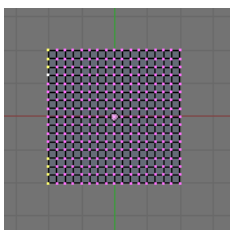


14. lekcia

Fyzika



alebo „Voda, šaty a hry“

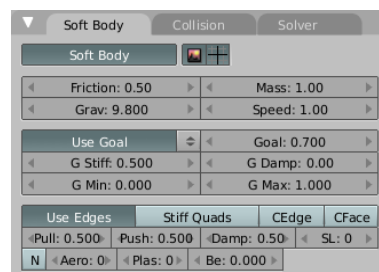
V názve tejto kapitoly som si dovoľil nahradiť ono starorímske „chlieb a hry“ inými komoditami. To, o čo v nej pôjde, je spôsob, akým Blender dokáže modelovať rôzne fyzikálne deje. Tí, ktorým pri slove „fyzika“ naskakujú červené vyrážky, nech zatiaľ neodchádzajú a ešte chvíľu vydržia. Budeme sa totiž zaoberať tým, ako vie Blender simulovať látky, ktoré sú mäkké a pružné, čo vie spraviť s tkaninami, ako sa dá urobiť voda či iná tekutina a ako sa dá dosiahnuť, aby telesá padali, strkali jedno do druhého a inak na seba reagovali. Nakoniec vyskúšame, ako sa dá priamo v Blenderi vytvoriť jednoduchá hra. Patričnú fyziku väčšinou našťudovali tí, ktorí to programovali a na vás je už iba použiť hotové nástroje. Koniec koncov v lekcii o časticiach bolo v pozadí tej fyziky tiež dosť a prežili ste to.



Obrázok 1: Plane

Začnime s mäkkými a pružnými látkami, ktoré sa v Blenderi zvyknú označovať pojmom *Soft Body* (plastické teleso). Ako príklad niečoho takého skúsime na začiatok vymodelovať handru zavesenú na šnúre. Najprv spravíme predbežnú modelársku prípravu: Pridajte si na scénu objekt typu *Plane* (na obrázku č. 1 ho vidíte pri pohľade zhora) a štyrikrát zahustíte vrcholy (v režime úpravy klávesa **W** a zvolíte *Subdivide*). Vytvorte novú skupinu vrcholov (na paneli *Link and Materials* v tlačidlách úpravy), nazvite ju *Stipce* a pridajte do nej tie vrcholy, ktoré sú na obrázku č. 1 aktívne. (Bolo by dobre, aby tam tie vrcholy patrili s hodnotou 1. Pre istotu sa prepnete do režimu *Weight Paint* a skontrolujte, či sú červené a či je všetko ostatné modré.)

Prípravu máme za sebou, môžeme sa pustiť do samotného nastavovania. Prepnete sa v okne tlačidiel na tlačidlá objektu (ikona ) a spomedzi nich vyberte tlačidlá fyziky (ikona ). Pamätníci vedia, že na tom istom mieste sme v lekcii o časticiach nastavovali vlastnosti jednotlivých silových polí. Tentokrát sa ale budeme venovať panelu *Soft Body*. Stlačte tlačidlo *Soft Body* a na paneli sa objaví množstvo vstupných polí, ktoré môžete vidieť na obrázku č. 2.



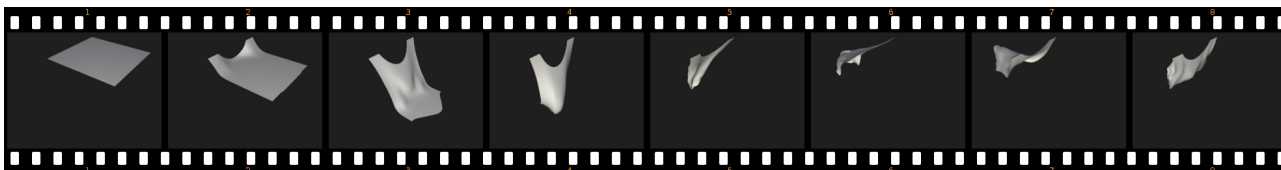
Obrázok 2: Soft Body

Prvé štyri vstupné polia sa týkajú nastavenia fyzikálnych vlastností telesa ako celku. Prvé z nich je *Friction* (trenie). Ak modelujete napríklad suchú hodvábnu šatku, túto hodnotu výrazne zväčšite. Ak modelujete mokrú handru, trenie s okolím bude samozrejme nízke. *Mass* je hmotnosť celého objektu (medzi vrcholy sa rovnomerne rozdelí). Čo sa týka gravitácie, je jedno, ako túto hodnotu nastavíte. Ale čím je teleso ťažšie, tým menej naň pôsobia silové polia. *Grav* určuje veľkosť gravitačného zrýchlenia. Štandardne je nastavená pozemská hodnota. Ak chcete modelovať sušenie prádla na Mesiaci, dajte tam 1,622. (Nezabudnite v tom prípade nastaviť *Friction* na nulu, lebo je tam vzduchoprázdno.) *Speed* určuje, po akých krokoch sa majú prepočítavať nové hodnoty. Ak tam dáte 10, dostanete zrýchlený film, ak 0,1, tak spomalený.

S plastickými telesami sa dá pracovať dvoma rôznymi spôsobmi. Buď povieme, že ich vrcholy (alebo aspoň niektoré z nich) sa majú (aspoň čiastočne) držať svojho miesta, alebo celé teleso necháme napospas fyzikálnym silám. V prípade, že hodláte použiť prvú možnosť, nechajte zapnuté tlačidlo *Use Goal* (použiť cieľ). V našom prípade chceme, aby handra držala na štipcoch a tak to zapnuté nechajte.

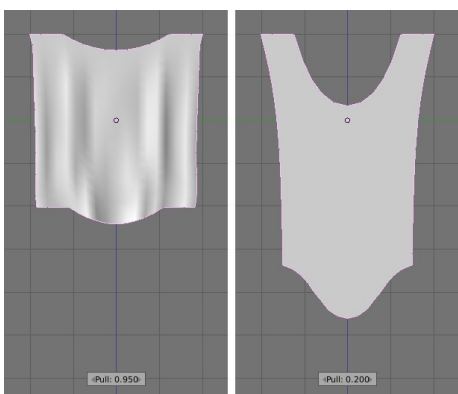
Vstupné pole *Goal* označuje, ako veľmi sa majú vrcholy držať jednotlivých pozícií. Hodnota 1 znamená, že na ne fyzika nepôsobí. Hodnota 0 znamená, že vrcholy úplne ignorujú, kde by mali

byť (napríklad podľa animačných IPO kriviek) a že podliehajú iba fyzike. V našom prípade by sme boli radi, aby vrcholy zo skupiny *Stipce* zostali na mieste a aby sa ostatné vrcholy správali podľa fyziky. Nastavte teda hodnotu *Goal* na 0 a dvojšípkou vedľa tlačidla *Use Goal* vyberte skupinu *Stipce*. Pre vrcholy z tejto skupiny sa použije váha, ktorú v skupine majú, pre ostatné vrcholy sa použije tá nula, ktorú ste nastavili.



Obrázok 3: Handra

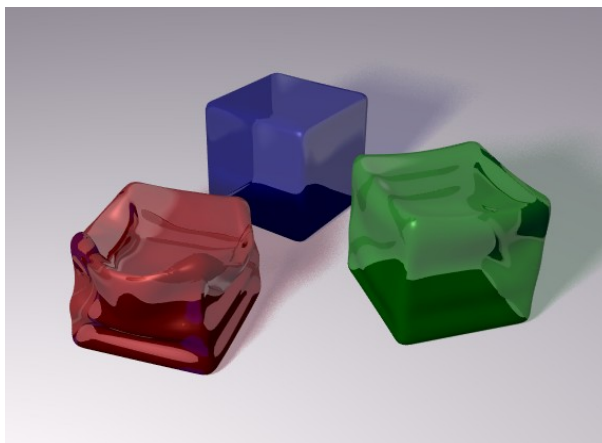
Keď to teraz zanimujete, alebo si pustíte prehrávanie animácie v 3D okne (**ALT-A**), prištipcované vrcholy ostanú na mieste, ale zvyšok handry sa začne správať ako kyvadlo a kmitať z jednej strany na druhú. Počiatočnú fázu v polsekundových záberoch môžete vidieť na obrázku č. 3.



Obrázok 4: Pull

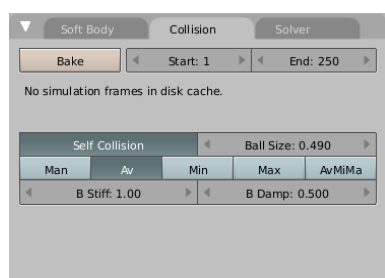
Na animácii si môžete všimnúť, že vrcholy, ktoré sme dali do skupiny *Stipce* a ktoré mali nastavené *Goal* na 1 sa ani nepohli. (Toto nám bude zatiaľ ku šťastiu stačiť, o vrcholoch s nastaveným *Goal* si povieme niečo viac o chvíľu.) Ostatné vrcholy sa ale hýbali celkom aktívne. Menila sa nielen ich poloha, ale aj ich vzájomná vzdialenosť. Je to dobre vidieť na oblúku, ktorý vytvorila gravitácia medzi štipcami tak, že handru natiahla. Hrany plastických telies sa totiž správajú ako pérovanie na aute. Ak vrcholy niečo (napríklad gravitácia) ťahá od seba, hrany sa môžu do istej miery natiahnuť. Ako veľmi sa môžu natiahnuť, určuje parameter *Pull* – ťah (na obrázku č. 2 v dolnej sekcii v strednom riadku vľavo). Môže nadobúdať

hodnotu od nula do jedna (interval je vpravo otvorený). Čím je táto hodnota menšia, tým sú pružiny pružnejšie – dajú sa viac natiahnuť. A čím je väčšia, tým je teleso menej elastické. Na obrázku č. 4 môžete vidieť vplyv tohto parametra na našu handru. Vľavo má *Pull* hodnotu 0,95, vpravo hodnotu 0,2. Hodnota *Push* (tlak) naopak určuje, ako sa dajú pružiny stlačiť. Na obrázku č. 5 vidíte tri kocky (na ktoré bolo trikrát použité *Subsurf*), ktoré sa okrem farby od seba líšia iba hodnotou *Push*. Červená kocka ju má nastavenú na 0,7, zelená na 0,9 a modrá na 0,999.



