input:: Name: Id
Type: UDINT
Pointer: false
Register: <undefined>
Comment: Identifier of logged data

Input:: Name: Value
Type: DINT
Pointer: false
Register: <undefined>
Comment: Data

Name: ConnectClientChannels
UseOldHiddenInstImpl: false
Virtual: true
AWL implementation: false

UseBaseCmd: false
Hidden: false
Comment: Connect client channels.

If implemented the derived class can connect dynamically
client channels.

Input:: Name: p_Object
Type: pVoid
Pointer: false
Register: <undefined>
Comment: Pointer to object to connect with.

Name: ResetObject
UseOldHiddenInstImpl: false
Virtual: true
Global access: true
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Reset internal object variables to the situation just after
Init() was finished.
So object can restart on a reproducable way

Name: RT_WriteActiveErrorWarning
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Methode to log an active error/warning in a realtime task.
Input:: Name: IntErrorWarningId
Type: UDINT
Pointer: false
Register: <undefined>
Input:: Name: LogLevel
Type: gt_LogLevel
Pointer: false
Register: <undefined>
Input:: Name: p_msg
Type: ^CHAR
Pointer: true
Register: <undefined>

Name: StartObjectTimer
UseOldHiddenInstImpl: false
Virtual: false
Global access: false
AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Start object timer

Note:
StartTimer variable has to be allocated in the derived class.
(add class variable)
Input:: Name: p_StartTimer
Type: ^UDINT
Pointer: true
Register: <undefined>
Comment: Pointer to timer variable

Name: CheckObjectTimer
UseOldHiddenInstImpl: false
Global access: false

AWL implementation: false
CDecl: false
UseBaseCmd: false
Hidden: false
Comment: Check if timer has been elapsed

Also valid for wrap around situations

Note:
StartTimer variable has to be allocated in the derived class.
(add class variable)

Input:: Name: p_StartTimer
Type: ^UDINT
Pointer: true
Register: <undefined>
Comment: Pointer to timer variable

Input:: Name: timeOut
Type: UDINT
Pointer: false
Register: <undefined>
Comment: Timeout value used to check

Output:: Name: timeOutStatus
Type: BOOL
Pointer: false
Register: <undefined>
Comment: Timeout status
FALSE: Timer not finished yet
TRUE: Timer is finished

Types: Name: TestOverRide
Type: ENUM
Size: 4
Enum type: UDINT
Public: false
Elements: een, twee

Variables: Name: initCnt
Type: USINT
ElementType: USINT
Pointer: false
Comment: Counts the Init() calls
This variable can be used to init objects in a certain order if it is requirerd.

Name: nextSfnObj
Type: ^pVoid
ElementType: pVoid
Pointer: true
Comment: Next SFN_Base object in chain.
Content of this variable is only available if this object is connected to Safanlogger!!!!
It is used for an application to be able to go through all objects which is derived from SFN_Base and connected to the SFN_Logger.

Name: firstError
Type: UDINT
ElementType: UDINT
Pointer: false
Comment: First error detected since last error handling was finished.

Defines:
Files: C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_Base\SafanGeneral.h
C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_Base\SFN_Base.pdf

Dependencies:
Class: SFN_StringUtils
Types: ChDsc
Types: ChMeth

Types:	ChMode
Types:	ClsHdr
Types:	ClsHdrConst
Types:	CltCh
Types:	ConfStates
Types:	gpt_LogData
Types:	gt_LogData
Types:	gt_LogLevel
Types:	gt_ObjectStatus
Types:	Obj
Types:	ObjDsc
Types:	pChar
Types:	pClsHdr
Types:	pFct
Types:	pVoid
Types:	Revision
Types:	SvrCh
Types:	SvrDsc
Types:	void
File:	C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_Base\SafanGeneral.h
File:	C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_Base\SFN_Base.pdf
File:	C:\DevEnv\Sigmatek\Projects\SafanApplication\SafanPressBrake_NewEnv Jan 2014\SafanPressbrake\Class\SFN_StringUtils\SFN_StringUtils.pdf