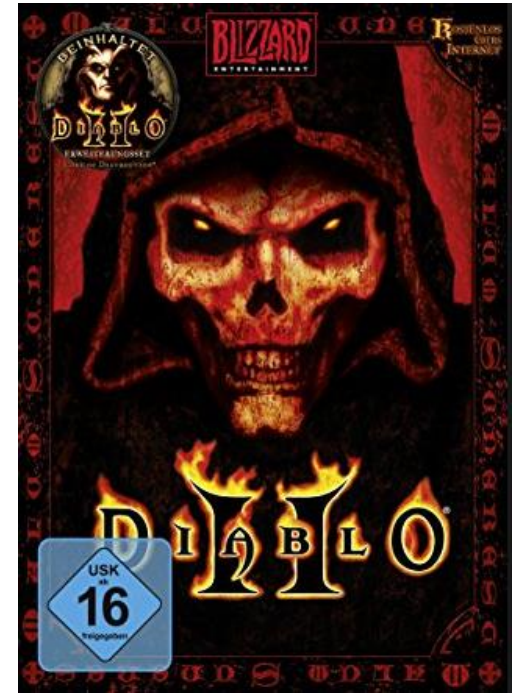


Diablo II Modding

Am Beispiel von Median XL



Agenda

1. Diablo II
2. Diablo II Modding und Tools
3. Modding Diablo II
 - Sprites und Animationen
 - Strings und Internationalisierung
 - Items, Skills, Treasureclasses
 - Monster und deren AI
4. Einblick in die Beta von Diablo II –Median XL Sigma (closed Beta)

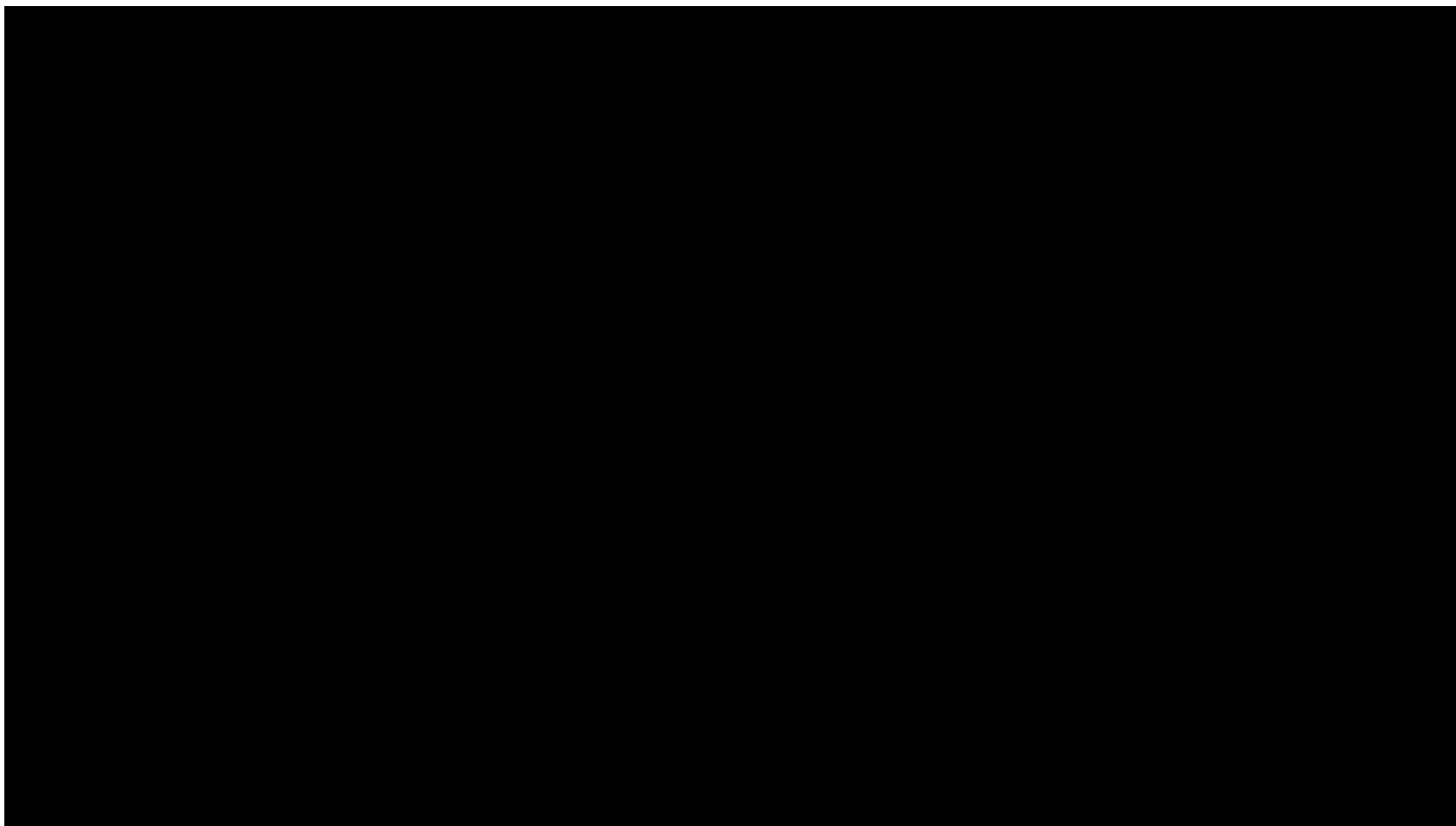
1. Diablo II

- Action Rollenspiel von Blizzard
- Veröffentlicht 2000, Lord of Destruction (Expansion Set) 2001

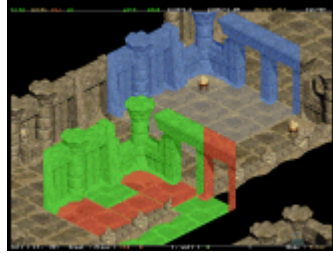


- Letzter Offizieller Patch, Version 1.14d, Veröffentlicht im Juni 2016

Gameplay –Median XL



2. Diablo II Modding und Tools

- Die Aktuelle Situation ist etwas durcheinander...
 - Für alle Verschiedenen Aspekte des Moddings, gibt es Verschiedene Tools
- Beispiele:
 - Extrahieren der Dateien zum Modden mittels: **MPQ Editor** 
 - Erstellen von Maps mit **Win_DS1 Map Editor** 
 - Ändern von Sprites, Animationen: **SixDice** 
 - Excel, D2Excel  
 - ...

