

# Command

Code

```
enum
{
  CGROUP_DEFAULT      = 0,
  CGROUP_ARREST      = 1,
  CGROUP_FIREHOSE    = 2,
  CGROUP_CARRY_PERSON = 3,
  CGROUP_PULLING     = 4,
  CGROUP_DLK_BASKET  = 5,
  CGROUP_LIGHT       = 6,
  CGROUP_CANNON_WAW  = 7,
  CGROUP_DRAW_WEAPON = 8,
  CGROUP_INSTALL     = 9,
  CGROUP_SEARCHLIGHT = 10,
  CGROUP_SHIPDROP    = 11,
  CGROUP_UNLOADVEHICLE = 12,
  CGROUP_BUCKET      = 13,
  CGROUP_DOG         = 14,
  CGROUP_PICKUP      = 15,
  CGROUP_CRANE       = 16,
  CGROUP_GETEQUIPMENT = 100
};
```

CommandGroup

Alles anzeigen

Code

```

enum
{
    RESTRICT_NONE,
    RESTRICT_LOCKED,
    RESTRICT_USABLE,
    RESTRICT_BULLDOZABLE,
    RESTRICT_TRANSPORTABLE,
    RESTRICT_ACCESSIBLE,
    RESTRICT_COOLABLE,
    RESTRICT_SHOOTABLE,
    RESTRICT_CUTABLE,
    RESTRICT_MEGAPHONE,
    RESTRICT_RECOVERABLE,
    RESTRICT_FLOTSAM,
    RESTRICT_NOTDESTROYED,
    RESTRICT_CARRYABLE_BD,
    RESTRICT_HASFIREEXT,
    RESTRICT_HASSHEARS,
    RESTRICT_HASCHAINSAW,
    RESTRICT_HASJUMPPAD,
    RESTRICT_HASROADBLOCK,
    RESTRICT_HASFLASHGRENADE,
    RESTRICT_HASFIREAXE,
    RESTRICT_HASFIREHOSE,
    RESTRICT_PULLABLE,
    RESTRICT_BURNING,
    RESTRICT_NOTBURNING,
    RESTRICT_LINKED,
    RESTRICT_NOTLINKED,
    RESTRICT_INJURED,
    RESTRICT_NOTINJURED,
    RESTRICT_ARRESTED,
    RESTRICT_NOTARRESTED,
    RESTRICT_IDLE,
    RESTRICT_SELFEXECUTE
};

```

CommandRestriction

Alles anzeigen

Code

```

enum
{
    RESTRICT2_NONE,
    RESTRICT2_ISHYDRANT,
    RESTRICT2_ISHYDRANTORSELF
};

```

CommandRestriction2

Code

```
enum
{
  CPE_CUTABLE_OBJECTS,
  CPE_LOCKED_HOUSE,
  CPE_HOUSE_FOR_JUMPPAD,
  CPE_CAR_W_ENCLOSED_PERSON,
  CPE_FREE_HOSE_CONNECTION,
  CPE_COOLABLE_OBJECTS,
  CPE_NONIJURED_PERSON,
  CPE_FREE_DLK,
  CPE_ACCESSIBLE_HOUSE,
  CPE_EXTINGUISHABLE_OBJECTS,
  CPE_OBJECTS_PULLABLE,
  CPE_REPAIRABLE_CAR,
  CPE_ASKABLE_PERSON,
  CPE_INJURED_PERSON,
  CPE_WOUNDED_SQUAD,
  CPE_USABLE_WM_OBJECTS,
  CPE_DROWNING_PERSONS,

  CPE_ALWAYS
};
```

CommandPossibleExists

Alles anzeigen

Code

```
enum
{
  CFN_IGNORE,
  CFN_NEEDED,
  CFN_FAIL
};
```

CommandFirehoseNeed

Code

```

class                                CommandScript
{
    CommandScript(const          char          *Class_,          const          char          *Object_);
    virtual                    ~CommandScript();

    void                        SetBitmapName(const          char          *Name);
    void                        SetActionCursor(const          char          *Name);
    void                        SetIcon(const          char          *Name);
    void                        SetCursor(const          char          *Name);
    void                        SetCommandName(const          char          *Name);
    void                        SetInfotextVisible(bool          Visible);
    void                        SetValidTargets(int          Types_);
    void                        SetDoubleClickable(bool          DbfClickable_);
    void                        SetActivationByLeftClick(bool          byLeftClick_);
    void                        SetGroupID(int          ID_);
    void                        SetGroupLeader(bool          IsLeader_);

    void                        SetRestrictions(int          Restrictions_);
    void                        AddRestriction(CommandRestriction          Restriction_);
    void                        RemoveRestriction(CommandRestriction          Restriction_);
    int                          GetRestrictions()          const;

    void                        SetRestrictions2(int          Restrictions_);
    void                        AddRestriction2(CommandRestriction2          Restriction_);
    void                        RemoveRestriction2(CommandRestriction2          Restriction_);
    int                          GetRestrictions2()          const;

    void                        SetHighlightingEnabled(bool          enabled_);
    void                        SetKeepAsCurrentCommand(bool          keep_);
    void                        SetDeselectCaller(bool          deselect_);
    void                        SetPossibleExists(int          Exists_);
    void                        SetPossibleEquipment(int          Equipment_);
    void                        SetPossibleCallers(int          Types_);
    void                        SetNeedsConnectedHose(CommandFirehoseNeed          NeedsHose_);
    void                        SetNeedsCarWithFlagSet(int          Flag_);
    void                        SetPriority(int          Priority_);
    void                        SetSelfClickActivation(bool          activate_);
};

```

Alles anzeigen  
Code

```

enum                                MoveMode
{
    MOVE_ABORT,
    MOVE_TO_POSITION,
    MOVE_INTO_HOUSE,
    MOVE_HOUSE_TO_HOUSE,
    MOVE_HOUSE_TO_POSITION
};

```

Code

```

struct                                MoveResult
{
    MoveMode                          Mode;
    Vector                             Target;
    Vector                             Intermediat1;
    Vector                             Intermediat2;
    bool                               UnInstall,
    OpenHouse                          BasketDown;
    OpenHouse                          EnterHouse;
    OpenHouse                          LeaveHouse;
};

```

## Code

```
namespace Commands
{
    MoveResult CheckMoveConditions(GameObject *Caller, Actor *Target, int ChildID);
    bool       IsEnterCarPossible(GameObject *Caller);
};
```