

3ds Studio Max – Animacija čestica s pFlow i Krakatoa sustavom

3ds Studio Max - Particle Animation with pFlow and Krakatoa System

Željka Ludošan, Bernik Andrija, Vusić Damir

Stručni članak

Sažetak: Čestice su ključni element kod vizualnih efekata i animacije. Koriste se za simulaciju prirodnih pojava poput vatre, dima, vode, oblaka, lišća koje pada, ali i za apstraktne efekte poput magičnih pojava, traga svjetlećih točkica, neobičnih geometrijskih figura itd. Sustav čestica predstavlja upravljački kontejner koji sadrži funkcije koje obavljaju čestice i koje se obavljaju nad njima. Ti sustavi mogu doći u paketu s kompletnim 3d programom, mogu biti kao zaseban program ili kao plug-in. Načini na koji se s njima radi također su različiti, no zasad se najučinkovitijim pokazalo korištenje tzv. čvorova koji se koriste kao objekti spajanja u široj mreži. Novina su također posebni sustavi za renderiranje isključivo čestica. Pošto su čestice zahtjevnije po broju objekata koje koriste, a zahtijevaju veoma malo kod stvari poput svjetlosti, refleksije, broja poligona, mogu se bolje i brže renderirati drugim algoritmima.

Ključne riječi: 3ds Max, 3d čestice, Krakatoa, pFlow, renderiranje, simulacija

Professional paper

Abstract: Particles are a key element when it comes to visual effects and animation. They are used for simulating occurrences in nature such as fire, smoke, clouds, falling leaves, but are also used for abstract effects such as magical occurrences, a trail of glowing dots, unusual geometrical shapes, etc. Particle system represents a control container that holds functions that are performed by and carried on the particles. Those systems can come in a package alongside a complete 3d software, they can be a stand-alone program, or a plug-in. There are also different ways to work with particles but so far the most efficient way seems to be by using the so called nodes that are used as objects of connection in a broader network. Special systems for rendering particles exclusively are also a novelty. Since particles are more demanding on the draw count used but require a lot less with things like lighting, reflection, polygon count, they can be rendered in a better and faster way with other algorithms.

Key words: 3ds Max, 3d particles, Krakatoa, pFlow, rendering, simulation

1. UVOD

U fizici su čestice mali lokalizirani objekti kojima se mogu pridodati fizička svojstva poput volumena ili mase. [1][2] Može li se neki objekt smatrati česticom ovisi o veličini konteksta, odnosno ako je veličina objekta veoma malena ili toliko malena da nije bitna ili ako su geometrijska struktura i svojstva nebitna, onda se taj objekt može zvati česticom.[3]

Čestice su često prikazane kao točke u prostoru koje imaju smjer. U računalnoj grafici termin "sustav čestica" referira se na tehniku koja koristi velik broj malih spriteova ili drugih grafičkih objekata kako bi se simulirali razni fenomeni koje bi inače bilo teško reproducirati konvencionalnim render tehnikama.[3]

Spriteovi su dvodimenzionalne sličice ili animacije koje se poželjno nalaze u formatu malih dimenzija.

Obično se tom tehnikom, sustavom čestica, rade kaotični sustavi, prirodni fenomeni ili procesi nastali kemijskim reakcijama.

Da biste napravili dobru animaciju s česticama potreban vam je 3d program koji u sebi ima integriran

alat za izradu sustava čestica i njihovu manipulaciju, te engine za renderiranje. Nakon što se dobiju sirove slike one se mogu još dodatno obraditi u postprodukciji s programom za obradu videomaterijala.

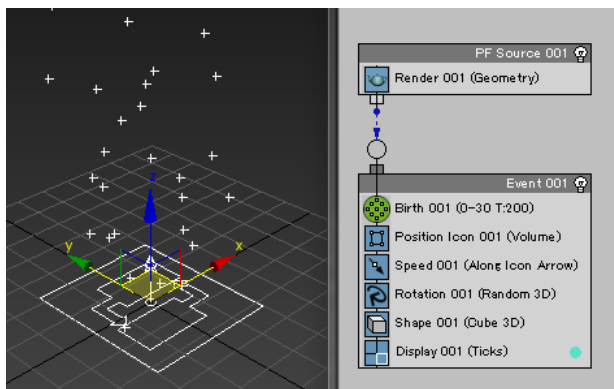
U ovom radu bit će korišten 3d program zvan Autodesk 3ds Max koji u sebi ima integriran pFlow za manipulaciju s česticama. Za renderiranje će se koristiti suvremeni engine Krakatoa koji je izrađen posebno za renderiranje čestica.