3. Modding Diablo II

- Getting Started:

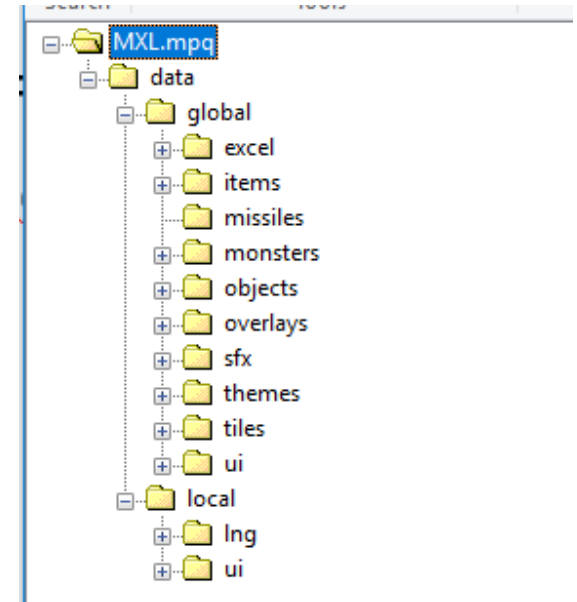
- Finde „patch_d2.mpq“ im Installationsfolder von D2
- Extrahiere den Inhalt in einen Ordner „data“ im Root Installationsverzeichnis

- Erstelle ein Verknüpfung der „Diablo II .exe“
- Starte mit extra Flags

- **-direct**
- **-txt**
- **-w**

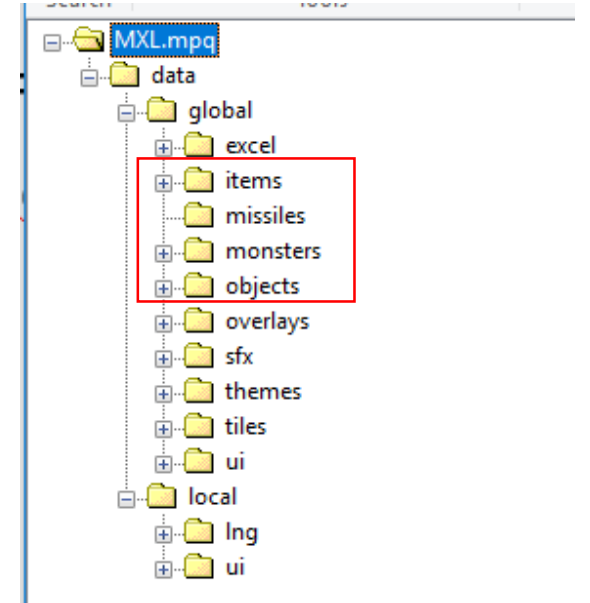
Zielort: Diablo II

Ziel: "D:\Diablo II\Diablo II.exe" -w -direct|



3. Modding Diablo II – Sprites und Animationen

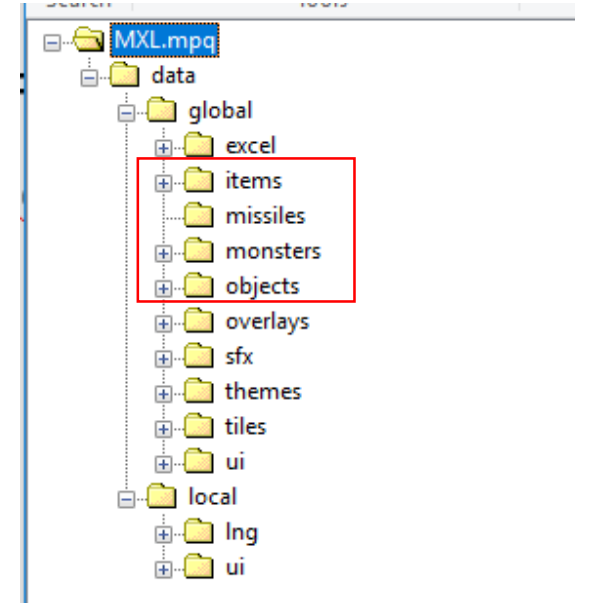
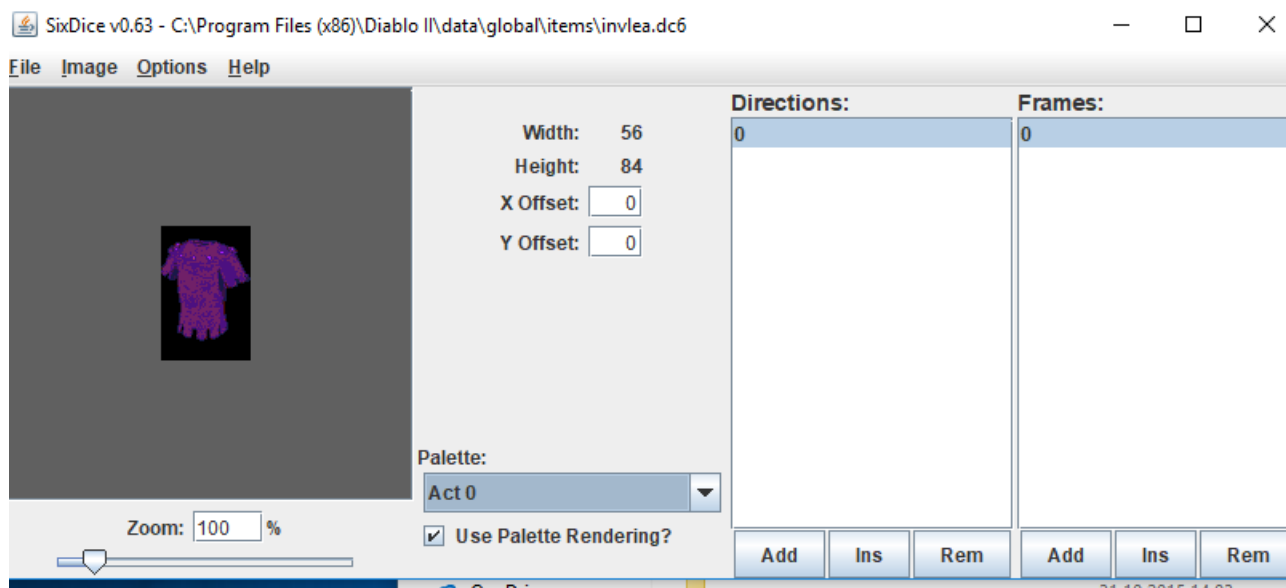
- Genutzt für:
 - „Missiles“ von Skills
 - Monster
 - UI
 - Items
 - ...





3. Modding Diablo II – Sprites und Animationen

- Diablo II nutzt dafür 2 verschiedene Formate
 - .DC6 (v. A. statische Bilder, unkomprimiert)
 - .DCC (Animationen, komprimiert)



3. Modding Diablo II – Sprites und Animationen

- Austausch einzelner Dateien bewirkt direkte Veränderung im Spiel:



Invlea.dc6



3. Modding Diablo II – Sprites und Animationen

- Einfache Veränderungen können durch das Anpassen und Austauschen einzelner Dateien vorgenommen werden
- Viele Monster bestehen allerdings aus mehreren Teilen

- The COF folder : it contains the cof files needed by Diablo II to correctly place the different parts of the animation together.
- The S1 and TR folders : they contain the dcc files of the BH token.

If you look at another token folder, you'll see many other folders like RH,LH,SH etc ...
Whats the meaning of these codes ?

- **HD** - Contains the Head part of the animation.
- **TR** - Contains the Torso part of the animation.
- **LG** - Contains the Leg part of the animation.
- **RA** - Contains the Right Arm part of the animation.
- **LA** - Contains the Left Arm part of the animation.
- **RH** - Contains the Right Hand part of the animation.
- **LH** - Contains the Left Hand part of the animation.
- **SH** - Contains the Shield part of the animation.
- **S1** - Contains the 1st Overlay part of the animation.
- **S2** - Contains the 2nd Overlay part of the animation.
- **S3** - Contains the 3rd Overlay part of the animation.
- **S4** - Contain the 4th Overlay part of the animation.
- **S5** - Contain the 5th Overlay part of the animation.
- **S6** - Contain the 6th Overlay part of the animation.
- **S7** - Contain the 7th Overlay part of the animation.
- **S8** - Contain the 8th Overlay part of the animation.

