

# Actor

Code

```
enum
{
  ACTOR_UNKNOWN,
  ACTOR_FLOOR,
  ACTOR_OBJECT,
  ACTOR_HOUSE,
  ACTOR_VEHICLE,
  ACTOR_PERSON,
  ACTOR_LIQUID,
  ACTOR_VIRTUAL,
  ACTOR_FIREOBJECT,
  ACTOR_PATH,
  ACTOR_FORCEFIELD,
  ACTOR_FORCEVOLUME,
  ACTOR_TRIGGER,
  ACTOR_STREET,
  ACTOR_OPEN_HOUSE,
  ACTOR_SPAWNPOINT,
  ACTOR_DETAILPOLYGON,
  ACTOR_WAITINGPOINT,
  ACTOR_AMBIENTPOLYGON,
  ACTOR_BRIDGEINSTALLPOINT,
  ACTOR_STOPPINGPOINT
};
```

ActorType

Alles anzeigen

Code

```
enum
{
    TARGET_ANY,
    TARGET_EQUIPMENTDOOR,
    TARGET_SHEARSDOOR,
    TARGET_PASSENGERDOOR,
    TARGET_REARDOOR,
    TARGET_FIREHOSE_HOOKUP,
    TARGET_ENGINE,
    TARGET_EXTINGUISH,
    TARGET_ENGINE_EXTINGUISH,
    TARGET_AXE,
    TARGET_CHAINSAW,
    TARGET_ENTRANCEDOOR,
    TARGET_MEGAPHONE_DISTANCE,
    TARGET_DLK_BASKET,
    TARGET_LOADUP,
    TARGET_ENTRY_WINDOW_PARKING,
    TARGET_DLK_BASKET_BASE,
    TARGET_TREATMENT,
    TARGET_UNLOAD,
    TARGET_ENTRY_WINDOW,
    TARGET_INSTALL_FGRB,
    TARGET_SHOOT,
    TARGET_FOLLOW,
    TARGET_DLK_EXTINGUISH,
    TARGET_UNLOAD_TFMB,
    TARGET_FLAME_EFFECT,
    TARGET_CROSS_BRIDGE,
    TARGET_OPPOSITE_BRIDGE,
    TARGET_EXTINGUISH_PERSON,
    TARGET_PONTON_BRIDGE,
    TARGET_TOUCHPERSON,
    TARGET_USE,
    TARGET_OBJECTSURFACE,
    TARGET_FREE_CONNECTOR,
    TARGET_RANDOM,
    TARGET_HOUSE_SAFE_DISTANCE,

    TARGET_MAX
};
```

Alles anzeigen  
Code

```

enum TerrainClass
{
    TERRAIN_ANYTHING,
    TERRAIN_CAR,
    TERRAIN_TRUCK,
    TERRAIN_OFFROAD,
    TERRAIN_TRACKVEHICLE,
    TERRAIN_BOAT,
    TERRAIN_SHIP,
    TERRAIN_DIVER,
    TERRAIN_AIRPLANE,
    TERRAIN_CIVILIAN,
    TERRAIN_SQUAD,
    TERRAIN_SQUADCAR,
    TERRAIN_HELILANDING,
    TERRAIN_ANIMAL,
    TERRAIN_VIRTUALFLOOR,
    TERRAIN_TRAFFIC,

    TERRAIN_NUMCLASSES
};

```

Alles anzeigen  
Code

```

class Actor
{
    const char *GetName();

    virtual Vector GetPosition() const;
    virtual void SetPosition(const Vector &v_);

    Vector GetTargetPoint(Actor *Initiator, TargetPoint Pnt);
    Vector GetTargetPoint(Actor &Initiator, TargetPoint Pnt);

    float GetBoundingRadius();
    float GetBoundingRadiusXY();
    float GetBoundingRadiusDistXYToObject(Actor *Target_);
    virtual void DrawBoundingBox(unsigned char red_=255, unsigned char green_=255, unsigned char blue_=255);
    bool SetVirtualObjectTerrain(const char* terrain_);
    bool SetTerrain(TerrainClass terrain_);
    bool RemoveFromRouter();

    virtual bool IsValid() const; //!<returnstrueifthisactorhasavalidworldrepresentation
    ActorType GetType() const;
    void Show();
    void Hide();
    bool IsHidden();
    int GetID() const;
    bool HasName(const char *Name_);
    bool HasNamePrefix(const char *Name_);

    void SetUserData(int Data);
    int GetUserData() const;

    bool IsInsideVirtualObject(Vector &pos);
};

```

Alles anzeigen