Budući da je glavna odlika ovog rada produkcijska kvaliteta koja je zaslužna uglavnom zbog Krakatoa sustava za renderiranje čestica, u radu će biti govora o renderiranju čestica preko Krakatoa. Ovaj način renderiranja koji se specijalizirao za renderiranje isključivo čestica još je nov u današnje vrijeme i zbog toga nema puno literature koja se time bavi.

Adobe Soundbooth će se koristiti kako bi se obradila glazba i promijenio format kojeg podržava 3ds Studio Max, a s Adobe After Effectom će se renderirane sličice spojiti u jedan komad i napraviti postproduksijski filter koji će zaokružiti cjelokupni projekt.

2. PFLOW

PF Source (slika 1.) je izvor izviranja čestica i dolazi sa svojim početnim postavkama.



Slika 1. Izvor čestica unutar preglednika s početnim postavkama

Postavke nekog sustava čestica su razvrstane u tzv. čvorove. Čvor (eng. Node) je vezna točka koja se ili nadovezuje na drugu točku ili je prekidna točka. Input je ulazna točka u koju se ubacuju podaci, a output je izlazna točka iz koje se modificirani podaci izbacuju i šalju u drugi čvor. Jedno izvorište može imati više setova događaja. Događaj je pokret unutar sustava koji radi promjene na podacima.

2.1. Funkcije pojedinih čvorova koji se češće koriste

Render je osnovni čvor koji je zaslužan za prikazivanje čestica. Postoje četiri tipa rendera: None, Bounding Boxes, Geometry i Phantom.

Visible % je opcija kojom se može u postocima smanjiti količina čestica koja će se prikazati tijekom renderiranja. Render result (ishod renderiranja) su opcije koje određuju kako će render program gledati na konačne rezultate.

Birth je osnovni čvor koji definira stvaranje čestica.

Position Icon je čvor koji kontrolira otkuda će se čestice početi stvarati. Lokacija je obavezno ovisna o pFlow ikoni ili o drugom objektu za čiju se površinu/vertex/ rub ili volumen lijepi.

Speed je osnovni čvor koji određuje brzinu i smjer kretanja čestica.

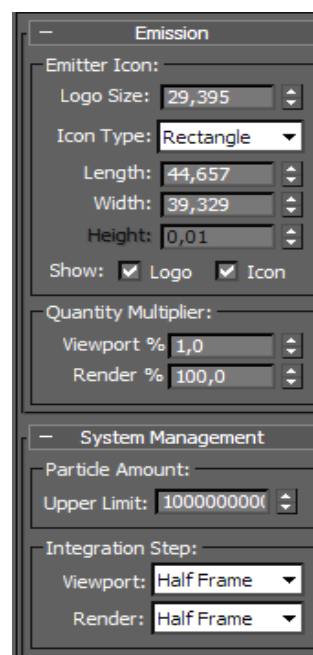
Rotation je osnovni čvor koji kontrolira okretanje čestica oko svoje osi.

Shape je osnovni čvor koji kontrolira oblik čestica unutar renderiranog ekrana. Čestice mogu poprimiti 2d ili 3d oblik po želji.

Display je čvor koji prikazuje čestice na ekranu. Pod Type se može podesiti kako će se čestice prikazivati u viewportu, Visible % opisuje koliko su čestice transparentne, a pod Selected se može odabrati oblik označenih čestica. Boju čestica možemo mijenjati uz pomoć kockice s desne strane.

2.2. Optimizacija prikazivanja čestica

Kako bi renderirane čestice imale što veću produkcijsku kvalitetu potrebno je imati što više čestica. Iako Krakatoa podržava renderiranje velikog broja čestica, 3ds Maxov viewport ne može toliko informacija podnijeti, pogotovo zato što mora kroz te informacije prolaziti “real-time”. Iz tog razloga postoji opcija unutar pFlow izvora pod nazivom “Viewport %” (slika 2.), gdje možemo kontrolirati koliki postotak od ukupnog broj čestica u izvoru će se prikazivati i viewportu. Kada se radi s velikim brojem čestica, taj broj je pametno držati između 0,1 i 1,0.