Obrázok 5: Push

Aby kocky úplne neskolabovali, bolo ale okrem nastavenia *Push* treba urobiť niekoľko ďalších vecí. Jednak bolo treba *Pull* nastaviť na 0,999, pretože ak by sa hrany mohli nafahovať, kocky by sa stále mohli poskladať na zem. Jednak

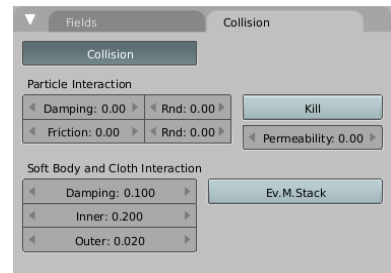


Obrázok 6: Collision

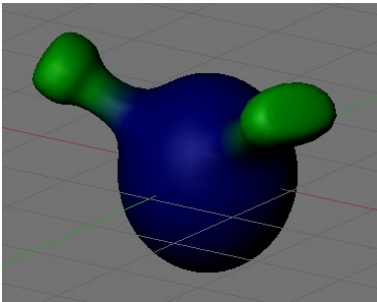
bolo treba zapnúť tlačidlo *Stiff Quads* (pevné štvoruholníky), ktoré spôsobí, že sa stenám štvoruholníkového tvaru pridajú okrem pružín na hranách ešte dve pružiny na uhlopriečky a nakoniec bolo treba na paneli *Collision* (Pozor, sú tam až tri panely s týmto názvom. Ide o ten, ktorý sa týka *Soft Body* a ktorý môžete vidieť na obrázku č. 6.) zapnúť tlačidlo *Self Collision*, ktoré spôsobí, že teleso nebude pretínať samo seba. Ďalej treba dlážke zapnúť na paneli *Collision* (tentokrát na tom, ktorý je v okne štandardne úplne vľavo a ktorý môžete vidieť na obrázku č. 7) tlačidlo

Collision, aby cez ňu kocky neprepadli. Keďže kocky iba ležia na dlážke a na ničom nedržia, majú vypnuté aj tlačidlo Use Goal. A aby to vyzeralo lepšie, tak som kockám zapol Subsurf modifikátor.

Každá fiktívna pružina medzi vrcholmi má k sebe pripojený aj tlmič podobne ako perovanie na aute. Silu tohto tlmiča určuje hodnota Damp (tlmenie). Ak ju nastavíte na niečo veľké (maximum je 50), pružiny budú kmitať menej a tlmenejšie. Hodnota SL určuje, na akú dĺžku sa majú pružiny skrútiť oproti originálu. Ak je hodnota 0, efekt je vypnutý. Ak chcete, aby sa vám handra po praní zrazila, vyskúšajte tam dať 50.



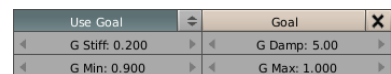
Obrázok 7: Collision č.2



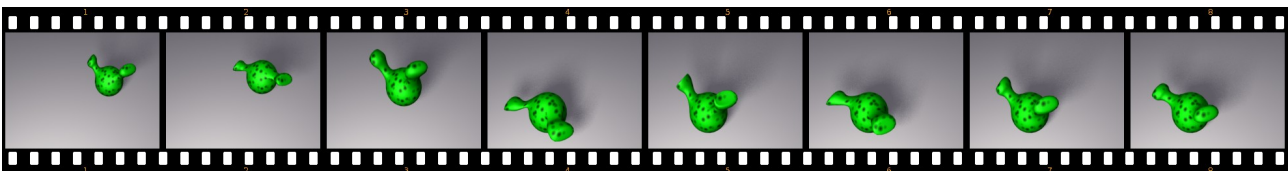
Obrázok 8: Tykadlá

Vráťme sa teraz na chvíľu k hodnotám, ktoré sa týkajú cieľovej pozície vrcholov. Predstavte si, že chceme modelovať mimozemšťana, ktorý bude mať pružné tykadlička. Celý zvyšok mimozemšťana sa má správať ako bežná animovaná postava, iba tykadlá majú kmitať, keď mimozemšťan pohne hlavou. Vytvoríme skupinu vrcholov, v ktorej budú iba tykadlá a aj to nie naplno, ale iba s polovičnou váhou. (skupinu vrcholov môžete vidieť na obrázku č. 8). Celému mimozemšťanovi nastavte hodnotu Goal na 1 (aby sa mu pri animácii vrcholy netúlali, kde nemajú) a potom zvolte, že na určenie váhy sa má použiť tá skupina vrcholov, ktorú ste vytvorili, pretože tým istú voľnosť doprajeme. Teraz vytvorte s mimozemšťanom nejakú jednoduchú animáciu – napríklad takú, že urobí dva skoky vpred a potom zastanie. Zistíte, že mimozemšťan sa správa tak, ako ste ho animovali, ale jeho tykadlá žijú svojim vlastným životom. Ale ako sa budú správať, môžeme ovplyvniť tými hodnotami, ktoré vidíte na obrázku č. 9.

Hodnota G Stiff určuje silu pružiny, ktorá vrchol ťahá na to miesto, na ktorom by mal podľa animácie byť. Ak ju dáte malú, vrchol bude kadekde traďať, ak veľkú, bude kmitať okolo cieľovej pozície tesnejšie. Hodnota G Damp určuje silu tlmiča. Pri veľkej hodnote dokmitá vrchol do cieľovej pozície rýchlejšie. Hodnoty G Min a G Max určujú, hranice samotnej hodnoty Goal – teda to, ako ďaleko sa vrchol od svojho miesta vlastne môže dostať. Ovplyvňujú aj štandardné nastavenie hodnoty Goal, aj váhy nastavené skupinou vrcholov, takže keď to pri nastavovaní skupiny spravíte tak, že sa to nespráva celkom ako by ste chceli, tu je to miesto, kde to môžete opraviť. Ufóna s pružnými tykadlami môžete vidieť na obrázku č. 10. Všimnite si, že na posledných piatich snímkoch je ufón stále na tom istom mieste. Jeho animácia už skončila. Len tykadlá ešte zotrvačnosťou bimbajú.



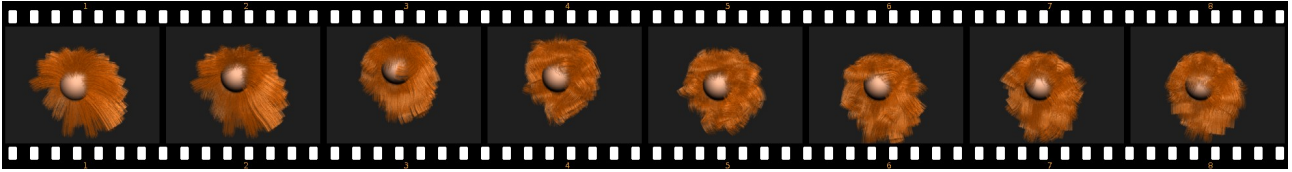
Obrázok 9: SoftBody – detail



Obrázok 10: Mimoszemšťan s tykadlami

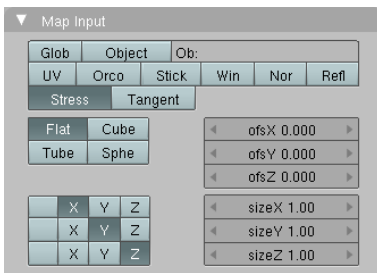
Plastické teleso môžu byť aj vlasy. Všetko sa nastavuje rovnako, ako u bežných plastických telies, len s tým rozdielom, že ak je k objektu pripojený nejaký časticový systém, na paneli Soft Body sa objaví roletové menu, v ktorom zvolíte, či sa nastavenie má týkať samotného telesa, alebo časticového systému – teda vlasov. Vlasy treba v tomto menu vybrať ešte predtým, než stlačíte tlačidlo Soft Body, inak by namiesto vlasov kmitala hlava, z ktorej vlasy rastú. Ako sa správajú vlasy, ktorým ste nastavili Soft Body, môžete vidieť na obrázku č. 11. Na prvých piatich zachytených snímkoch sme hlavou pohli hore a dole. Ďalší pohyb vlasov má na svedomí už len fyzika.

Keď nastavíte vlasy ako `Soft Body`, je dobré zapnúť kontrolu na `Collision` a `Self Collision`, aby vám vlasy nelietali krížom cez hlavu a divne sa nemotali. Ak je ale vlasov priveľa, detekcia kolízií môže trvať veľmi dlho, pretože výpočty sú príliš náročné. Situáciu môžete vyriešiť tak, že zmenšíte počet vlasov a každému zväčšíte počet potomkov.

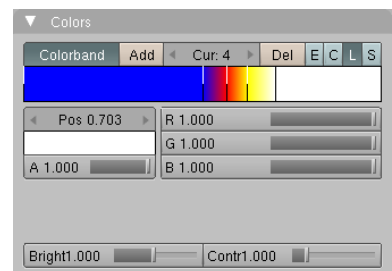


Obrázok 11: Vlasy

V súvislosti s plastickými telesami treba ešte pripomenúť jednu vec, ktorá bola okrajovo spomenutá v kapitole o textúrach. Jeden zo spôsobov, ktorým sa dala textúra priradiť k objektu bola voľba súradníc `Stress` – záťaž. (Pozrite obrázok č. 12.) Tieto súradnice určujú, ako veľmi sa zmenila dĺžka hrany pružného telesa oproti jej bežnej dĺžke vplyvom fyziky. Tento parameter nadobúda hodnotu od -1 do 1 , pričom -1 znamená, že hrana bola stlačená na nulovú veľkosť, 0 znamená, že hrana má svoju štandardnú dĺžku a 1 znamená, že dĺžka hrany sa nekonečne zväčšila.



Obrázok 12: Map Input



Obrázok 13: Colors

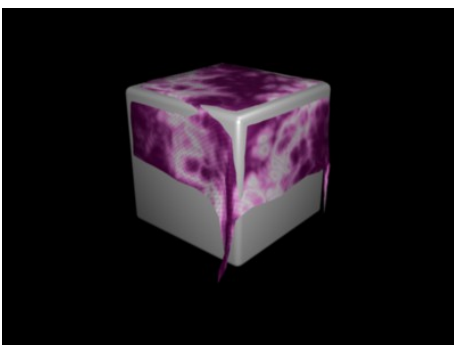
Keď takýmto spôsobom nanesieme na teleso textúru typu `Blend`, ktorá bude mať farebný prechod rovnaký, ako môžete vidieť na obrázku č. 13, tak uvoľnené časti telesa budú mať modrú farbu a čím viac budú niektoré jeho časti namáhané, tým bude farba „žeravejšia“. Všimnite si, že vo farebnom prechode sa všetko podstatné udeje v jeho pravej polovici, pretože momentálne hodláme vykresľovať iba naťahovanie materiálu a stláčanie nás nezaujímá. Pád gumenej plachty zachytenej za rohy, na ktorú bola použitá táto textúra môžete vidieť na obrázku č. 14.