- Tutorials unter: <https://d2mods.info/forum/viewtopic.php?f=6&t=23089>

3. Modding Diablo II – Sprites und Animationen

- Für Animationen nutzt Blizzard „Hardcoded“ Codes
- Zb. Die Datei (Leerzeichen manuell hinzugefügt):

BH TR LIT A1 HTH.DCC

- **Monster ID (oder Token)**
- **Body Part**
- **Armor Typ (lit=Lite armor, med=Medium armor, hvy=Heavy armor;)**
- **Monster & Player Mode**
- **Weapon/Hitclass**

3. Modding Diablo II – Sprites und Animationen

- Für Animationen nutzt Blizzard „Hardcoded“ Codes
- Zb. Die Datei (Leerzeichen manuell hinzugefügt):

BH TR LIT A1 HTH.DCC

- **Monster & Player Mode**

Player & Monster Modes:

Name	Token
Death	DT
Neutral	NU
Walk	WL
Run	RN
Get Hit	GH
Town Neutral	TN
Town Walk	TW
Attack1	A1
Attack2	A2
Block	BL
Cast	SC
Throw	TH
Kick	KK
Skill1	S1
Skill2	S2
Skill3	S3
Skill4	S4
Dead	DD
Sequence	??
Knock Back	GH

3. Modding Diablo II – Sprites und Animationen

- Für Animationen nutzt Blizzard „Hardcoded“ Codes
- Zb. Die Datei (Leerzeichen manuell hinzugefügt):

BH TR LIT A1 HTH.DCC

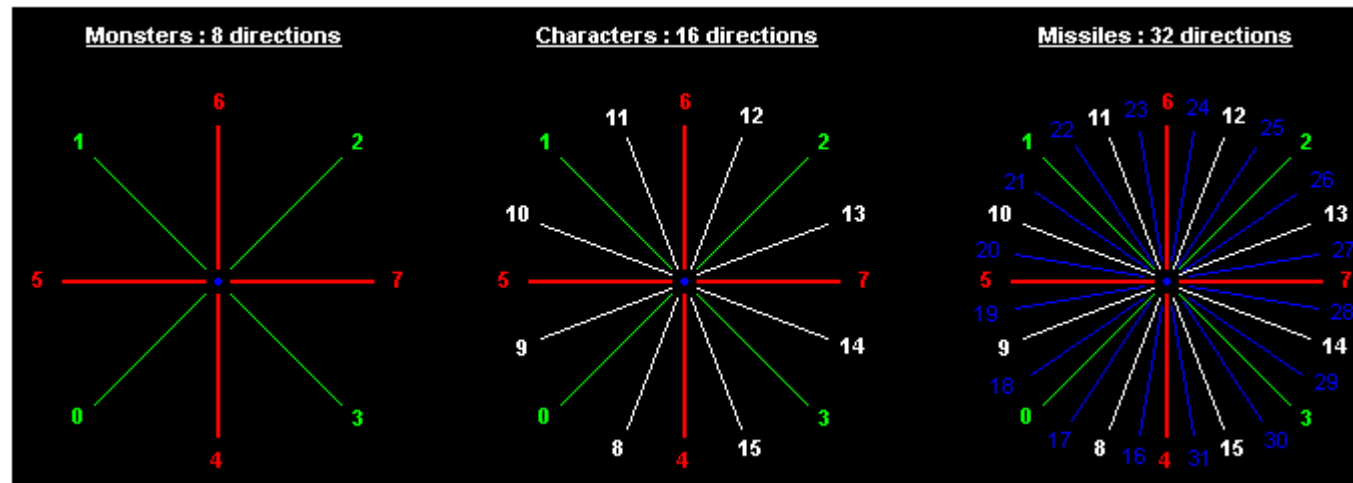
- **Weapon/Hitclass**

Hit & Weapon Classes:

Name	Code
Hand-to-Hand	HTH
One Hand Swing vs. Small	1hss
One Hand Swing vs. Large	1hsl
Two Hand Swing vs. Small	2hss
Two Hand Swing vs. Large	2hsl
One Hand Thrust	1ht
Two Hand Thrust	2ht
Club	club
Staff	staf
Bow	bow
Crossbow	xbow
Claw	claw
Overlay	over
Left Jab Right Swing	1js
Left Jab Right Thrust	1jt
Left Swing Right Swing	1ss
Left Swing Right Thrust	1st
One Hand-to-Hand	ht1
Two Hand-to-Hand	ht2

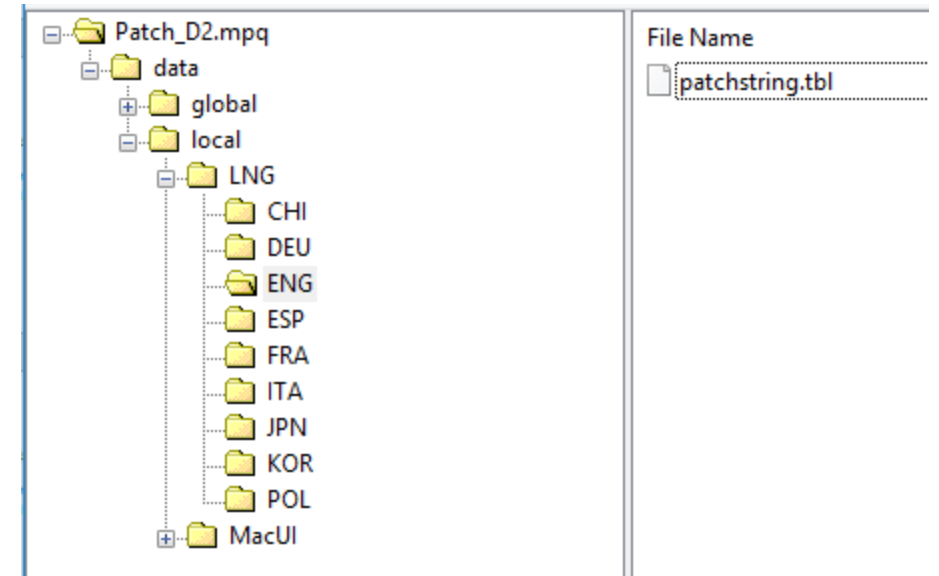
3. Modding Diablo II – Sprites und Animationen

- In Diablo II sind alle Animation in Richtungen unterteilt, angepasst an den „3D Isometrischen Style“



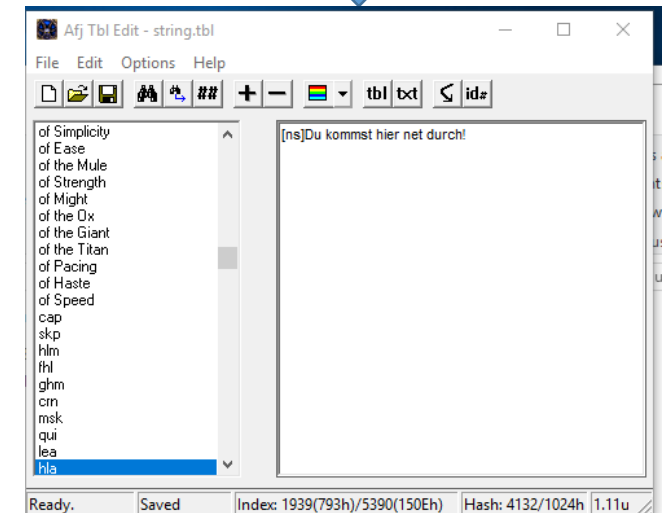
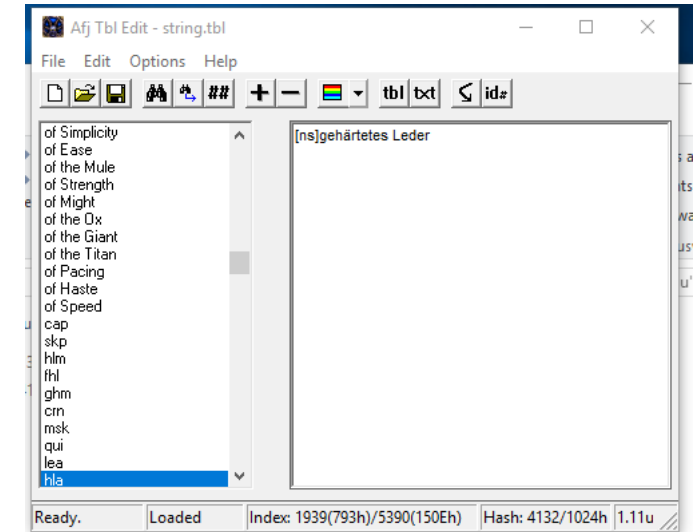
3. Strings und Internationalisierung