Slika 2. PF Izvor modifikator

Isto tako je važno napomenuti da se Particle Amount: Upper Limit (gornja granica količine čestica) mora povećati za nekoliko nula, jer kada se radi s Krakatoa-om broj čestica koje koristimo zna biti veći od nekoliko milijuna.

2.3. Automatizirano upravljanje česticama

Da bi se čestice kretale prema glazbi, umjesto da se ih se namješta ručno može se iskoristiti i “Curve Editor”. Jednostavno se odabere atribut čije se vrijednosti žele mijenjati kao npr. Birth Rate (brzina stvaranja), zatim se ide na Controller>Assign>AudioFloat i doda se željeni .wav dokumenat. U Controller Range upisuje se najmanja i najveća vrijednost atributa koja se podudara s minimalnom i maksimalnom vrijednošću decibela odabranog zvuka.

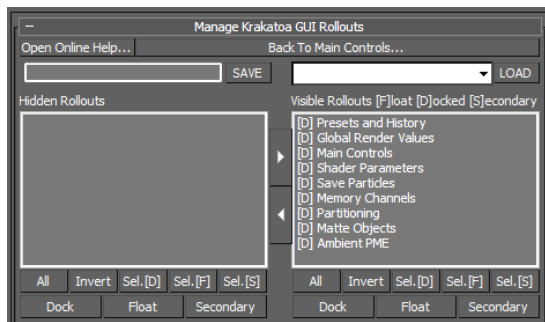
Tako će se vrijednost atributa mijenjati kako se mijenja i glasnoća zvuka.

Drugim riječima, u ovom slučaju čestice će se stvarati u skladu s glazbom. Pošto nema velike kontrole nad postavkama, najbolje je ipak raditi ručno.

3. KRAKATOA

Za početak već treba imati postavljenu kameru i svjetlo koje je naročito važno da bi se čestice vidjele prilikom renderiranja.

3.1. Važniji elementi sučelja

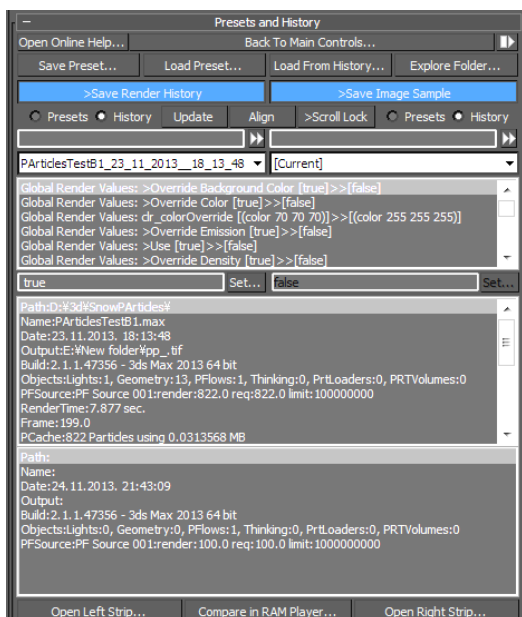


Slika 3. Postavljanje grafičkog sučelja za Krakatou

Kratice [D][F][S] označuju na koji način se otvara svaki od tabova. [D] je za "Docked", odnosno taj tab će se nadovezati kao dio ovog glavnog sučelja. [F] je za "Floated", što znači da će taj tab lebdjeti sa strane kao poseban prozor i [S] je za "Secondary" i to su tabovi nadovezani na [F] sučelje.

"Hidden Rollouts" su tabovi koji su vidljivi za razliku od "Visible Rollouts" koji su vidljivi i možemo ih nesmetano koristiti.

