```
_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)
            CommonDBChild
 + static $itemtype
 + static $items_id
 + static $checkParentRights
 + static $mustBeAttached
 + static $logs_for_parent
 + static $log_history_add
 + static $log_history
   _update
 + static $log_history
   _delete
 + static $log_history_lock
 + static $log_history
    _unlock
 + canCreateItem()
 + canViewItem()
  canUpdateItem()
 + canDeleteItem()
 + canChildItem($methodItem,
   $methodNotItem)
 + getItem($getFromDB
   =true, $getEmpty=true)
 + recursivelyGetItems()
 + getEntityID()
 + isEntityAssign()
 + maybeRecursive()
   et 12 de plus...
 + static getSQLCriteriaToSearch
   ForItem($itemtype, $items_id)
 + static canCreate()
 + static canView()
 + static canUpdate()
 + static canDelete()
 + static canPurge()
 + static canChild($method)
+ static displayRecursiveItems
   (array $recursiveItems,
   $elementToDisplay, bool $display=true)
 + static getJSCodeToAddFor
   ItemChild($field_name,
   $child_count_js_var)
 + static showAddChildButton
   ForItemForm(CommonDBTM
   $item, $field_name, $canedit
   =null, $display=true)
 + static getItemField
   ($itemtype)
 # autoinventoryInformation()
```

CommonGLPIInterface

CalDAVCompatibleItemInterface

+ static getGroupItemsAsVCalendars

+ static getUserItemsAsVCalendars

+ getAsVCalendar()

(\$groups\_id)

(\$users\_id)

+ getInputFromVCalendar

(VCalendar \$vcalendar)

ProjectTask

+ \$dohistory

# \$team

# \$usenotepad + canViewItem() + canUpdateItem() + cleanDBonPurge() + getRights(\$interface

='central')

(\$nb=0)
+ static getIcon()
+ static canPurge()
+ static canView()
+ static canCreate()
+ static canUpdate()

+ defineTabs(\$options=[])
+ post\_getFromDB()
+ post\_getEmpty()
+ isInTheTeam()
+ getTeamCount()
+ post\_purgeItem()
et 12 de plus...
+ static getTypeName

+ static displayTabContent
 ForItem(CommonGLPI \$item,
 \$tabnum=1, \$withtemplate=0)

+ static populateNotPlanned

+ static getGroupItemsAsVCalendars

+ static getUserItemsAsVCalendars

(\$options=[])

(\$groups\_id)

(\$users\_id)

+ static getTeamRoles()+ static getTeamRoleName (int \$role, int \$nb=1)+ static getTeamItemtypes()

+ \$can\_be\_translated+ const READMY+ const UPDATEMY+ static \$itemtype+ static \$items\_id+ static \$rightname