- In Diablo II sind alle Strings Extern in dedizierten Dateien gespeichert. („.tbl“)
- Eine .tbl Datei ist eine Art gehashte Struktur, für jeden String existieren:
 - Hash → Key → Value (String)
- 3 Solcher Dateien für alle Strings
 - String.tbl (Index 0-10000)
 - PatchString.tbl (10001-20000)
 - Expansionstring.tbl (LoD, 20001-30000)



3. Strings und Internationalisierung

- Editieren mit Afj Tbl Edit, Änderungen sind direkt im Spiel sichtbar



3. Items

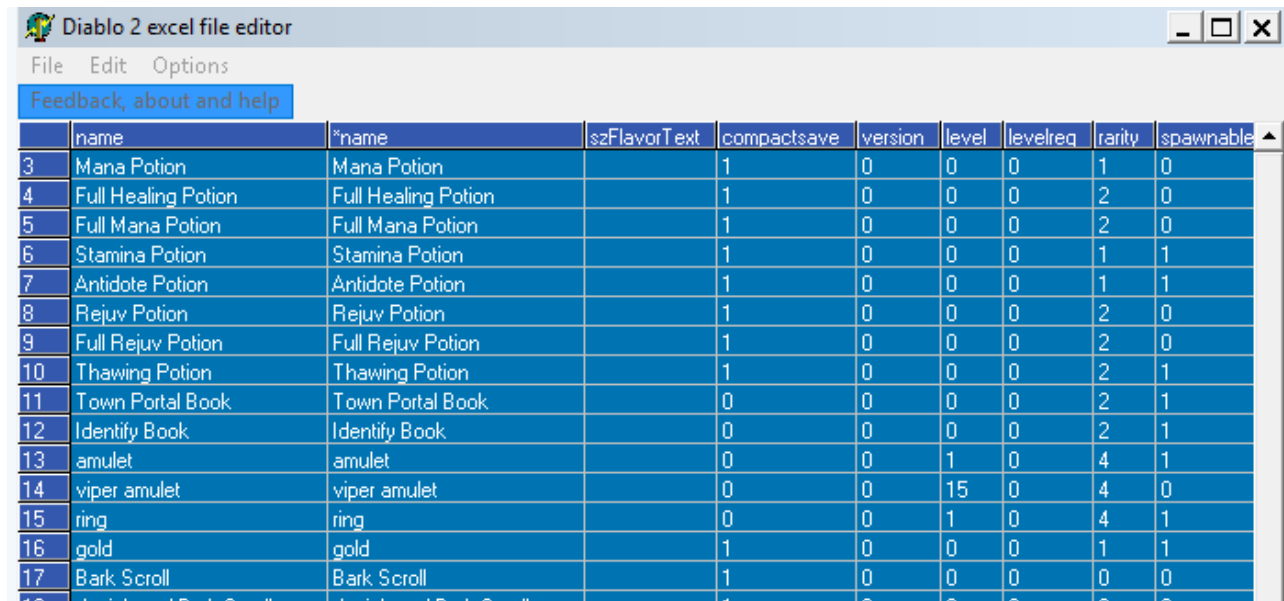
- In Diablo II gibt es Items in verschiedenen Qualitätsstufen:

1. Misc
2. White/Grey
3. Magic
4. Rare
5. Crafted
6. Tempered
7. Set
8. Unique



3. Items

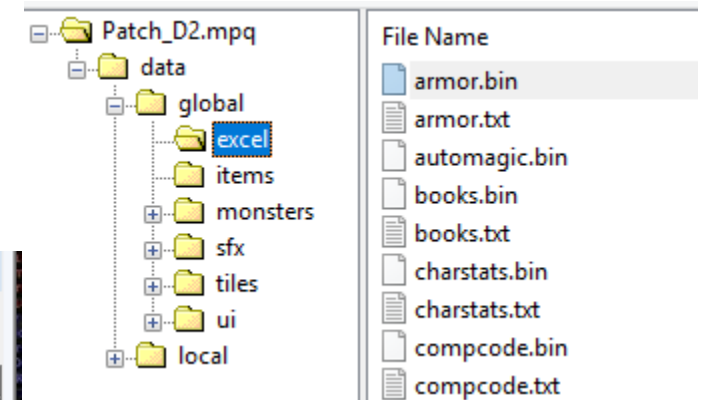
- Die Eigenschaften der Items sind in Tabellen gehalten („.txt“)
- Tabellen können mit D2Excel oder MSEXcel editiert werden



The screenshot shows the 'Diablo 2 excel file editor' window. It has a menu bar with 'File', 'Edit', and 'Options'. Below the menu is a blue button labeled 'Feedback, about and help'. The main area contains a table with 10 columns: 'name', '*name', 'szFlavorText', 'compactsave', 'version', 'level', 'levelreq', 'rarity', and 'spawnable'. The table lists various items like 'Mana Potion', 'Full Healing Potion', 'Full Mana Potion', 'Stamina Potion', 'Antidote Potion', 'Rejuv Potion', 'Full Rejuv Potion', 'Thawing Potion', 'Town Portal Book', 'Identify Book', 'amulet', 'viper amulet', 'ring', 'gold', and 'Bark Scroll'.

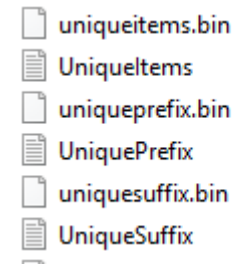
	name	*name	szFlavorText	compactsave	version	level	levelreq	rarity	spawnable
3	Mana Potion	Mana Potion		1	0	0	0	1	0
4	Full Healing Potion	Full Healing Potion		1	0	0	0	2	0
5	Full Mana Potion	Full Mana Potion		1	0	0	0	2	0
6	Stamina Potion	Stamina Potion		1	0	0	0	1	1
7	Antidote Potion	Antidote Potion		1	0	0	0	1	1
8	Rejuv Potion	Rejuv Potion		1	0	0	0	2	0
9	Full Rejuv Potion	Full Rejuv Potion		1	0	0	0	2	0
10	Thawing Potion	Thawing Potion		1	0	0	0	2	1
11	Town Portal Book	Town Portal Book		0	0	0	0	2	1
12	Identify Book	Identify Book		0	0	0	0	2	1
13	amulet	amulet		0	0	1	0	4	1
14	viper amulet	viper amulet		0	0	15	0	4	0
15	ring	ring		0	0	1	0	4	1
16	gold	gold		1	0	0	0	1	1
17	Bark Scroll	Bark Scroll		1	0	0	0	0	0
18	...	...		1	0	0	0	0	0