Obrázok 14: Záťaž

S týmto spôsobom nanášania textúry sa bežne pracuje tak, že si vytvoríte materiál, ktorý predstavuje masku namáhania objektu. Bude to fungovať rovnako, ako v našej ukážke, iba farebný prechod bude jednoduchší – z čiernej do bielej. Tento materiál potom v editore uzlov použijete, ako uznáte za vhodné pri vytváraní finálneho materiálu objektu. (Napríklad tak, že gumová plachta bude tam, kde je najviac natahnutá priesvitnejšia.)

Šaty a tkaniny



Obrázok 15: Obrus cez `Soft Body`

Predstavte si teraz, že by ste chceli s pomocou plastických telies spraviť obrus, ktorý dopadá na stôl. Spraviť sa to samozrejme dá, ale bude to vyzeráť podobne, ako môžete vidieť na obrázku č. 15 (obrus je `Plane`, na ktorý sme niekoľkokrát použili `Subdivide`, pre jednoduchosť sme ako stôl použili kocku so zaoblenými hranami). Obrus nám prepadne dovnútra kocky a napriek tomu, že sme nastavili `Pull` na maximum, rohy obrusu prevísajú.

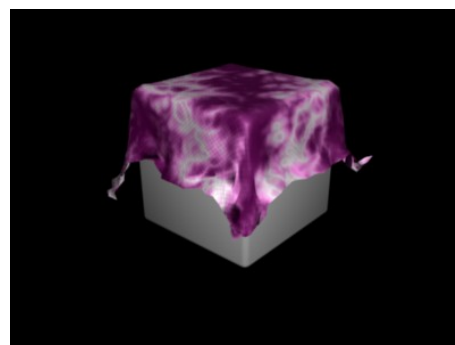
Tomu, aby obrus nepadal dovnútra kocky sa dá do istej miery zabrániť pomerne jednoducho. Stačí na paneli `Soft Body` stlačiť tlačidlá `CEdge` a `CFace`. Blender nebude kontrolovať iba kolíziu vrcholov, ale aj hrán a stien. Môžete prípadne ešte zahustiť vrcholy s pomocou `Subdivide`. To značne celú vec vylepší, aj keď celý výpočet sa samozrejme spomalí. Výsledok môžete vidieť na obrázku č. 16.

Najlepšia a najjednoduchšia možnosť je ale v danej situácii sa na pružné telesá úplne vykašľať a použiť `Cloth` – textílie. Blender obsahuje celý podsystem na fyziku textílií. Takže v paneli `Soft Body` vypnete tlačidlo `Soft Body`,



Obrázok 17: Cloth

presuňte sa o panel vedľa na panel `Cloth` a tam stlačte tlačidlo `Cloth`. Panel by mal vyzeráť podobne ako na obrázku č. 17. (Až na ten detail, že som trochu zväčšil parameter `Quality`, ktorý určuje, ako presne sa má výpočet vykonať.) Ostatné parametre nechajte tak, ako sú štandardne nastavené. Výsledok by mal byť podobný, ako môžete

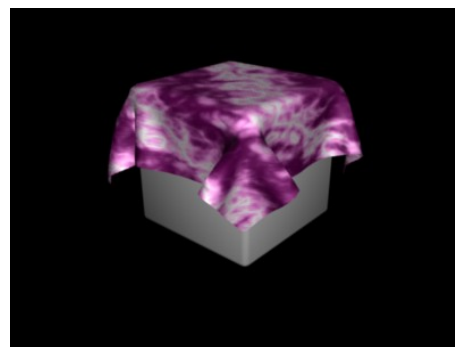


Obrázok 16: Vylepšený obrus

vidieť na obrázku č. 18.

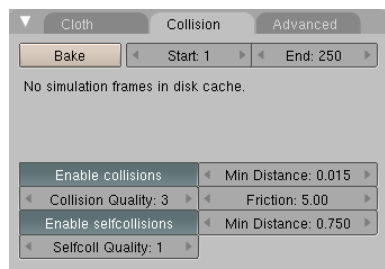
Roletové menu na paneli obsahuje niekoľko štandardných nastavení, ale nič vám samozrejme nebráni nastaviť si všetko po svojom. Možnosti sú `Cotton` – bavna, `Silk` – hodváb, `Rubber` – guma, `Denim` – riflovina (alebo vo všeobecnosti hrubé plátno), `Leather` – koža a nakoniec `Custom` – teda nejaké vaše špeciálne nastavenie.

Keď zvolíte niektorý z preddefinovaných materiálov, zmení to rovno aj ostatné nastavenia. `StructStiff` určuje tuhosť štruktúry materiálu – čím je vyššie, tým je materiál pevnejší. `BendStiff` je pevnosť materiálu v ohybe. Čím je vyššie, tým väčšie záhyby látka vytvára. `Spring Damp` je vnútorné tmenie, čím je hodnota vyššia, tým sa látka menej krčí. `Air Damp` je trenie o vzduch. Čím je vyššie, tým sa tkanina pohybuje vzduchom pomalšie. (Ak teda chcete jemnú hodvábnu šatku, zvolte `Silk` a potom túto hodnotu zvýšte.) `Quality` určuje kvalitu simulácie a `Mass` určuje hmotnosť tkaniny. Tri ďalšie hodnoty hovoria, odkiaľ gravitácia ťahá (keď chcete vedieť, odkiaľ vietor fúka, tak si tam dajte `Empty`, zapnete na ňom pole typu `Wind` a otočte ho ako chcete).



Obrázok 18: Cloth

Ak chcete, aby sa niektoré vrcholy nehýbali, spravíte to podobne, ako pri plastických telesách. Vytvoríte skupinu vrcholov, dáte do nej tie vrcholy, ktoré majú ostať stabilné, potom stlačíte tlačidlo `Pinning of cloth` (špendlenie plátna) a vyberiete z roletového menu tú správnu skupinu.

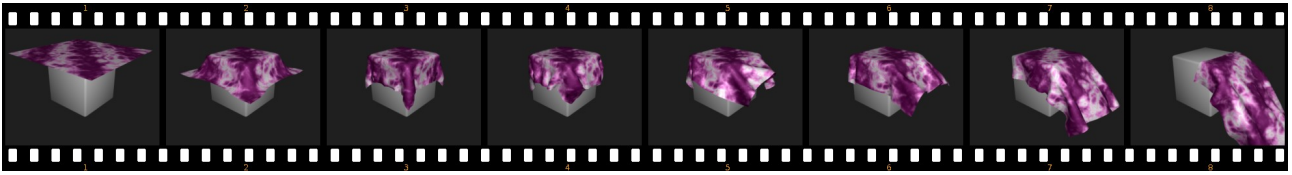


Obrázok 19: Collision

Na paneli `Collision` sa dejú podobné veci ako v prípade `Soft Body`. Jednak tam nastavujete, pre aký rozsah snímkou sa má fyzika počítať a prípadne tam môžete zmazať veci, čo sa dočasne uložili na disk a už nie sú aktuálne. Okrem toho tam zapínate, či má látka reagovať na kolízie so sebou a s inými objektami. Pri kolízii

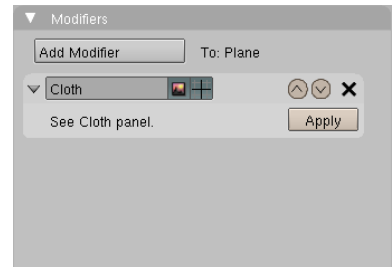
s inými objektami môžete nastaviť trenie (`Friction`). Hodnota `Min Distance` určuje minimálnu vzdialenosť, ktorú si tkanina udržuje od ostatných telies (alebo od seba samej). Ak by sa stalo, že šaty pretínajú podklad, túto hodnotu zväčšíte. Parameter `Collision Quality` hovorí, ako presne sa majú kolízie počítať. Čím je väčší, tým sa to počíta presnejšie, ale tým dlhšie to trvá. Panel v celej jeho kráse môžete vidieť na obrázku č. 19.

V ukázkovej animácii na obrázku č. 20 okrem padajúceho obrusu fúka na kocku zozadu dosť silný vietor, ktorý šatku nakoniec z kocky sfúkne. Ako materiál obrusu bola zvolená bavlna.



Obrázok 20: Bavlna a vietor

Príjemná stránka, ktorú majú spoločnú `Soft Body` aj `Cloth` je tá, že sú to modifikátory objektu. To znamená, že ich nemusíte nutne používať iba na animáciu. Môžete vytvoriť napríklad štvorcový obrus, potom ho nechať dopadnúť na stôl a keď nadobudne tvar, aký potrebujete (prepnite sa na vhodný snímok), v paneli `Modifiers` medzi tlačidlami úpravy stlačte pri patričnom modifikátore tlačidlo `Apply`. Panel aj tlačidlo môžete vidieť na obrázku č. 21. Modifikátor zmizne a deformácia, ktorú spôsobil obrusu sa stane jeho trvalou súčasťou. Týmto spôsobom si môžete výrazne ušetriť prácu pri modelovaní.

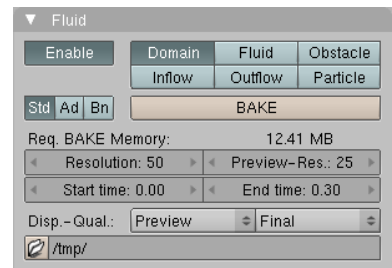


Obrázok 21: Modifiers

Kvapaliny

Ďalšia zaujímavá vec, ktorú môžete v Blenderi simulovať, sú kvapaliny. Nechať Blender počítať, ako niečo na scéne tečie či žblnká, je síce vec po výpočtovej stránke náročná, ale zato veľmi efektívna.

Prvý krok, ktorý musíte urobiť, keď chcete pracovať s kvapalinou je vytvoriť si doménu. Doména je kváder, v ktorom sa všetko, čo sa kvapaliny týka, bude odohrávať. Na scénu pridáte kocku (`SPACE`, `Add` → `Mesh` → `Cube`) a zväčšíte ju tak, aby obsahovala celý priestor, v ktorom sa bude vaša kvapalina pohybovať. Nemusíte ju zväčšovať v každom smere rovnako, kváder je rovnako dobrý ako kocka. Kým je doména aktívna, prepnete sa na tlačidlá s fyzikou a na paneli `Fluid` (kvapalina) stlačte tlačidlo `Enable`. Panel bude vyzerať tak, ako na obrázku č. 22. Všimnite si, že je stlačené tlačidlo `Domain`.



