



## HardFate-Engine (HF) Design Sheet

### Einleitung

Die HardFate-Engine (im folgenden HF genannt) wird eine für Grafikanwendungen konzipierte Engine, welche alle benötigten Komponenten abdeckt (Grafik, Sound und Input).

### Namespaces

Die komplette Engine wird im Namespace *HF* angesiedelt. Daneben gibt es 4 weitere Namespaces im *HF*-Namespace:

- HF::Core** In diesem Namespace befinden sich alle eigenen Typen der Engine wie zum Beispiel die Vektor- und Matrixklassen.
- HF::Graphic** In diesem Namespace befinden sich alle Klassen die im Zusammenhang mit der Ausgabe auf dem Bildschirm stehen. Darunter fallen unter anderem die Klasse zur Initialisierung von Direct3D, zur Darstellung von Modellen, Schriften, usw. Ausserdem enthält dieser Namespace einen weiteren namens *HF::Graphic::GUI* für das Benutzerinterface (Windows, Buttons, ...).
- HF::Media** In diesem Namespace befinden sich alle Klassen zum Laden und Abspielen von Sounds (evt. später auch Videos) in verschiedenen Formaten, sowie zum Laden von Texturen.
- HF::Input** In diesem Namespace befinden sich alle Klassen zum Abfragen der Eingabe von allen kompatiblen Eingabegeräten.

### Klassenhierarchie

Die Einzüge stellen die Hierarchie dar. Weiter eingezogene Klassen sind von den weniger weit eingezogenen abgeleitet. Namespaces sind fett. Hinter den Objekten steht jeweils eine sehr kurze Beschreibung zur Klasse. Eine detailliertere Beschreibung folgt weiter unten.

#### HF

Memory	Speicherverwaltung (statische Klasse)
Config	Zum Auslesen diverser Systeminfos (Singleton)
Log	Zum führen einer Logdatei (Singleton)
<b>Core</b>	
Matrix	4x4-Matrix
Vec2	2D-Vektor
Vec3	3D-Vektor
Color	Farbe
Angle	Winkel
Triangle	Dreieck
List	Doppelt verkettete Liste
<b>Graphic</b>	
Main	Initialisierung von Direct3D (Singleton)
RessourceManager	Verwaltet Texturen, usw.
Ressource	Abstrakte Basisklasse für Ressourcen der Grafikhardware
Texture	Adapter zum Arbeiten mit Texturen
VertexBuffer	Adapter zum Arbeiten mit VertexBuffern
IndexBuffer	Adapter zum Arbeiten mit IndexBuffern
Object	Abstrakte Basisklasse für alle Grafikobjekte
2DObject	Abstrakte Basisklasse für alle 2D-Grafikobjekte
Sprite	Einfaches 2D-Objekt
Text	2D-Text
3DObject	Abstrakte Basisklasse für alle 3D-Grafikobjekte
Model	Modell-Klasse
ModelEx	Modell-Klasse mit erweiterten Funktionen
UserDefined	Für einfache Modelle, die direkt gecodet werden können.
TextEx	3D-Text
Lightning	Beleuchtungs-Klasse (Singleton)



Light	Adapter zum Arbeiten mit Lichter
Font	Schriftarten
Camera	Kamera-Klasse
<b>GUI</b>	
Main	GUI-Hauptklasse (Singleton)
Element	Element der GUI
Window	Fenster
Button	Knopf
EditBox	EditBox
...	und viele mehr
<b>Media</b>	
Main	Initialisierung von DirectSound (Singleton)
Sound	Abstrakte Basisklasse für ein Musikstück
SoundEffect	Klasse für kurze Musikstücke
SoundEffectEx	Klasse für kurze Musikstücke mit erweiterten Funktionen
Music	Klasse für lange Musikstücke
MusicEx	Klasse für lange Musikstücke mit erweiterten Funktionen
Image	Klasse zum Laden von Bildern in diversen Formaten
<b>Input</b>	
Main	Initialisierung von DirectInput (Singleton)
Device	Abstrakte Basisklasse für ein DirectInput-Gerät
Keyboard	Tastatur
Mouse	Maus
Joystick	Joystick
Element	Abstrakte Basisklasse für ein einzelnes „Eingabe-Element“
Button	Knopf
Axis	Achse
Slider	Slider
POV	POV

## Klassen

Kommt noch...