Misc.txt



Modifizieren der Textdateien

- Modding von Diablo II wird in 2 Bereiche unterteilt:
 1. Hardcoding: Reverse Engineering, Code injection etc...
 2. Softcoding: Editieren der Txt-Dateien, Tbl, Monster, etc...
- Die Textdateien werden beim Start des Spiels mit dem Flag (-txt) in gleichnamige .bin Strukturen umgewandelt



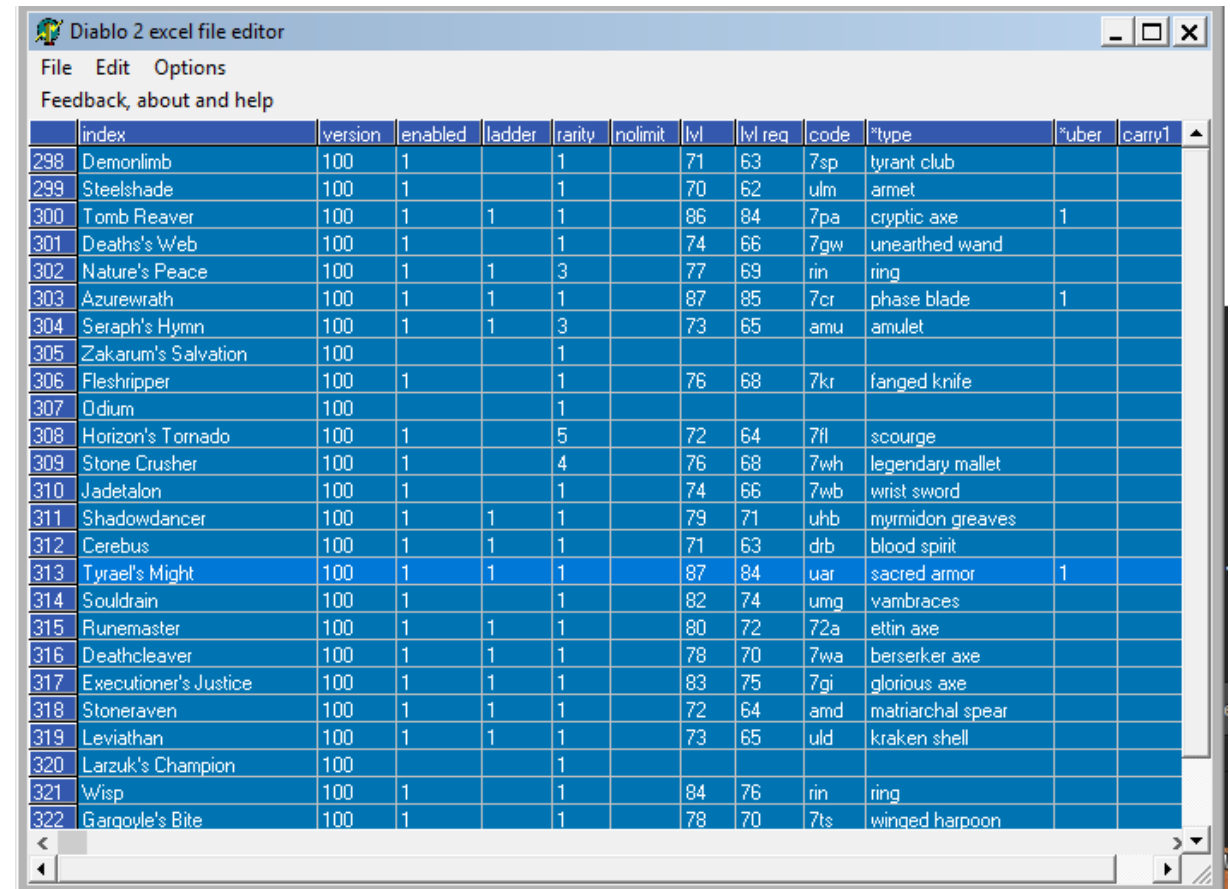
➔ Blizzard hat es selbst versemmt und die .txt Dateien mit ausgeliefert



3. Items – Beispiel Uniques

- Alle Eigenschaften der Unique-Items sind in der Tabelle „Uniqueltems.txt“
- Neuer Eintrag → Neues Unique
- Erklärung der Datei:

<https://d2mods.info/forum/viewtopic.php?f=4&t=34455>



The screenshot shows a window titled "Diablo 2 excel file editor" with a menu bar (File, Edit, Options) and a status bar. The main area contains a table with 13 columns: index, name, version, enabled, ladder, rarity, nolimit, lvl, lvl req, code, *type, *uber, and carry1. The table lists various unique items from index 298 to 322.

	index	name	version	enabled	ladder	rarity	nolimit	lvl	lvl req	code	*type	*uber	carry1
298	Demonlimb	100	1		1			71	63	7sp	tyrant club		
299	Steelshade	100	1		1			70	62	ulm	armet		
300	Tomb Reaver	100	1	1	1			86	84	7pa	cryptic axe	1	
301	Deaths's Web	100	1		1			74	66	7gw	uneearthed wand		
302	Nature's Peace	100	1	1	3			77	69	rin	ring		
303	Azurewrath	100	1	1	1			87	85	7cr	phase blade	1	
304	Seraph's Hymn	100	1	1	3			73	65	amu	amulet		
305	Zakarum's Salvation	100			1								
306	Fleshripper	100	1		1			76	68	7kr	fanged knife		
307	Odium	100			1								
308	Horizon's Tornado	100	1		5			72	64	7fl	scourge		
309	Stone Crusher	100	1		4			76	68	7wh	legendary mallet		
310	Jadetalon	100	1		1			74	66	7wb	wrist sword		
311	Shadowdancer	100	1	1	1			79	71	uhb	myrmidon greaves		
312	Cerebus	100	1	1	1			71	63	dib	blood spirit		
313	Tyrael's Might	100	1	1	1			87	84	uar	sacred armor	1	
314	Soulrain	100	1		1			82	74	umg	vambraces		
315	Runemaster	100	1	1	1			80	72	72a	ettin axe		
316	Deathcleaver	100	1	1	1			78	70	7wa	berserker axe		
317	Executioner's Justice	100	1	1	1			83	75	7gi	glorious axe		
318	Stoneraven	100	1	1	1			72	64	amd	matriarchal spear		
319	Leviathan	100	1	1	1			73	65	uld	kraken shell		
320	Larzuk's Champion	100			1								
321	Wisp	100	1					84	76	rin	ring		
322	Gargoyle's Bite	100	1		1			78	70	7ts	winged harpoon		