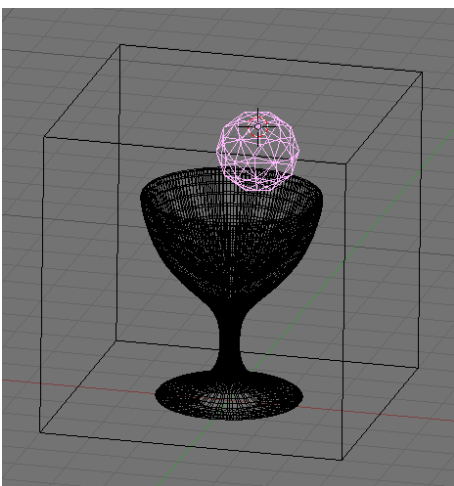
Obrázok 22: Fluid

Úlohou domény sú dve veci. Jednak to, že určuje, v akej oblasti sa kvapalina bude pohybovať. V tejto súvislosti treba spomenúť parameter `Resolution` (rozlíšenie). Totiž výpočet pohybu kvapaliny sa deje po priestorovej mriežke vo vnútri domény a tento parameter určuje, aká hustá tá mriežka bude. Čím je väčší, tým je simulácia kvalitnejšia, ale tým viac pamäte to zožerie. Koľko presne pamäte to bude, sa môžete

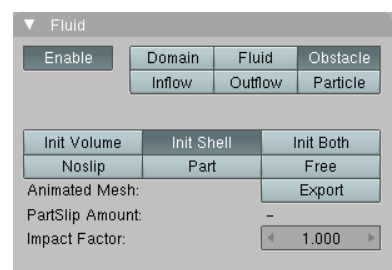
dočítať priamo na paneli `Fluid`. Dajte pozor, aby to nepresiahlo reálne možnosti vášho počítača. Druhou úlohou domény je to, že sa pre ňu nastavuje materiál a ďalšie vlastnosti samotnej kvapaliny.

Pre náš úvodný príklad si vytvoríme scénu, akú môžete vidieť na obrázku č. 23. Kocka okolo zúčastnených objektov je doména, o ktorej sme hovorili. Ďalšie dva objekty – pohár a guľa s kvapalinou sa musia nachádzať vo vnútri domény.

Každému z týchto objektov musíme prideliť jeho úlohu v procese simulácie kvapaliny. Najprv si všimneme



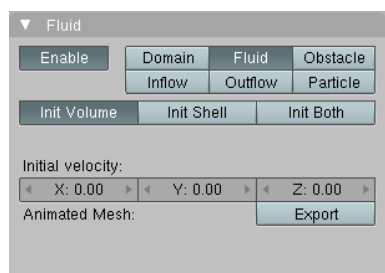
Obrázok 23: Scéna s kvapalinou



Obrázok 24: Fluid

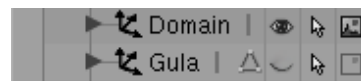
pohára. Tomu treba nastaviť, že má pre kvapalinu znamenať prekážku. Tiež sa prepnete na panel Fluid, zapnete Enable a z možných úloh vyberte Obstacle (prekážka). To spôsobí, že voda nebude cez pohár pretekať. Náš pohár je solídne teleso, ktoré má vnútrajšok aj vonkajšok. Keby sa ale pohár skladal iba z jednej vrstvy (napríklad keby ste z kocky odstránili hornú stenu a vzniklo vám tak akvárium) je dôležité namiesto tlačidla Init Volume (inicializovať objem) zapnúť tlačidlo Init Shell (inicializovať škrupinu). V prvom prípade totiž tvorí prekážku celý objem telesa (a voda by teda do akvária vôbec nenatiekla). V druhom prípade tvorí prekážku iba povrch telesa a keďže hore stena chýba, voda dnu natečie.

Ďalšie tri tlačidlá určujú, ako sa prekážka bude ku kvapaline správať. Noslip reprezentuje drsný povrch na ktorom kvapalina zanecháva kvapky. Free znamená, že sa kvapalina po prekážke kľže a nezmača ju. A ak zvolíte Part, môžete nastaviť, ako veľmi sa budú kvapky tvoriť. Hodnota 0 znamená to isté, ako Noslip, hodnota 1 to isté ako Free.



Obrázok 25: Fluid

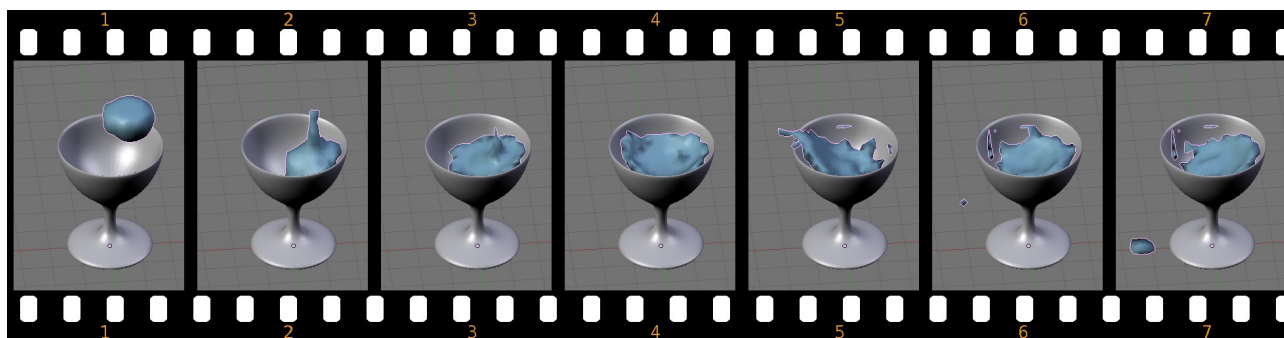
Guli, ktorá bude reprezentovať počiatočný stav kvapaliny, treba tiež na paneli Fluid zapnúť Enable a z možností, ktoré sú k dispozícii si treba vybrať Fluid. Celá guľa sa vyplní kvapalinou, ktorá bude ponechaná napospas fyzike. Okrem toho sa ale bude stále renderovať aj samotná guľa. Ak tomu chcete zabrániť a mať viditeľnú iba kvapalinu, vypnite guľu viditeľnosť aj renderovanie v osnove (pozrite obrázok



Obrázok 26: Osnova

č. 26). Ak nechcete všetko ponechať iba na gravitáciu, môžete kvapaline nastaviť aj nejakú počiatočnú rýchlosť. Panel Fluid s voľbou Fluid môžete vidieť na obrázku č. 25.

Keď je všetko prichystané (do fázy chystania patrí aj pridanie prípadných nekvapalinových animácií), aktivujte znovu doménu a na paneli Fluid stlačte tlačidlo BAKE. Nezľaknite sa – v tomto momente hranica domény zmizne a stane sa z nej kvapalina samotná. Prebehne výpočet a môžete renderovať. Ak sa ale zmenia nejaké dôležité okolnosti či nastavenia, treba tlačidlo BAKE stlačiť znova.¹ Výslednú simuláciu (tentokrát nie vyrenderovanú, ale ako snímky z 3D okna) si môžete pozrieť na obrázku č. 27.

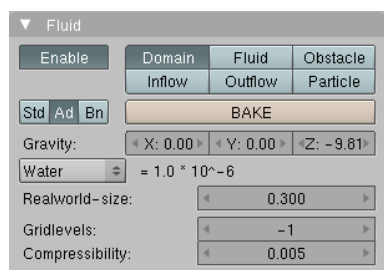


Obrázok 27: Kvapalina

Dobre. Teraz, keď už viete, ako to funguje, poďme sa pozrieť na niektoré ďalšie nastavenia. Vráťme sa najskôr k doméne a k panelu z obrázka č. 22. Okrem rozlíšenia, ktoré sa použije pri renderovaní tam môžete nastaviť aj rozlíšenie, ktoré sa použije v zobrazení v 3D okne (hodnota Preview-Res.). Veci, ktoré je ale dôležité nastaviť, sú parametre Start time a End time. Totiž simulácia kvapaliny nebeží podľa blenderového času ale podľa svojho súkromného. A tento súkromný čas je treba z blenderovým zladiť. Ak vaša animácia končí snímkom č. 250, tak pri frekvencii dvadsaťpäť snímkov za sekundu trvá desať sekúnd. Ak teda chcete, aby sa kvapalina správala fyzikálne verne, musíte nastaviť Start time na 0 a End time na 10. (Čas pre Fluid sa udáva v sekundách.) Ak chcete spomalený záber, nastavte End time na 2 a celý dej bude

¹ Pozor! Vo verzii 2.47 a starších sa na rozdiel od častíc a pružných telies, keď niečo podstatné zmeníte, výpočet sa neudeje automaticky odznova ale musíte stlačiť BAKE. Možno sa to časom zmení, ale zatiaľ je to takto.

prebiehať päťkrát pomalšie. Ak tam nechcete na začiatku animácie tú hrču vody nad pohárom, ale iba rozhojdanú hladinu, nastavte *Start time* na 0,5. Na paneli sú ďalej dve roletové menu. V prvom z nich si môžete vybrať, či sa v 3D okne bude kvapalina zobrazovať v kvalite *Preview* (prehliadať) alebo *Final* (výsledná kvalita) alebo *Geometry* (kedy za ukáže pôvodná kocka tvoriaca doménu). V druhom menu si môžete vybrať, v akej kvalite má byť renderovaný výsledok. Do vstupného poľa pod tým môžete zadať adresár, do ktorého sa budú ukladať predpočítané dáta celej simulácie.

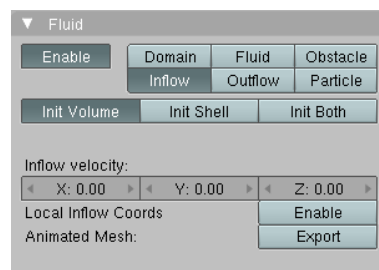


Obrázok 28: Advanced options

Ak stlačíte tlačidlo *Ad*, objavia sa ďalšie nastavenia. Panel môžete vidieť na obrázku č. 28. Jednak tu môžete určiť smer pôsobenia gravitácie (hodnoty X, Y a Z), jednak si v roletovom menu môžete nastaviť viskozitu kvapaliny – ak chcete vodu, alkohol alebo niečo podobné, voľte *Water* (voda), ak chcete olej, vyberte *Oil* a ak chcete med, voľte *Honey*. Ak by ste chceli niečo ešte hustejšie, napríklad bahno, môžete vybrať možnosť *Manual* (ručné nastavenie) a nastaviť takú viskozitu, aká sa vám páči. Hodnota *Realworld-size* určuje dĺžku najdlhšej strany domény v metroch. Túto hodnotu

je tiež dobré nastaviť kvôli tomu, aby vedel Blender previesť simuláciu fyzikálne verne.

