CommonGLPIInterface _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 + \$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() CommonDBConnexity + const DONT_CHECK_ITEM _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 + 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=[]) CommonItilObject_Item + static getSpecificValueTo Display(\$field, \$values, array \$options=[]) + static getSpecificValueTo Select(\$field, \$name=", \$values=", array \$options=[]) Item_Ticket + static \$itemtype_1 + static \$items_id_1 + static \$itemtype_2 + static \$items_id_2 + static \$checkItem_2 Rights + getForbiddenStandardMassive Action() Item Problem + canCreateItem() + static \$itemtype_1 + post_addItem() + static \$items_id_1 + post_purgeItem() + static \$itemtype_2 + prepareInputForAdd (\$input) + static \$items_id_2 + getTabNameForItem + static \$checkItem_2 (CommonGLPI \$item, Rights \$withtemplate=0) + getForbiddenStandardMassive + getForbiddenStandardMassive + rawSearchOptions() Action() + addMessageOnAddAction() prepareInputForAdd + addMessageOnPurgeAction() (\$input) static getTypeName + static getTypeName (\$nb=0)(\$nb=0)static showForProblem + static showItemToAdd (Problem \$problem) (\$tickets_id, \$itemtype, \$items_id, \$options) static displayTabContent ForItem(CommonGLPI \$item, ForItem(CommonGLPI \$item, + static showForTicket \$tabnum=1, \$withtemplate=0) \$tabnum=1, \$withtemplate=0) (Ticket \$ticket) + static displayTabContent ForItem(CommonGLPI \$item, \$tabnum=1, \$withtemplate=0) + static getUsedItems (\$tickets_id) + static showMassiveActions SubForm(MassiveAction \$ma) + static processMassiveActions ForOneItemtype(MassiveAction

Change_Item

+ static \$itemtype_1

+ static \$items_id_1

+ static \$itemtype_2

+ static \$items_id_2

Rights

Action()

(\$input)

(\$nb=0)

+ static \$checkItem_2

prepareInputForAdd

static getTypeName

+ static showForChange

+ static displayTabContent

\$ma, CommonDBTM \$item, array \$ids)

(Change \$change)