```
__construct()
        CommonGLPI
+ $showdebug
+ $taborientation
+ $get_item_to_display_tab
+ static $rightname
# $type
# $displaylist
# static $showTitleInNavigation
 Header
# static $othertabs
+ __construct()
+ can($ID, $right, array
  &$input=null)
+ defineTabs($options=[])
+ addStandardTab($itemtype,
  array &$ong, array $options)
+ addDefaultFormTab
  (array &$ong)
+ getTabNameForItem
  (CommonGLPI $item,
  $withtemplate=0)
+ isNewItem()
+ showTabsContent($options=[])
+ getHeaderName()
+ display($options=[])
+ getErrorMessage($error,
  $object=")
+ static getTypeName
  ($nb=0)
+ static getFieldLabel()
+ static getType()
+ static canCreate()
+ static canView()
+ static canUpdate()
+ static canDelete()
+ static canPurge()
+ static registerStandardTab
 ($typeform, $typetab)
+ static getOtherTabs
  ($typeform)
  et 17 de plus...
# getFormHeaderToolbar()
        CommonDBTM
```

CommonGLPIInterface

+ \$fields + \$input + \$updates + \$oldvalues + \$dohistory + \$history\_blacklist + \$auto\_message\_on\_action + \$no\_form\_page + \$additional\_fields \_for\_dictionnary + \$taborientation + \$get\_item\_to\_display\_tab + \$deduplicate\_queued \_notifications + \$right + static \$undisclosedFields # \$fkfield # \$searchopt # \$usenotepad # static \$notable # static \$forward\_entity\_to # static \$plugins\_forward \_entity # static \$tables\_of # static \$foreign\_key \_fields\_of + getFromResultSet(\$rs) + getID() + post\_getFromDB() + showForm(\$ID, array \$options=[]) + getSNMPCredential() + getLockedFields() + post\_getEmpty() + getLogTypeID() + post\_deleteFromDB() + cleanDBonPurge() et 87 de plus... + static getTable(\$classname =null) + static forceTable( \$table) + static getForeignKeyField() + static getTableField (\$field, \$classname=null) + static getFromIter (DBmysqlIterator \$iter) + static unsetUndisclosedFields (&\$fields) + static getIndexName() + static isNewID(\$ID) + static getNameField() + static getCompleteNameField() et 26 de plus... # saveInput() # clearSavedInput() # restoreInput(array \$default=[]) # restoreSavedValues (array \$saved=[]) # cleanLockedsOnAdd() # cleanLockeds() # computeFriendlyName()

```
_RIGHTS
+ const HAVE_VIEW_RIGHT
 _ON_ITEM
+ const HAVE_SAME_RIGHT
  _ON_ITEM
+ static $canDeleteOnItemClean
+ static $disableAutoEntity
 Forwarding
+ getCloneRelations()
+ getConnexityItem(
  $itemtype, $items_id,
  $getFromDB=true, $getEmpty
 =true, $getFromDBOrEmpty=false)
+ checkAttachedItemChanges
 Allowed(array $input, array
 $fields)
+ tryEntityForwarding()
+ canConnexityItem(
  $methodItem, $methodNotItem,
  $item_right, $itemtype,
  $items_id, CommonDBTM &$item=null)
+ getHistoryChangeWhenUpdate
 Field($field)
+ getItemsForLog($itemtype,
  $items_id)
+ static getSQLCriteriaToSearch
 ForItem($itemtype, $items_id)
+ static getItemsAssociated
 To($itemtype, $items_id)
+ static getItemField
 ($itemtype)
+ static getItemFromArray
 ($itemtype, $items_id,
  array $array, $getFromDB
 =true, $getEmpty=true, $getFromDBOr
 Empty=false)
+ static canConnexity
 ($method, $item_right,
  $itemtype, $items_id)
+ static getConnexityMassive
 ActionsSpecificities()
+ static getMassiveActions
 ForItemtype(array &$actions,
  $itemtype, $is_deleted=false,
  CommonDBTM $checkitem=null)
+ static showMassiveActions
 SubForm(MassiveAction $ma)
+ static getConnexityInput ForProcessingOfMassiveActions
 ($action, CommonDBTM $item,
  `array $ids, array $input)
+ static processMassiveActions
 ForOneItemtype(MassiveAction
  $ma, CommonDBTM $item, array $ids)
```

CommonDBRelation

+ \$no\_form\_page + static \$itemtype\_1

CommonDBConnexity

+ const DONT\_CHECK\_ITEM

```
+ static $items_id_1
+ static $take_entity_1
+ static $checkItem_1
  _Rights
+ static $mustBeAttached_1
+ static $logs_for_item_1
+ static $log_history
 _1_add
+ static $log_history
  _1_update
+ static $log_history
 _1_delete
+ static $log_history
  _1_lock
  et 14 de plus...
# $_force_log_option
+ getOnePeer($number)
+ getFromDBForItems
  (CommonDBTM $item1,
  CommonDBTM $item2)
+ rawSearchOptions()
+ isAttach2Valid(array
  &$input)
+ isAttach1Valid(array
  &$input)
+ canRelationItem($method,
  $methodNotItem, $check
  _entity=true, $forceCheckBoth=false)
+ canCreateItem()
+ canViewItem()
+ canUpdateItem()
+ canDeleteItem()
  et 13 de plus...
+ static getOpposite
  (CommonDBTM $item, &
  $relations_id=null)
+ static canRelation
  ($method, $forceCheckBoth
  =false)
+ static canCreate()
+ static canView()
+ static canUpdate()
+ static canDelete()
+ static canPurge()
+ static getHTMLTableHeader
  ($itemtype, HTMLTableBase
  $base, HTMLTableSuperHeader
  $super=null, HTMLTableHeader
  $father=null, array $options=[])
+ static getRelationMassive
  ActionsSpecificities()
+ static showRelationMassive
  ActionsSubForm(MassiveAction
  $ma, $peer_number)
  et 6 de plus...
# static getDistinctTypesParams
  ($items_id, $extra_where=[])
         CommonITILActor
                           action
```

	+ \$auto_message_on_action
	+ const REQUESTER
	+ const ASSIGN
	+ const OBSERVER
	+ static \$checkItem_2 _Rights
	+ static \$logs_for_item_2
	+ getActorForeignKey()
	+ isAttach2Valid(array &\$input)
	+ canUpdateItem()
	+ canDeleteItem()
_	+ showUserNotificationForm (\$ID, \$options=[])
	+ showSupplierNotification Form(\$ID, \$options=[])
7	+ prepareInputForAdd (\$input)
	+ prepareInputForUpdate (\$input)
	+ post_addItem()
1	+ static getItilObjectForeign Key()
	Λ

Change_Group		Change_Supplier			
+ static \$itemtype_1		+ static \$itemtype_1			
+ static \$items_id_1		+ static \$items_id_1			
+ static \$itemtype_2		+ static \$itemtype_2			
+ static \$items_id_2		+ static \$items_id_2			

Problem\_User + static \$itemtype\_1 + static \$items\_id\_1 + static \$itemtype\_2 + static \$items\_id\_2

Supplier\_Ticket + static \$itemtype\_1 + static \$items\_id\_1 + static \$itemtype\_2 + static \$items\_id\_2 + post\_addItem()

Ticket\_User + static \$itemtype\_1 + static \$items\_id\_1 + static \$itemtype\_2 + static \$items\_id\_2 + post\_addItem()