Pristavme sa na chvíľu pri ďalších objektoch, ktoré môžete zvoliť na paneli *Fluid*. Objekt typu *Inflow* (prítok) je objekt, z ktorého kvapalina neustále tečie. Skrátka niečo ako rieka nad vodopádom alebo vodovodný kohútik. Hodnotami X, Y a Z môžete nastaviť rýchlosť a smer, ktorým kvapalina vyteká a ak stlačíte tlačidlo *Enable* pri nápis *Local Inflow Coords* (lokálne súradnice pritekajúceho), prúd vody bude meniť smer podľa toho, ako budete objektom so zapnutým *Inflow* točiť. Podobne objekt typu *Outflow* (odtok) je objekt, ktorým kvapalina zo systému vyteká.



Obrázok 29: Inflow

Poslednou možnosťou o ktorej sme sa ešte nez zmienili je možnosť *Particle* (častice). Keď totiž dopadá masa vody na nejaký povrch, často z nej odletujú drobné kvapky, ktorých objem neovplyvňuje fyziku celej vodnej masy nijako významne, ale tie kvapky je vidno a bez nich to tak nejako nie je ono. Skúste porovnať snímky na obrázku č. 30. Na ľavom snímku vyzerá voda ako odliata z vosku. Na obrázku vpravo ale vidíte kvapky,

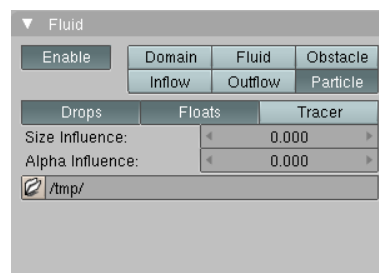


Obrázok 30: Častice

ktoré tomu dodajú väčšiu vierohodnosť. A práve tieto kvapky sa vytvárajú s pomocou objektu typu *Particle*.

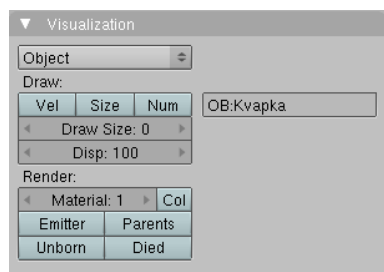
Objekt typu *Particle* treba pridať niekam do domény, pričom vôbec nezáleží na tom, ako bude vyzeráť ani kde presne bude umiestnený. Stačí, že do domény taký objekt vložíte a poloha a pohyb častíc sa bude počítať automaticky. Panel *Fluid* vyzerá v tomto prípade tak, ako môžete vidieť na obrázku č. 31.

Prvé tri tlačidlá určujú, aký typ častíc sa má generovať. *Drops* (kvapky) sú drobnejšie kvapky, ktoré vznikajú z kvapalín, ktoré majú nízke povrchové napätie (ako napríklad voda). *Floats* (bóje) sú väčšie kvapky, ktoré vznikajú z kvapalín s vyšším povrchovým napätím. Sú väčšie a rýchlejšie padajú späť do kvapaliny. *Tracer* (stopár) generuje



Obrázok 31: Particle

častice, ktoré vznikajú tam, kadiaľ sa práve nejaká masa vody prevalila. Tieto častice sa ale už príliš nehýbu a ostávajú na mieste. Experimentujte s tým, ale nič vážne nepokazíte, ak zapnete prvé dve, prípadne všetky tri možnosti. Ak zväčšíte hodnoty pri parametroch *Size Influence* alebo *Alpha Influence*, častice budú meniť svoju veľkosť, prípadne podľa veľkosti aj priehľadnosť. Do vstupného poľa môžete určiť, kam sa majú uložiť predpočítané dáta. Väčšinou sa používa ten istý adresár, ako na dáta zo samotnej simulácie kvapalín.

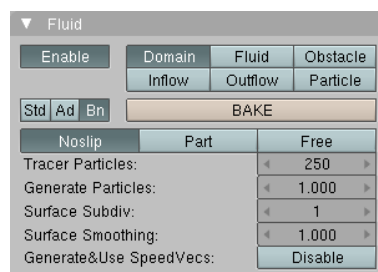


Obrázok 32: Visualisation

Keď sa prepnete v okne tlačidiel na tlačidlá častíc, Blender vás pustí iba k panelu *Visualisation*, na ktorom môžete nastaviť, ako budú častice vyzeráť. (Ak nevidíte ani ten, skúste nájsť pole **1 Part 2** a kliknutím na šipku vľavo sa prepnete na časticový systém č. 1.) Panel vidíte na obrázku č. 32. Na našom paneli sme ako tvar častíc zvolili objekt *Kvapka*, ktorý je malá guľôčka (UV sféra so štyrmi rovnobežkami a poludníkmi), ktorá sa nachádza v počiatku súradnicovej sústavy a má nastavený rovnaký materiál, ako voda. Detaily ohľadom nastavovania toho, ako sa majú častice zobrazovať,

sa dočítate v lekcii o časticách.

Aby ale častice v simulácii kvapaliny fungovali, je ale potrebné spraviť ešte jednu vec. Aktivujte doménu a stlačte tlačidlo *Bn* (*Boundary options* – okrajové nastavenia). Tam si môžete vybrať, či sa má samotná kvapalina správať viac ako voda, teda či bude mať nízke povrchové napätie a na materiáloch bude vytvárať kvapky (*Noslip*), alebo či bude po veciach stekať (*Free*) alebo niečo medzitým (*Part*). S týmito nastaveniami sme sa už stretli pri objektoch typu *Obstacle*. S časticami ale nesúvisia. Ak chceme, aby nám častice fungovali, dôležité sú iné nastavenia. Najdôležitejšia je hodnota *Generate Particles*, ktorá určuje, ako mnoho častíc sa bude generovať. Ak tam necháte nulu, nebude sa generovať nič. Ak tam dáte jednotku, bude sa generovať štandardné množstvo častíc. A ak dáte väčšie číslo, bude častíc viac. Táto hodnota sama o sebe má vplyv na pohyb kvapaliny aj vtedy, keď nepridáte žiadny objekt typu *Particle*. Stačí, že ju zväčšíte a kvapalina bude mať väčší sklon rozpadávať sa na kvapky. Hodnota *Tracer Particles* určuje množstvo trasovacích častíc.

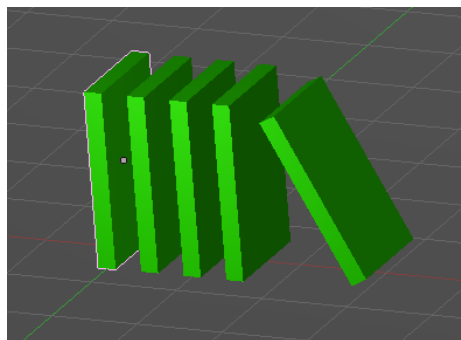


Obrázok 33: Boundary options

Ďalšie dve hodnoty tiež nesúvisia s časticami, ale keď už opisujem tento panel, tak sa o nich zmienim. *Surface Subdiv* je niečo podobné, ako modifikátor *Subsurf*. Ak má hodnotu 1, nič sa nedeje, každé zväčšenie tejto hodnoty o 1 spôsobí dvojnásobné zahustenie povrchu plochy. Dajte pozor, aby sa vám to vošlo do pamäte. *Surface Smoothing* sa zas stará o vyhladzovanie povrchu kvapaliny. Čím vyššie číslo, tým bude povrch hladší, ale tým väčšie nároky na pamäť.


V súvislosti s kvapalinami by som ešte chcel spomenúť pekný videotutoriál, ktorý vytvoril Antonio Gessi a ktorý môžete nájsť na adrese <http://www.pkblender.it/FluidSim/domain.htm>

Herný systém

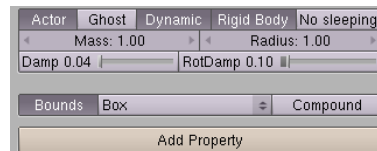


Obrázok 34: Domino

Blender obsahuje zaujímavý subsystém, ktorý sa dá použiť na vytváranie hier. Nie je to však jeho jediné použitie. Môžete ho pokojne využiť na simuláciu fyzikálnych dejov. Vytvorte si napríklad scénu, akú môžete vidieť na obrázku č. 34. Na objekte typu *Plane*, ktorý tvorí dlážku, stojí päť kvádrov, ktoré predstavujú dominové kocky. Posledný z nich je nahnutý, takže by spadol, keby bol ponechaný na gravitáciu. No a teraz nám pôjde o to, ako to zariadiť, aby bol skutočne na gravitáciu ponechaný.

V okne tlačidiel sa prepnete na tlačidlá logiky (buď ikonou , alebo tlačidlom F4). Zatiaľ čo ostatné tlačidlá sú rozdelené do jednotlivých panelov, pri tlačidlách hernej logiky to tak nie je a všetky tlačidlá sú umiestnené vedľa seba po celej šírke panela. Má to dobrý dôvod, s ktorým sa oboznámime neskôr, zatiaľ je dobré si pamätať, že ak chcete pracovať s herným subsystémom, je dobré si okná zorganizovať tak, aby okno tlačidiel bolo umiestnené skôr na šírku, než na výšku. Teraz sa ale budeme zaoberať ľavou časťou tlačidiel, ktorú môžete vidieť na obrázku č. 35.

Aktivujte nahnutú dominovú kocku a stlačte tlačidlo Actor (herec). Tým zapnete, že sa kocky začnú týkať nastavenia, ktoré súvisia s herným systémom. Okrem toho sa tam objavia ďalšie dve tlačidlá – Ghost a Dynamic. Ghost spôsobuje, že sa objekt správa ako duch – iné predmety cez neho prepadávajú. To momentálne nepotrebujeme. Dynamic znamená, že pre objekt začína platiť fyzika. Ak ho stlačíte, objavia sa aj ďalšie s tým súvisiace tlačidlá. S tou platnosťou fyziky je to ale zatiaľ dosť obmedzené. Bez ohľadu na to, ako objekt vyzerá, správa sa zatiaľ ako guľôčka s polomerom určeným parametrom Radius, ktorá sa navyše nemôže otáčať a na jej tvare vôbec nezáleží. Môže ale už narážať do ďalších objektov (pri čom sa prejaví parameter Mass – hmotnosť) a ak ho umiestnite niekde nad dĺžku, tak už na ňu vie padnúť. Keď zapnete aj tlačidlo Rigid Body, teleso sa už bude vedieť otáčať a nebude stále natočené rovnako. Stále sa ale bude správať ako guľa. Ak chcete, aby Blender počítal s komplikovanejšími tvarmi, stlačte tlačidlo Bounds (hranice) a z roletového menu vyberte, ako sú určené hranice telesa. V prípade kvádra (ono dominová kocka vlastne nie je kocka, ale kváder) stačí vybrať Box (krabica). Ak máte nejaké zložitejšie teleso, vyberte možnosť Convex Hull Polytope (konvexný obal) ale snažte sa, aby boli vaše telesá čo najjednoduchšie, aby to Blender stihol počítať v reálnom čase.