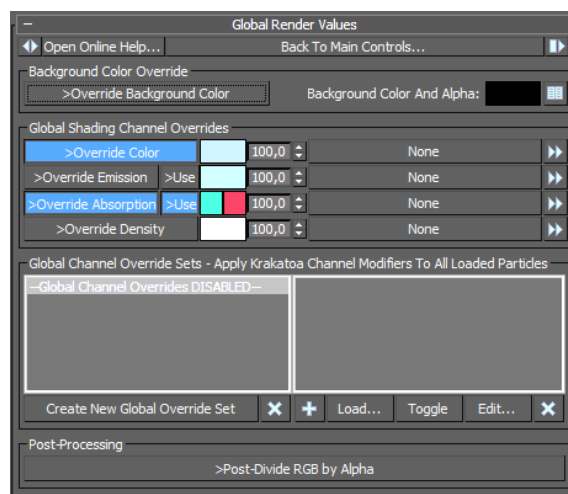
Presets and History tab je tab koji se bavi spremanjem postavki za renderiranje. Ako ste već radili nekakav projekt i želite si postavke spremi, ovo je mjesto za to. Uz to ima opciju da postavke sprema automatski svaki put kada izađete iz programa. Postavke su imenovane po danima i ne brišu se same od sebe. Zbog toga je potrebno paziti i postavke čistiti s vremena na vrijeme jer mogu zauzimati mnogo prostora na disku, ili ih možete jednostavno isključiti i sami birati kada želite da vam se postavke sprema (slika 4.).



Slika 4. Spremanje postavki unutar Krakatoa sučelja

Dva kvadrata u donjem dijelu prikazuju zadnje renderirane slike koje su napravljene s tim postavkama.

Global Render Values je tab u kojem možemo podešavati neke općenite postavke čestica vezane za boju, vidljivost i gustoću (slika 5.).



Slika 5. Glavne globalne postavke

Pozadina renderirane slike prema početnim postavkama je podešena unutar Environment (okolina) postavkama, što je jedan od tabova unutar 3ds Max nezvan za render sustav. Preko tih postavki možemo prijeći preko, ako imamo pritisnuto "Override Background Color" i tada s desne strane možemo odabrati boju pozadine.

"Override Color" koristimo kada želimo da se čestice renderiraju s određenom bojom ili teksturom bez obzira na to kako su postavljene unutar sustava čestica ili nekih drugih programa koji se miješaju u Krakatoa postavke. S desne strane u kućici promijenimo boju, broj od 0 do 100 određuje koliko se jako mapa koju miješamo s bojom vidi i samu mapu možemo dodati odmah pokraj toga. Mapa vuče link iz Material Editora.

"Override Emission" koristimo kada želimo da nam čestice odsijavaju svoju vlastitu svjetlost pa ih tako možemo vidjeti i bez pravog svjetla u sceni.

"Override Absorption" koristimo kada imamo svjetlo u sceni. Svako svjetlo ima boju: bijelu, žutu ili neku nijansu plave. Kada svjetlo prolazi kroz čestice, tada su čestice pod utjecajem svjetla i do određene mjere poprimaju njegovu boju. Ono što Override Absorption radi jest da uzme čestice odabrane boje i tu boju "apsorbira", odnosno čestice koje se bile te boje sada su obojene u boji koja je komplementarna originalnoj. Zato ovdje postoje dvije kućice s bojom. Kada mijenjate jednu automatski se mijenja i druga koja pokazuje koje će boje biti čestice. To je korisno kada želite da su vam čestice drukčije boje od svjetla koje na njih pada, i da to možete postići da ne mijenjate postavke svjetla koje vam možda treba da bude određene boje na drugom mjestu u sceni.

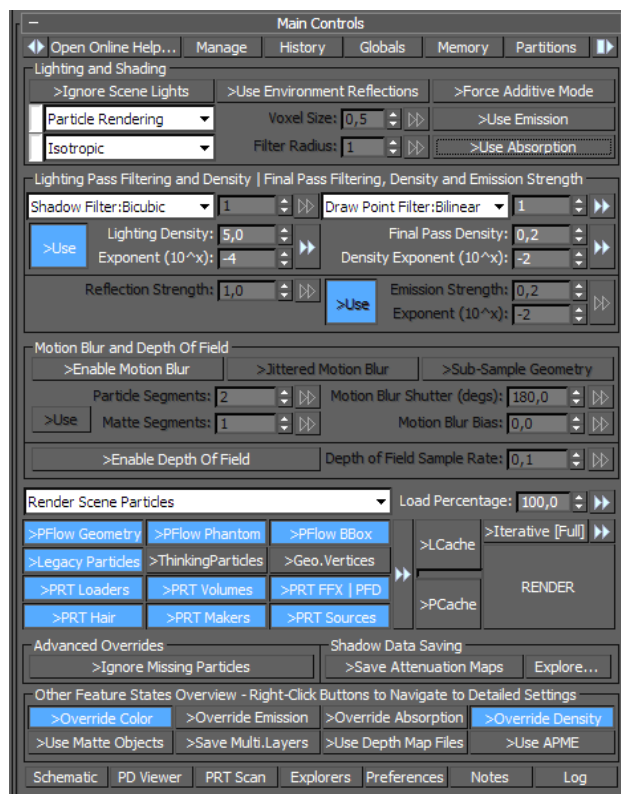
"Override Density" koristimo kada želimo postaviti globalnu gustoću česticama. Naime, što ima više čestica to se kontura čestičnog sustava bolje vidi, ali to nije jedini način kako se čestice mogu bolje vidjeti.