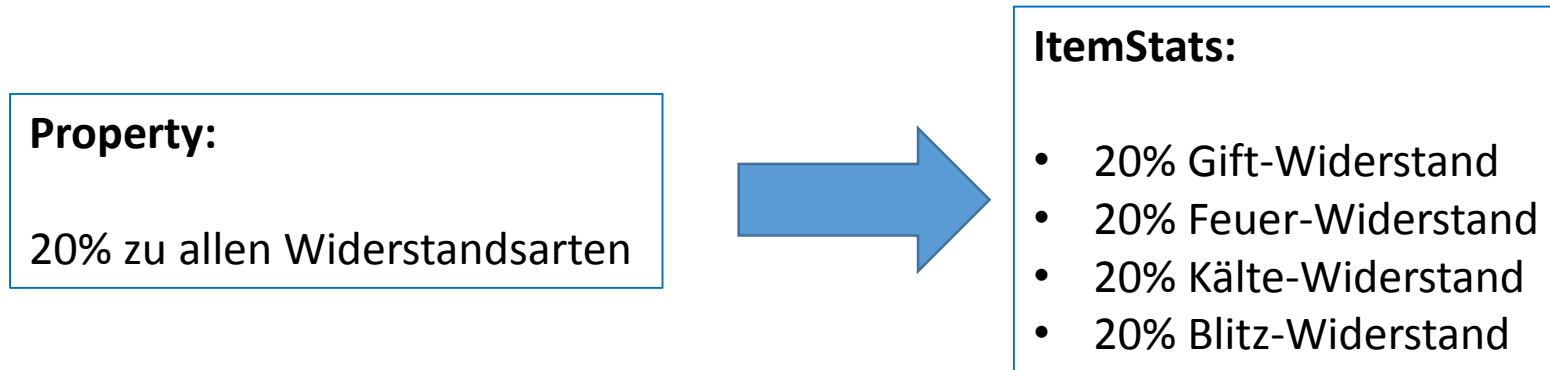
3. Items – Beispiel Uniques

- Ein Unique hat jede Menge Eigenschaften, z.B:
 - Level-Requirement
 - Stärke/Geschicklichkeit Anforderungen
 - DC6 Image
 - ItemType
 - **Properties**

prop1	par1	min1	max1	prop2	par2	min2	max2	prop3	par3	min3	max3	prop4	par4	min4	max4	prop5
ac%		70	100	move2		30	30	balance2		30	30	dex		15	25	skilltab
ac%		130	140	skilltab	16	2	4	lifesteal		7	10	att%		60	120	openw
ease		-100	-100	indestruct		1	1	ac%		120	150	rip		1	1	dmg-d
ac%		90	120	manasteal		4	7	lifesteal		4	7	hit-skill	Weaken	8	3	dmg-a

3. Items – Properties

- In Diablo 2 sind alle Eigenschaften in 2 Dateien ausgelagert
 1. Properties.txt
 2. ItemstatCost.txt
- Eine „Property“ ist dabei eine Abbildung auf 1 oder mehrere „Itemstats“.

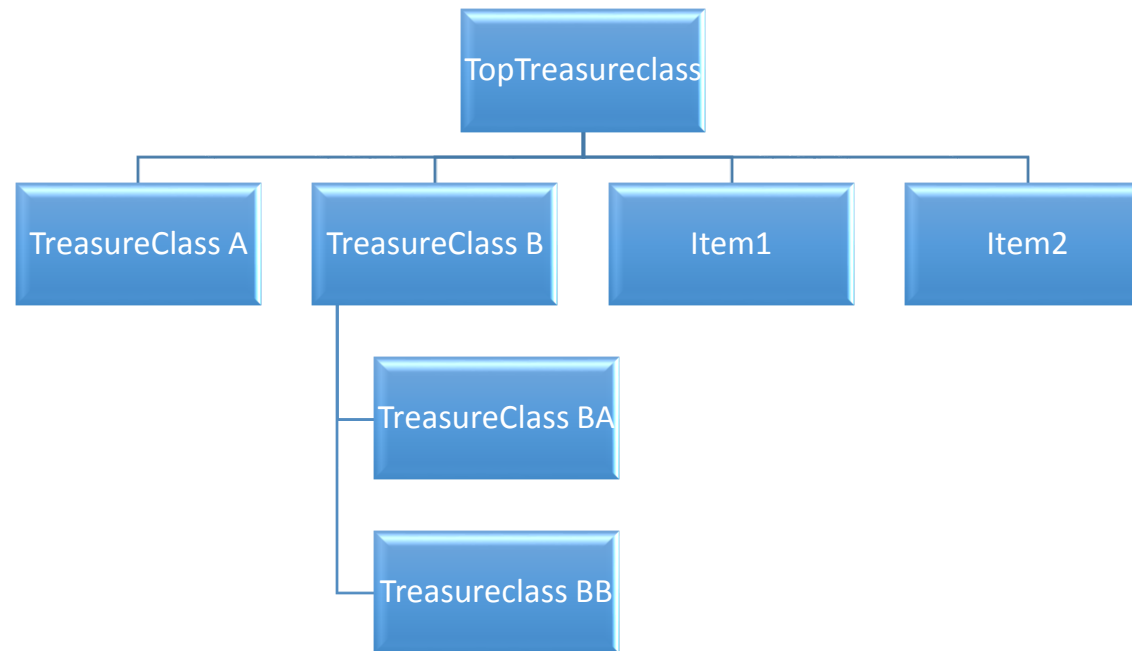


3. Items und TreasureClasses

- Um ein Item korrekt in das Spiel zu integrieren, muss man dafür sorgen, dass das Item auch im Spiel „droppen“ kann und auch die richtige „Rarity“ hat
- ➔ Jedes Objekt (Monster, Schatzkisten, etc..) kann 1 oder mehrere „Treasureclasses“ zugeordnet bekommen, die bestimmen, was ein Monster droppen kann.

3. TreasureClasses

- Treasureclasses können als Baum-Visualisiert werden



Treasure Class	group	level	Picks	Unique	Set	Rare	Magic	NoDrop	Item1	Prob1	Item2	Prob2	Item3	Prob3	Item4	Prob4	Item5
Act 2 (N) Chest B	6	46	4					100	gld	15	Act 2 (N) Junk	15	Act 2 (N) Equip B	10	Act 2 (N) Good	2	
Act 2 (N) Chest C	6	48	4					100	gld	15	Act 2 (N) Junk	15	Act 2 (N) Equip B	10	Act 2 (N) Good	2	
Act 3 (N) Chest A	6	50	4					100	gld	15	Act 3 (N) Junk	15	Act 3 (N) Equip A	10	Act 3 (N) Good	2	
Act 3 (N) Chest B	6	52	4					100	gld	15	Act 3 (N) Junk	15	Act 3 (N) Equip B	10	Act 3 (N) Good	2	
Act 3 (N) Chest C	6	54	4					100	gld	15	Act 3 (N) Junk	15	Act 3 (N) Equip B	10	Act 3 (N) Good	2	
Act 4 (N) Chest A	6	56	4					100	gld	15	Act 4 (N) Junk	15	Act 4 (N) Equip A	10	Act 4 (N) Good	2	

3. Skills und Missiles

- Skills werden in D2 ebenfalls über Tabellen editiert, maßgeblich dafür sind 2 Dateien:
 - Skills.txt
 - Missiles.txt
- ➔ Beide Dateien haben über 200 Spalten für die Datenstruktur!!