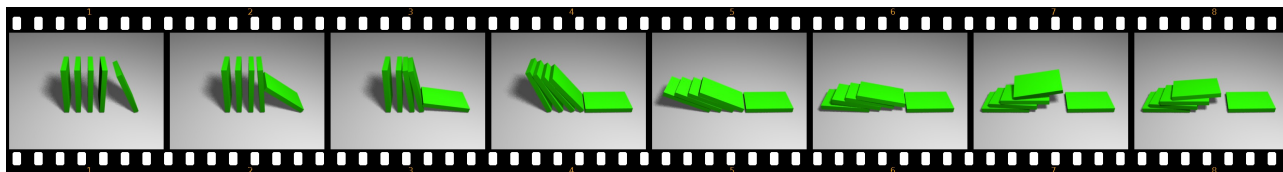


Obrázok 35: Logic buttons

Keď ste teda všetko pekne nastavili, stlačte v 3D okne klávesu P. Môžete sa kochať, ako sa vaša dominová kocka kláti k zemi. Teraz rovnako nastavte aj ostatné kocky a dominový vláčik môže začať fungovať. Keď to všetko popadá a chcete, aby simulácia prestala, zastavíte to klávesou Esc.

Ak chcete, aby to počas behu herného systému vyzeralo lepšie, prepnete spôsob zobrazenia z drôteného na tieňovaný (Shaded) alebo textúrový (Textured). Možno sa vám pri štarte herného systému stane, že kocky, ktoré stoja na dlážke jemne poskočia. Je to tým, že sú jemne v dlážke zaborené. Ak tomu chcete predísť, posuňte dlážku o maličký kúsok nižšie, alebo kocky o malý kúsok vyššie.

Ak by vám nestačilo iba sa pozerať, ako pekne to v hernom systéme funguje a chceli by ste to vyrenderovať ako animáciu, treba sa obrátiť na menu v používateľských nastaveniach (väčšinou výva v Blenderi úplne hore). Tam zapnete položku Game a vyberte Record Game Physics to IPO (zaznamenať hernú fyziku do IPO). Okrem toho treba v tom istom menu zapnúť položku Enable All Frames (vybrať všetky rámce). Keď teraz znovu spustíte klávesou P herný stroj, pohyb kociek sa zaznamená s pomocou IPO kriviek. Túto animáciu potom môžete prehrať aj s pomocou ALT-A alebo vyrenderovať za prispenia ray-tracingu, tieňov, zaujímavého osvetlenia a iných šikovných vecí, ktoré priamo v hernom stroji k dispozícii nie sú, pretože by ich renderovanie trvalo prídlho. Výsledný efekt môžete vidieť na obrázku č. 36.



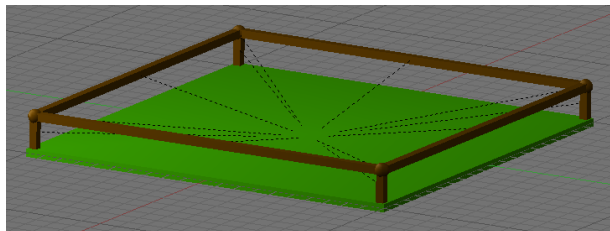
Obrázok 36: Dominový efekt

Hra

Herný systém môžete využívať pre tvorbu animácií. Môžete sa ale rozhodnúť, že sa vydáte ešte úplne inou cestou – cestou tvorby hier. Herný systém bol totiž vyvinutý presne za týmto účelom. A rovnako, ako existujú v Blenderi machri modelári či machri animátori, existujú ľudia,

ktorí vedú v Blenderi vyrábať hry, či už sa jedná o jednoduché logické hádanky, alebo o FPS² hry so zložitým modelom prostredia.

Základnú funkčnosť herného systému môžu ľudia, ktorí programujú, doplniť o kód v jazyku Python. V tejto lekcii ale predvedieme iba jednoduchú ukážku, v ktorej sa žiadny kód písať nebude a ktorá bude využívať len to, čo je k dispozícii priamo. Inšpiráciou nám bude klasická open-source hra Neverball.³

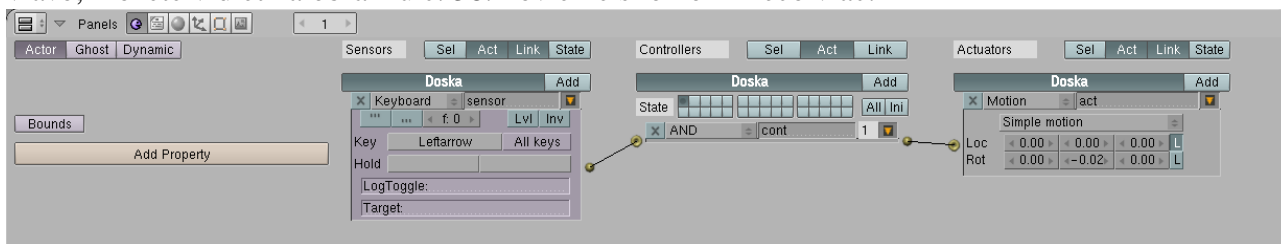


Obrázok 37: Základná doska

Začneme tým, že vyrobíme základnú dosku. Základnú plochu vyrobíme škálovaním kocky a pridáme zábradlie. Jednotlivé prvky zábradlia modelujte ako samostatné objekty, pretože tie sa hernému stroju vyhodnocujú ľahšie, než zložené nekonvexné teleso. Keď sú hotové, tak im základnú dosku nastavte ako rodiča, aby sa pohybovali spolu s ňou. Potom základnej doske aj zábradliu nastavte

nejaké zaujímavé materiály. Výsledok by mohol vyzeráť podobne, ako na obrázku č. 37. Tie prerušované čiary, ktoré idú od jednotlivých častí zábradlia k základnej doske znázorňujú vzťah rodič – potomok.

Základnú dosku máme hotovú. Teraz by sme chceli dosiahnuť, aby sa nakláňala na jednotlivé strany, keď stlačíme patričnú šípku. Nastavenie, ktoré spôsobí, že doska bude reagovať na šípku vľavo, môžete vidieť na obrázku č. 38. Povieme si o ňom niečo viac.



Obrázok 38: Senzor, ovládač a akcia

Takže aktivujte dosku a prepnite sa na tlačidlá s hernou logikou. Stlačte tlačidlo Actor, aby si herný systém začal dosku všímať, ale nezapnite Dynamic, pretože by bez inej pevnej dlážky doska spadla niekam dole a to nechceme.

Teraz si všimnite zvyšok panelu. Nachádzajú sa v ňom tri stĺpce, prvý s názvom Sensors (senzory), druhý s názvom Controllers (ovládače) a tretí s názvom Actuators (akcie⁴). Funguje to tak, že senzory dávajú pozor, či sa niečo neudialo, ovládače sledujú stav senzorov a ak nastane ich vhodná kombinácia, zavolajú určené akcie.

Senzory reagujú na udalosti, ktoré nastali. Môžete si vybrať medzi udalosťami spôsobenými klávesnicou (Keyboard), myšou (Mouse), vzájomnou polohou objektov (Collision alebo Touch – kolízia, Near – priblíženie sa k objektu, Ray a Radar – detekcia objektu v danom smere), prijatím správy od iného objektu (Message), zmenou nejakej vlastnosti (Property) alebo udalosťou, ktorá nastáva stále (Always).

Ak chcete pridať nový senzor, kliknite na tlačidlo Add. Objaví sa udalosť typu Always. V našom prípade ale chceme, aby doska reagovala na klávesnicu, takže kliknite na dvojšípku a z roletového menu vyberte možnosť Keyboard. Vedľa nápisu Key (klávesa) uvidíte tlačidlo bez nadpisu. Kliknite naň. Objaví sa v ňom nápis Press a key (stlačíť klávesu). Stlačte tú klávesu, ktorá má danú udalosť vyvolať (v našom prípade šípku vľavo). V políčku sa objaví meno danej klávesy a senzor je pripravený. Skontrolujte, či vyzerá rovnako, ako na obrázku č. 38. Ak vám nezáleží na tom, aká klávesa bola stlačená, namiesto voľby klávesy stlačte tlačidlo All keys.

² First-person shooter, strieľačka pri zobrazení z pohľadu hlavnej postavy.

³ <http://icculus.org/neverball>

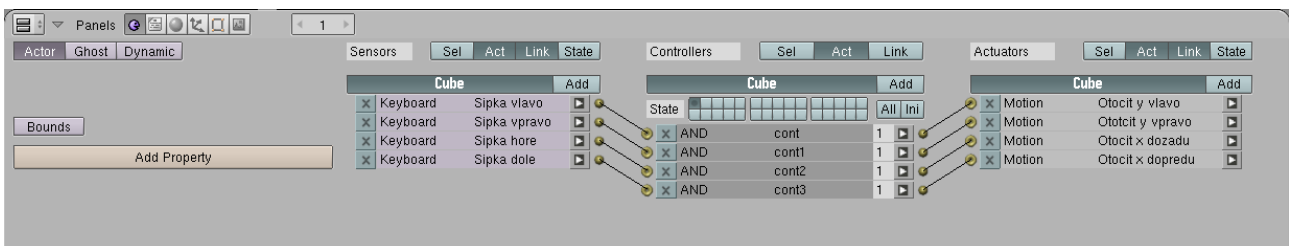
⁴ Preklad nie je celkom presný, actuator je niečo ako „výkonný člen“. Ale slovo „akcia“ je v danom kontexte výstižnejšie.

Ovládač má v danom prípade jednoduchú robotu, keďže sleduje iba jeden senzor. Pridajte ho (tlačidlom Add v sekcii *Controllers*), nechajte nastavené AND (znamená to, že na to, aby tento ovládač zavolať patričné akcie, musí dostať povolenie od všetkých senzorov, ktoré sú k nemu pripojené, ale keďže bude iba jeden, tak je to jedno) a myšou spojte ovládač so senzorom (kliknite na guľôčku vľavo od ovládača a potiahnite čiaru ku guľôčke senzora).

Posledná vec, ktorú potrebujeme pridať, je akcia. Stlačte tlačidlo Add v sekcii *Actuators*. Keďže chceme doskou točiť, nechajte zvolenú možnosť *Motion* (pohyb) a v riadku *Rot* napíšte do druhého stĺpca, ktorý sa týka otáčania okolo osi *y*, hodnotu $-0,02$. Okrem toho vypnite to tlačidlo *L* vľavo, aby sa otáčanie dialo okolo hlavných súradnicových osí a nie podľa toho, ako je práve natočená súkromná súradnicová sústava dosky. Akciu pripojte k ovládaču podobne, ako ste k nemu pripájali senzor.

Skontrolujte, či máte všetko nastavené tak, ako na obrázku č. 38. Keď teraz v 3D okne stlačíte klávesu **P** a spustíte herný stroj, doska by mala reagovať na šípku vľavo a mala by sa otáčať. (Pripomínam, že herný stroj zastavíte klávesou **Esc**.) Keď nechcete, aby vám senzor, ovládač a akcia zaberali v okne hernej logiky priveľa miesta, stlačte pri každom ten malý oranžový trojuholníček a zbaľte ich do kompaktnejšej podoby. Kedykoľvek ich budete môcť znovu rozbaľiť. Aby ste sa v tom dobre orientovali, dajte im nejaké zmysluplné názvy.

Úloha č. 1: Dorobte senzory, ovládače a akcie na ostatné tri šípky. V zabalenej podobe to (po patričnom premenovaní) bude vyzeráť tak, ako na obrázku č. 39. Dajte pozor, aby ste všetky senzory nepripojili k tomu istému ovládaču, ale každej vyrobili samostatný ovládač. Pri aktuálnom nastavení by ste totiž museli stlačiť všetky štyri šípky naraz, aby sa vôbec niečo dialo.



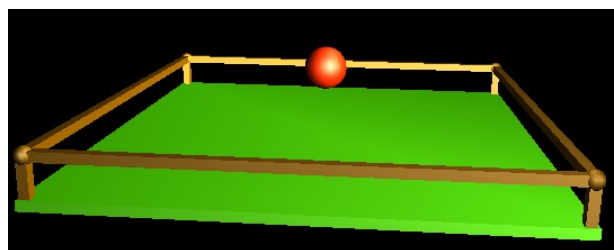
Obrázok 39: Ovládanie dosky

Skvelé. Doskou už teda môžeme hýbať. Aby to vyzeralo lepšie, prepnite spôsob zobrazenia v 3D okne na *Shaded* (prípadne rovno *Textured*, to nevedí, že tam žiadne textúry nie sú) a nejakto to osvetlite.



Obrázok 40: Nastavenie lopty

Aby bolo s čím hru vôbec hrať, pridáme na scénu loptu. Pridajte UV guľu s polomerom 1 niekam nad dosku, pridajte jej materiál podľa vášho vkusu a nastavte ju v okne hernej logiky podľa obrázka č. 40. Žiadne udalosti nenastavujte a všetko ponechajte na fyziku. Keď spustíte herný systém, zistíte, že môžete točiť doskou a guľa sa bude po nej kotúľať podľa toho, ako dosku práve nakloníte.

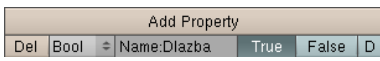


Obrázok 41: Doska s guľou

Všetko by bolo fajn, ale keď sa s guľou chvíľu hráte, zistíte, že vám môže preskočiť cez zábradlie, prípadne, že celú dosku môžete otočiť naopak a guľa spadne niekam dole. A vtedy je guľa nenávratne preč a neostáva nič iné, než zastaviť herný systém a spustiť ho od začiatku. Bolo by príjemné, keby sa to dalo zariadiť tak, že v týchto prípadoch sa hra jednoducho reštartuje. To ale môžeme zariadiť.

Pridajte dostatočne hlboko pod dosku objekt typu *Plane* a zväčšite ho natolko, aby ho guľa nemohla minúť. Bude predstavovať dlažbu. Keď guľa na túto dlažbu dopadne, potrebujeme zariadiť, aby sa hra reštartovala.

V prvom rade potrebujeme, aby mohla guľa odlíšiť dlažbu od ostatných objektov. Dlažba musí mať nejakú vlastnosť, ktorú iné objekty nemajú. To zabezpečíme tak, že aktivujeme dlažbu a v okne s hernou logikou stlačíme tlačidlo Add Property (pridať vlastnosť). Pod tlačidlom sa objaví riadok, ktorý môžete vidieť na obrázku č. 42. V ňom je momentálne najdôležitejšia kolónka Name (meno). Nastavili sme ho na Dlazba.



Obrázok 42: Vlastnosť

Okrem toho, že sme objektu pridali vlastnosť, môžeme nastaviť, aká má tá vlastnosť vlastne byť. V tomto prípade sme nastavili, že to je logická hodnota (Bool) a že pri štarte herného systému má byť táto hodnota True (pravda). Na to, na čo ale teraz hodláme vlastnosť Dlazba použiť, ale typ a aktuálna hodnota tejto vlastnosti nemá žiaden vplyv.

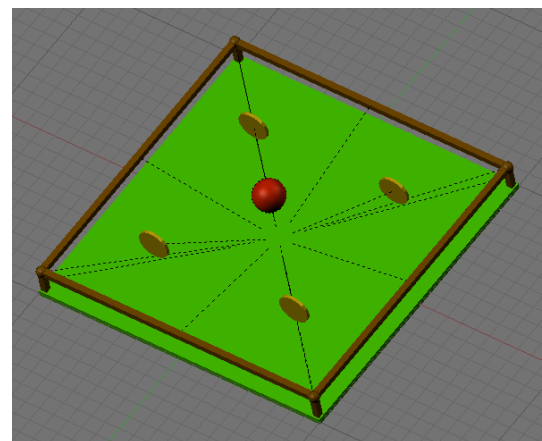
Aktivujte guľu, ktorá predstavuje loptu a nastavte jej reakciu na udalosť tak, ako môžete vidieť na obrázku č. 43. Senzor bude očakávať udalosť typu Collision (zrážka) a to konkrétne s objektom, ktorý má vlastnosť Dlazba. Ak táto udalosť nastane (a pritom je úplne jedno, či je hodnota vlastnosti Dlazba True, False alebo – ak je jej typ Float – 3,14, stačí, že objekt tú vlastnosť má), ovládač zavolá akciu typu Scene, ktorá scénu reštartuje (vybrali sme možnosť Restart). Akcia typu Scene vás môže okrem toho napríklad prepnúť na inú scénu s ďalším levelom (možnosť Set Scene) alebo môže vymeniť kameru (Set Camera).



Obrázok 43: Kolízia gule s dlažbou

Je tu ale ešte drobný problém – nie vždy stojíme o to, aby bola dlažba viditeľná. Ak chcete, aby počas hry dlažbu vidieť nebolo, aktivujte ju, prepnete sa na tlačidlá úpravy a tam na paneli Mesh vypnite tlačidlo Double Sided. Dlažba bude viditeľná iba z jednej strany. Ak je to tá strana, ktorú vidíte pri hraní, otočte dlažbu o 180 stupňov, takže bude hore neviditeľnou stranou. Stále ale bude reagovať na kolízie, takže bude plniť aj naďalej svoju funkciu zachytávača lôpt.

Aby sme loptou len tak bezcieľne nepohadzovali sem a tam a aby hra mala aj nejaký cieľ, pridajte do hry štyri mince (objekty typu Cylinder, ktoré patrične vyškálujete a stenám po obvode zapnete Set Smooth). Nastavte im vhodný materiál, aby vyzerali, ako zlaté. Aby nemenili polohu vzhľadom na dosku, nastavte im dosku ako rodiča. Umiestnite ich tak, aby boli kúsok nad doskou, ale dosť ďaleko od stredu (aby na ne guľa priamo nepadla) a dosť ďaleko od zábradlia (aby nebolo také jednoduché ich guľou trafiť tak, že ju budete kotúľať popri zábradlí).



Obrázok 44: Mince

Aby bolo jasné, že práve mince sú tým, o čo v hre ide, bolo by fajn, keby sa nejako výrazne pohybovali, napríklad, keby sa otáčali okolo vlastnej osi. To si môžeme zabezpečiť s pomocou senzora typu Always. Nastavenie môžete vidieť na obrázku č. 45. Udalosť Always nastáva stále, tým pádom sa minca bude stále otáčať okolo osi z. V tomto prípade je vhodné nechať zapnuté to tlačidlo L v riadku

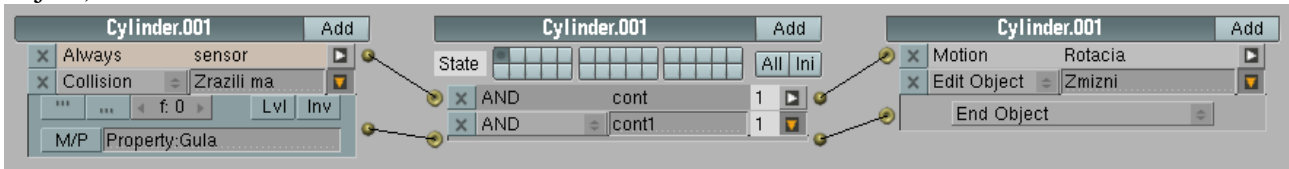


Obrázok 45: Otáčanie mince

Rot, pretože minca sa pohybuje spolu s doskou a tým pádom stále mení natočenie vzhľadom na

„globálnu“ os z. Keď ju necháme točiť sa okolo jej lokálnej osi, pohyb stola sa do toho nebude nevhodne miešať. (Môžete to tlačidlo prípadne vypnúť a pozrieť sa, čo to robí bez neho.)

Okrem toho, že sa mince točia, by ale mali ešte aj miznúť, keď do nich guľa narazí. To budeme riešiť podobne, ako kolíziu s dlažbou. Najprv guľu pridáme vlastnosť `Gula`. Nezáleží na tom, aký bude mať typ alebo hodnotu. Potom každej minci nastavíme, že ak dôjde ku kolízii s objektom, ktorý má vlastnosť `Gula`, že má zmiznúť. To, že má minca zmiznúť, nastavíme s pomocou akcie typu `Edit Object` (upraviť objekt) a to konkrétne `End Object` (ukončiť objekt).



Obrázok 46: Kolízia

Keď takto nastavíte všetky štyri mince a spustíte hru, zistíte, že mince utešene reagujú na kolíziu a miznú. Lenže na kolíziu reaguje aj lopta a pri zrážke s mincou zmení smer. To by principiálne nemuselo vadiť, ale keď chcete, aby sa to nedialo, použite namiesto senzora typu `Collision` senzor typu `Near`. Tiež mu nastavte, že má reagovať na objekty s vlastnosťou `Gula`, ale reakčnú vzdialenosť (`Dist`) nastavte na 1,1. Minca potom zmizne o kúsok skôr, než do nej stihne guľa, ktorá má polomer 1, vrazí a dráhu guľe tak už nič nezmení.