3. Skills und Missiles

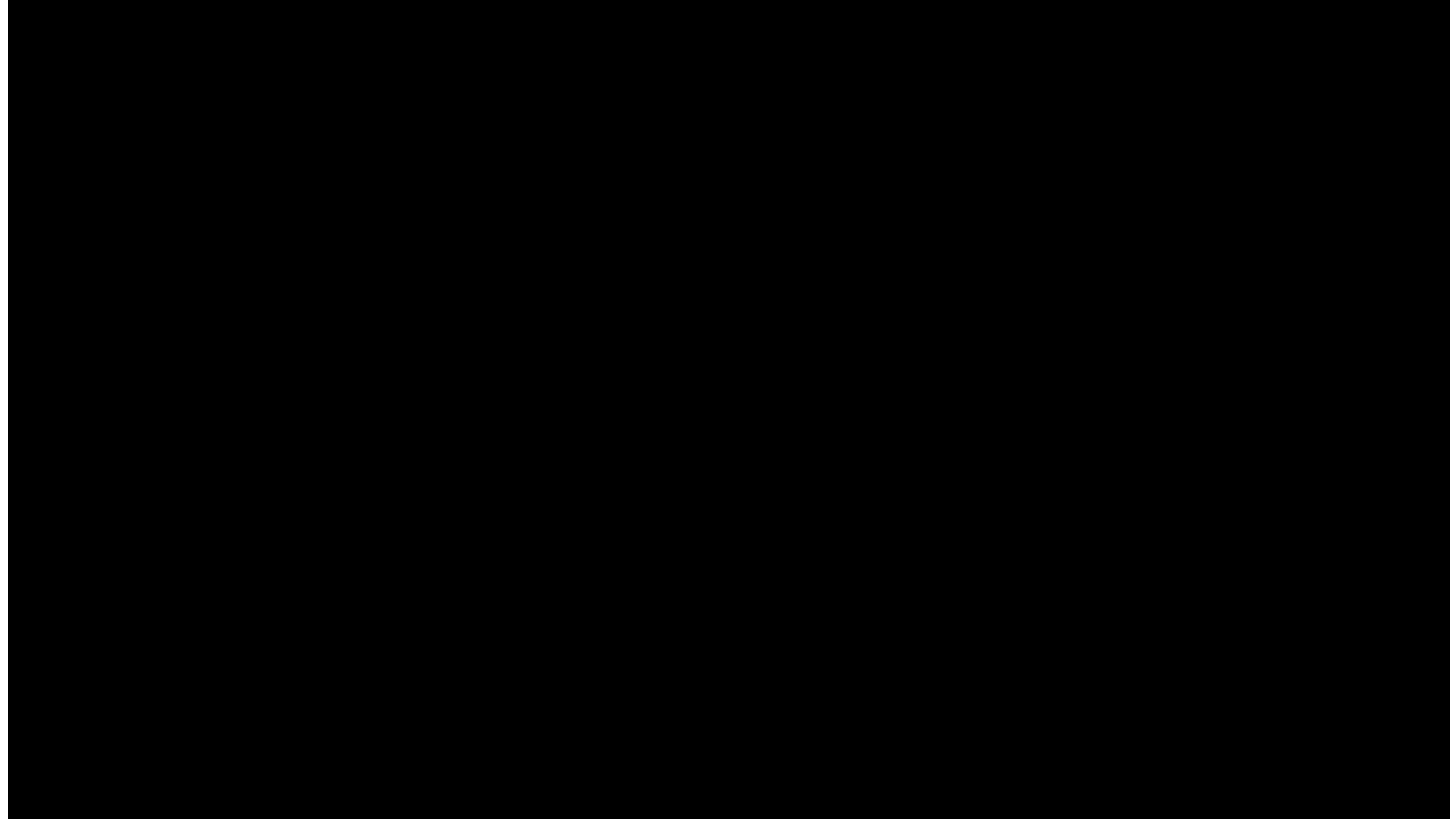
- Skills können prinzipiell unterteilt werden in:
 1. **Melee**
 2. **Ranged** (Egal, ob Spell, Bogen, etc...)
 3. Summon
 4. Aura
 5. Passive
 6. Diverse Hardcoded Skills

3. Skills und Missiles

- Grundidee von Melee/Ranged Skills
 1. Per Klick/Cast wird eine Callback-Funktion Ausgeführt, sog. StartFunction (und „Do-Function“)
 2. Diese Funktionen können als Argumente „Missiles“ erhalten und bestimmt das Verhalten dieser (srvmis, srvmissA-srvMissC)

adam	srvmissile	d
	frozenorb	

3. Skills und Missiles (Charged Bolt)



	skill	id	charclass	skilldesc	svstfunc	svdolunc	
36	Lightning Fury	35	ama	lightning fury	4		
37	Fire Bolt	36	sor	fire bolt			
38	Warmth	37	sor	warmth			
39	Charged Bolt	38	sor	charged bolt		17	
40	Fire Bolt	39		fire bolt			

srvmissilea	srvmissileb	srvmissilec	s
chargedbolt	chargedbolt	chargedbolt	

3. Skills und Missiles

- Grundidee von Melee/Ranged Skills
 1. Per Klick/Cast wird eine Callback-Funktion Ausgeführt, sog. StartFunction (und „Do-Function“)
 2. Diese Funktionen können als Argumente „Missiles“ erhalten und bestimmt das Verhalten dieser (srvmis, srvmissA-srvMissC)
 3. Diese Missiles sind wiederum Objekte (mit Bewegungsverhalten) mit weiteren CallBack-Funktionen (onHit, onExplode, onTravel) in denen weitere Missiles mit anderem Bewegungsverhalten erzeugt werden
 4. Am Ende dieser „Missile-Chain“ werden Missiles mit Schaden/States eingesetzt

adam	srvmissile	id
	frozenorb	

3. Skills und Missiles

- Beispiel Frozen Orb
- Beispiel Firebolt nova

3. Monster

- Monster bestehen aus Grafikfiles (.dc6, .dcc), „gluefiles“ (.cof) und Logik in Form von Tabellen
- Monster haben klassische Eigenschaften wie:
 - Leben
 - Attack Rating (wie wahrscheinlich trifft es dich)
 - Armor Class (seine Defense)
 - Level
 - Geschwindigkeit
 - Skills
 - Properties

3. Monster – Diablo II und AI

- Die AI in Diablo II ist sehr stark Distanzgesteuert, typisch:
 - Falls Spieler in Range x kommt, switche zu Verhalten 1
 - Falls Spieler in Range y kommt, switche zu Verhalten 2
 - Falls Spieler in Range z kommt, switche zu Verhalten 3
 - Sont Idle
- Typisches Verhalten:
 - Per Zufall Nutze Skill 1, Skill 2 oder Skill 3
- Manche Ais nutzen noch die Aktuellen Leben des Monsters aus

3. Monster – Diablo II und AI

- Aus Diablo II AI Compendium

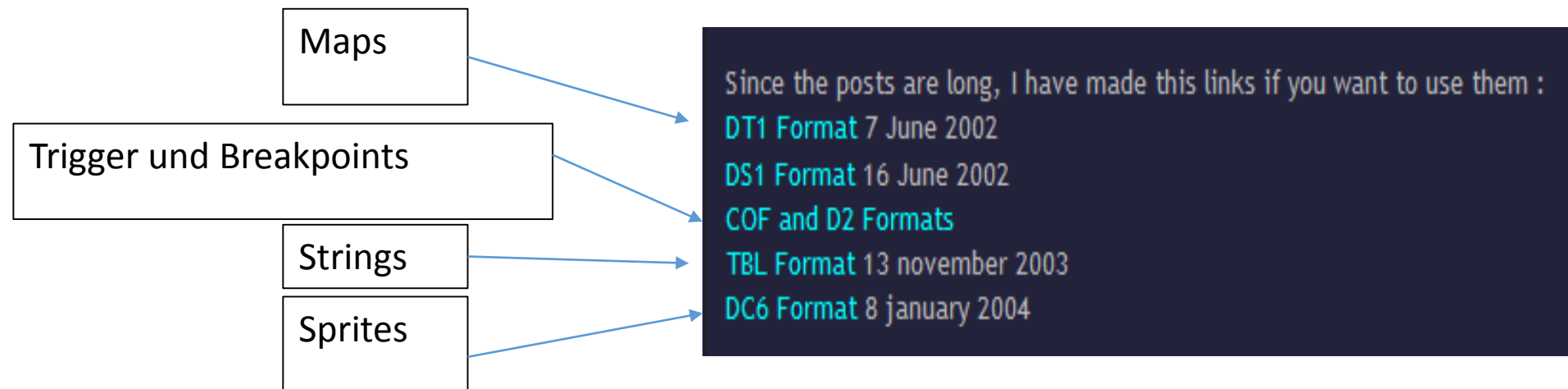
Summoner

By default the **summoner** will wander around randomly. When the foe comes within **AIDist** the **summoner** will start rolling **par1**, if it succeeds he will either cast **SK4** (with the interval specified by **par5**) or **SK5** (if **par2** succeeds, which btw overrides all other AI parameters!), once the foe comes within **par8**-range the **summoner** starts rolling **par3**, which will either make him use **SK1** or **SK3** (never both of them, which he uses is apparently specified by the game seed or a similar mechanism). If the foe comes closer, into **par7**-range, he will cast **SK2** (in the interval specified by **par4**). The **summoner** does not use **par6** despite Blizzard documentation.