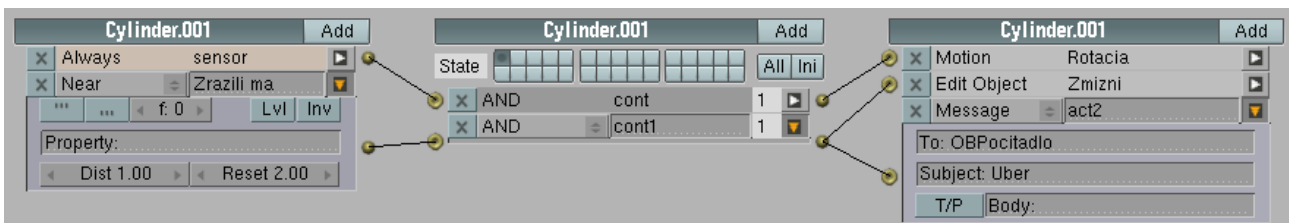
Mince utešene miznú. Teraz by to chcelo nejako zariadiť, aby hra vedela zistiť, že už sú všetky mince pozbierané a hráč vyhral. Na to treba nejako spočítať, koľko mincí je ešte v obehu. Na



Obrázok 47: Vlastnosť počítadla

to si zriadime ďalší objekt – `Pocitadlo`. Bude to objekt typu `Empty`, pretože nechceme, aby ho bolo na scéne vidieť a je jedno na ktoré miesto scény ho pridáte. Tomuto objektu vytvoríme vlastnosť `Pocet` a tentokrát nám bude záležať na tom, aká vlastnosť to je. Bude to vlastnosť typu `Int`, to znamená, že môže nadobúdať celočíselné hodnoty a počiatočná hodnota, ktorú vlastnosť nadobudne vždy pri začiatku hry, bude štyri. Nastavené to bude tak, ako môžete vidieť na obrázku č. 47. To stlačené tlačidlo `D` znamená, že hodnota tejto vlastnosti sa počas hry bude neustále vypisovať na obrazovku. Je to užitočné, aby ste videli, či vám veci fungujú, ale keď už budete mať všetko hotové, môžete to vypnúť, nech to nekazí dojem.

Keď sme si už spravili počítadlo, treba zabezpečiť, aby zakaždým, keď minca zmizne, poslala počítadlu správu, že sa tak stalo. Každéj minci teda pridajte novú akciu typu `Message` (správa) a nastavte jej, že má správu poslať počítadlu. To sa spraví tak, že do kolónky `To:` napíšete meno cieľového objektu, pred ktoré pridáte `OB`. Takže ak posielate správu objektu `Pocitadlo`, napíšete tam `OBPocitadlo`. Nastavte aj predmet správy (kolónka `Subject`) na `Uber`. Keďže sa táto akcia má vykonať vtedy, keď minca zmizne, pripojte ju na ten istý ovládač. Výsledné nastavenie je na obrázku č. 48.



Obrázok 48: Akcia typu Message

Teraz treba zariadiť, aby počítadlo správu zachytilo a upravilo patrične svoju vlastnosť `Pocet`. Takže aktualizujeme počítadlo a pridáme mu senzor typu `Message`. Do kolónky `Subject` napíšeme `Uber`, aby počítadlo reagovalo len na správy s týmto predmetom. Akcia, ktorú tento senzor bude vyvolávať, bude typu `Property`. Necháme zvolenú možnosť `Assign` (určiť). Do kolónky `Prop` napíšeme meno vlastnosti, ktorú chceme meniť – v našom prípade je to `Pocet` – a do kolónky `Value` napíšeme, akú hodnotu má daná vlastnosť získať, pričom sa môžeme odvolať aj

na pôvodnú hodnotu danej vlastnosti. Keď tam teda napíšeme $Pocet - 1$, spraví to to, že sa od pôvodnej hodnoty odčíta jednotka a toto číslo sa zapíše ako nová hodnota vlastnosti *Pocet*. (Pre programátorov: je to klasické priradenie.) Nastavenie môžete vidieť na obrázku č. 49. Keď hru spustíte a pri vlastnosti *Pocet* máte zapnuté tlačidlo *D*, mali by ste vidieť, že sa po kolízii s mincou hodnota znižuje.



Obrázok 49: Príjem správy

Keď teda vieme, koľko mincí je aktuálne na ploche, bolo by fajn, keby hra vypísala nejaký uznanlivý výrok, keď hráč pozberie všetky štyri mince. Na to je potrebné zriadiť novú scénu. Pri zozname scén v hlavnom menu kliknite na dvojšipku a vyberte *ADD NEW*. Novú scénu nazvite *Vitazstvo*, pridajte do nej víťazný nápis (nejaký objekt typu *text*), vypnite režim úpravy, na paneli *Curve and Surface* zvýšte textu hodnotu *Extrude* alebo *Bevel Depth* a klávesovou skratkou *ALT-C* ho zmeňte na *Mesh*, lebo objekt typu *Text* sa (aspoň pri mojich pokusoch) odmieta pri spustení herného stroja zobraziť.



Obrázok 50: Víťazný nápis

Nezabudnite novej scéne nastaviť kameru, svetlá a prostredie sveta. V každom prípade, keď v tejto scéne spustíte herný systém, malo by sa vám ukázať niečo podobné tomu, čo vidíte na obrázku č. 50. Okrem toho, keď už sme v tejto scéne, bolo by vhodné zabezpečiť, aby sa hra pri stlačení ľubovoľnej klávesy dôstojne ukončila. To zabezpečíte tak, ako môžete vidieť na obrázku č. 51. Vyskúšajte, či vám to funguje.



Obrázok 51: Ukončenie hry

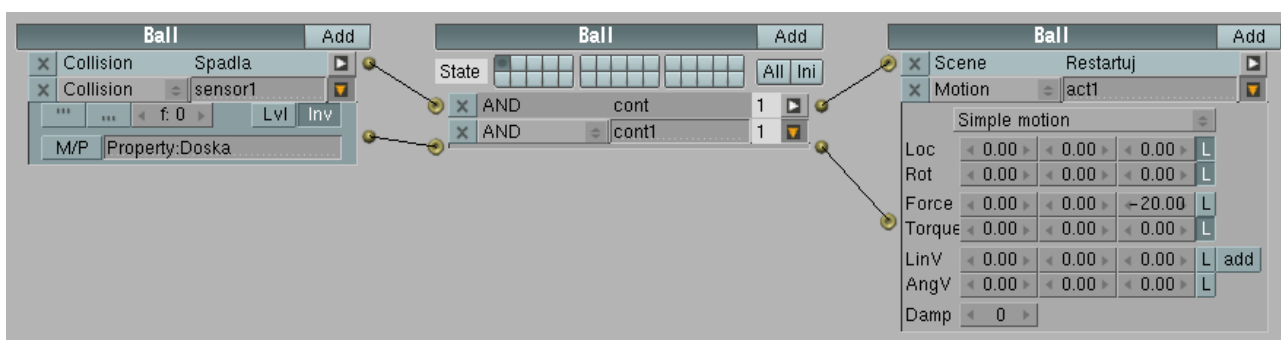
Teraz už len treba zabezpečiť, aby sa hra na túto scénu prepla, keď vychytáte všetky mince. Vráťte sa na pôvodnú scénu, aktivujte objekt *Pocitadlo* a pridajte mu senzor typu *Property*. Tento senzor má na starosti sledovať vlastnosti objektu a ak nejaká vlastnosť dosiahne vhodnú hodnotu, má sa ozvať. Sú štyri typy tohto senzora. *Changed* (zmena) sa aktivuje, keď sa hodnota určenej vlastnosti zmení. *Interval* sa aktivuje, keď sa hodnota vlastnosti dostane medzi hodnoty *Min* a *Max*, ktoré nastavíte. *Not Equal* (nie je rovné) sa aktivuje, keď je hodnota vlastnosti iná, než určíte a *Equal* (rovná sa) sa aktivuje, keď vlastnosť dosiahne stanovenú hodnotu. Ako môžete vidieť na obrázku č. 52, použili sme poslednú možnosť a senzor sa aktivuje, keď počet mincí klesne na nulu. Akcia, ktorá sa vyvolá je typu *Scene* a spraví presne to, čo sa od nej očakáva – nastaví ako aktuálnu scénu s víťazným nadpisom.



Obrázok 52: Prepínanie na inú scénu

Keď sa pokúsite zahrať si túto hru, zistíte, že to celé ešte má istú slabinu. Keď dosku nakloníte, guľa sa skotúľa k zábradliu. Keď sa potom pokúsite dosku vyrovnať, guľa má sklon príliš skákať a pravidelne vám vyletí z hernej plochy. Na jednej strane nie je vždy vhodné hru úplne zľahčiť, na druhej strane ale nie je optimálne ani spraviť ju úplne nehrateľnou a preto by bolo fajn, keby sa prílišnému skákaniu dalo predísť. Za týmto účelom zmanipulujeme fyziku.

Najprv pridajte doske vlastnosť `Doska`, nech ju lopta vie od ostatných objektov rozoznať. Potom aktivujte guľu a nastavte jej, že ak práve nedochádza k žiadnej kolízii s doskou, tak na guľu pôsobí sila, ktorá ju ťahá nadol. Ako senzor použijete `Collision`, overíte, či sa testuje kolízia s objektom, ktorý má vlastnosť `Doska` a stlačíte tlačidlo `Inv`, aby sa senzor aktivoval nie vtedy, keď ku kolízii dochádza, ale vtedy, keď k nej nedochádza. Keďže má guľa zapnuté `Dynamic`, v akcii typu `Motion` máte k dispozícii viac možností, než len `Loc` a `Rot`. Nastavte nejakú zápornú silu v smere osi `z` (zápornú preto, aby guľa padala) a vypnite `L`, lebo potrebujeme, aby guľa padala v smere hlavnej osi `z` a nie v smere momentálneho natočenia jej súkromnej osi `z`. Vyskúšajte, aká veľká sila vám vyhovuje, aby to neskákalo príliš, ale aby si zas hráč nemohol dovoliť úplne čokoľvek.



Obrázok 53: Sila pôsobiaca na guľu

Dokončili sme teda našu malú ukážkovú hru. O hernom systéme by sa dala napísať samostatná knižka, dúfam ale, že tento príklad dal aspoň malý návod, aby ste mohli začať s vlastnými experimentami. Herný systém Blenderu čaká zaujímavá budúcnosť. Už v ďalšej verzii (aktuálna verzia v čase písania tejto kapitoly je 2.47) je prisľúbená podpora výpočtu tieňov v reálnom čase či textúr mapovaných na normály, ak grafická karta počítača podporuje GLSL⁵. A projekt Apricot⁶, ktorý si dal za cieľ dokončiť v hernom systéme Blenderu rozsiahlejšiu hru a popritom systém vylepšiť, dosiahol tiež celkom zaujímavé výsledky.

5 OpenGL Shading Language

6 <http://www.yofrankie.org/>