1. Chance to use a **skill**
2. Chance to use **SK5** (**par1** succeeds)
3. Chance to use **SK1** or **SK3** (**par1** succeeds, **par2** fails and within **par8**-range)
4. Interval (frames) (between uses of **SK2**)
5. Interval (frames) (between uses of **SK4**)
6. n/a
7. Range (subtiles)
8. Range (subtiles)

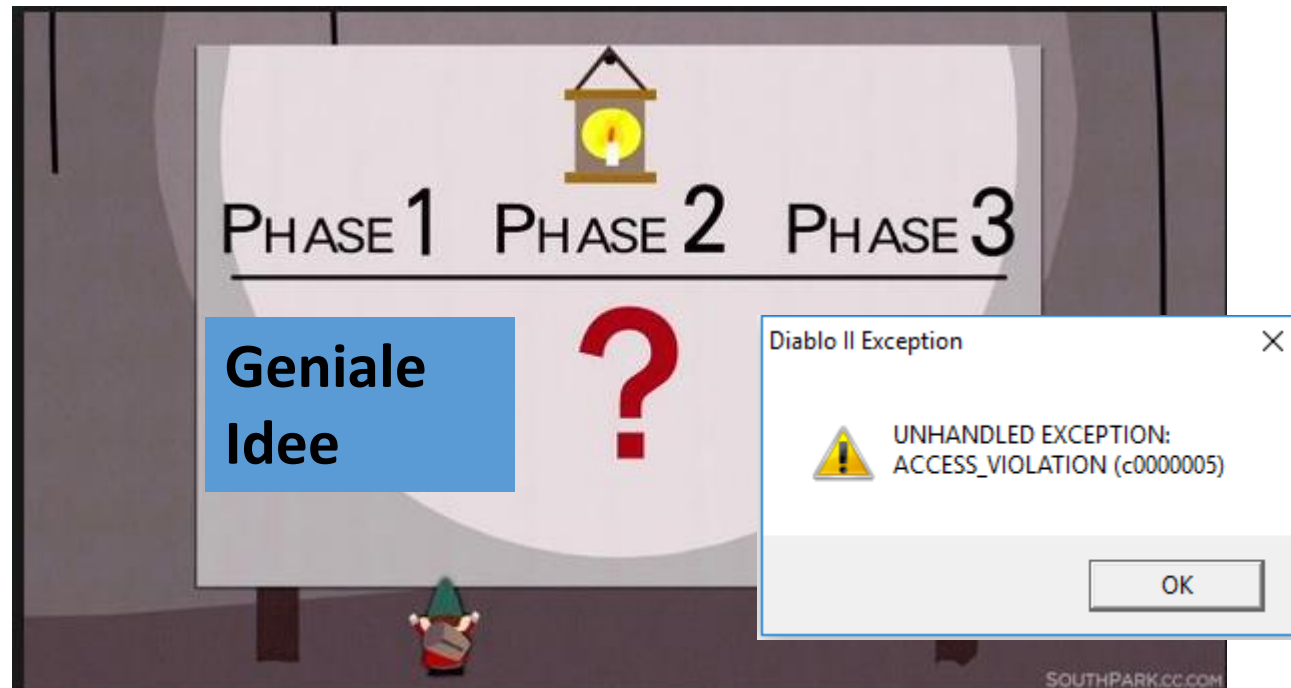
3. Modding Diablo II - Dateiformate

- Die Dateiformate wurden alle durch Reverse Engineering herausgefunden, und können hier begutachtet werden:
- <https://d2mods.info/forum/viewtopic.php?t=724>

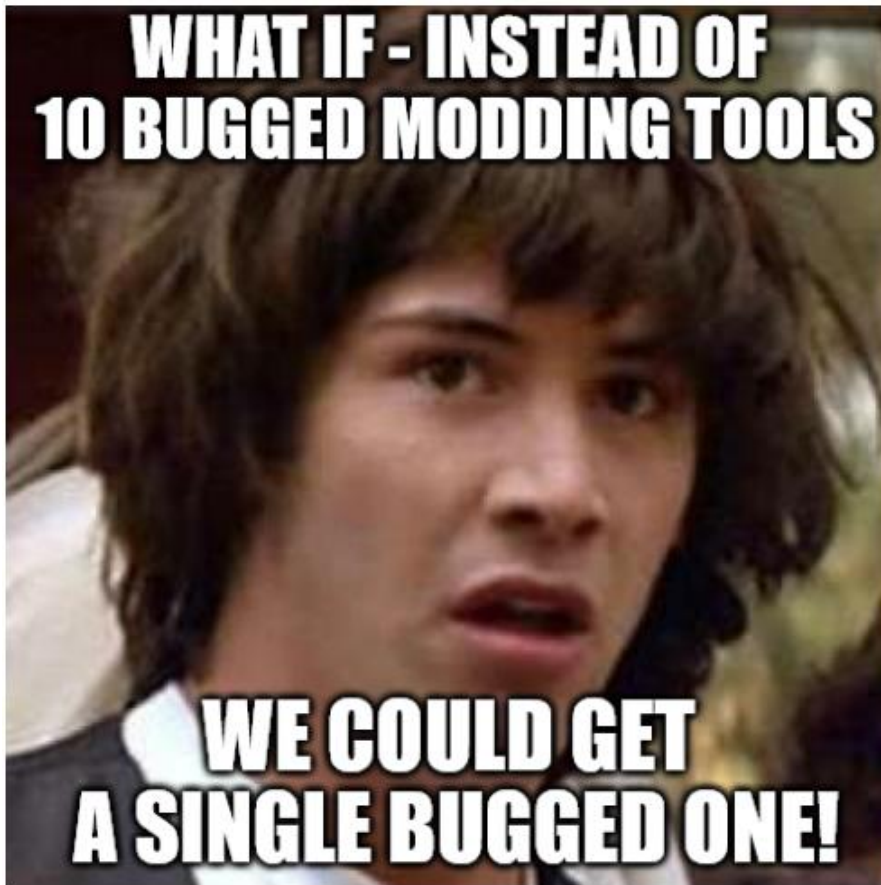


Nicht in der Agendapunkt!

- Modding Diablo 2 ist aktuell wie folgt:
 - Nutze ca. 10 verschiedene Tools um letztendlich potentiell inkonsistente Dinge zu erzeugen, daher passiert meistens



Nicht in der Agendapunkt!



- Falls 1-2 Leute Interesse haben „one tool to rule them all“ zu entwerfen.

➔ Ich bin sicher dafür kann man auch ECTS vergeben (Masterpraktikum, Projekt etc..)

Fragen.txt

Nicht einmal der Tod kann dich vor ihnen retten!

(Diablo Akt4)