Gustoća čestica u Krakatoa-i nije nešto što se može opisati brojem već bojom.

To je ona boja koju nosi najjače osvijetljena čestica na sceni. Ako je npr. Override Density podešen na bijelo, to znači da će sve čestice izgledati bjelje i jasnije nego da je podešen na tamnosivo. Isto tako, ako bi Override Density podesili na crno, to znači da je najsvjetlija čestica crne boje, a sve ostale čestice mogu biti samo još tamnije tako da se na renderu ne bi vidjelo ništa.

“Global Channel Override Set” tab služi tome da postavke koje ste kreirali, vezane uz globalne vrijednosti postavki, možete spremirati za buduću referencu.

“Post-Divide RGB by Alpha” je opcija kod koje se, kad je uključena, renderira samo alpha kanal koji ne razlikuje gustoću tako da je za njega svaka čestica potpuno bijele boje.



Slika 6. Glavne postavke za renderiranje

“Main Controls”– ovdje se nalaze glavne kontrole za osvijetljenje i renderiranje čestica. Odmah ispod naslova nalaze se prečaci koji vode na druge tabove poput “Manage”, “History”, “Globals”, “Memory”, “Partitions” itd.

“Ignore Scene Light” je postavka koja ignorira svjetla koja su postavljena zasebno u sceni i koristi samo početno svjetlo koje se automatski isključuje u 3ds Maxu, ako u scenu ubacite posebna svjetla.

“Use Environment Reflections” je postavka koja uključuje globalnu sliku ili boju koja se koristi za refleksije i obično je postavljena u Maxovom Environment tabu. Takvim slikama pripada npr. vrlo poznati HDRI (High Dynamic Range Image) format. Ako je Environment Reflections uključen, to znači da će materijal na česticama koristiti refleksije.

“Force Additive Mode” je postavka koja forsira aditivno renderiranje tako da unutar programa kopira kanal boje u emisijski kanal i pritom boju i apsorpciju podesi na crno. Time ćemo dobiti potpuno osvijetljene

čestice koje svijetle više na mjestima na kojima se preklapaju iz određenog kuta.

Postoje dvije glavne vrste renderiranja: Particle Rendering i Voxel Rendering.

“Particle Rendering” je vrsta renderiranja gdje je svaka čestica renderirana kao jedna pixelna točka.

“Voxel Rendering” je vrsta renderiranja gdje je svaka čestica postavljena na voksalnu mrežu i na kraju se ta mreža osjenča.

Ako je uključeno Voxel Rendering tada možemo podesiti Voxel size i Filter Radius.

“Voxel size” kontrolira rezolucije mreže, a “Filter Radius” kontrolira koliko će se voksalnih točaka moći preklapati (filtrirati) kad će se mreža osjenčavati. Ako je postavka postavljena na 1, tada se ništa neće preklapati.

I za Particle Rendering i za Voxel Rendering mogu se postaviti fazne funkcije koje se koriste za izračunavanje disperzije svjetlosti kod čestica.

Razlikujemo šest različitih faznih funkcija: Isotropic ili izotropna funkcija kaže da će svjetlo biti uniformno raspršeno u sve smjerove; “Phong Surface” kaže da će se zrake svjetla raspršivati prema Phong modelu za sjenčanje koji koristi Normal kanal; Henyey-Greenstein kaže da će se zrake svjetla raspršivati prema kutu koji se nalazi između svjetla i kamere i uvažavat će postavke iz Eccentricity kanala koji služi kako bi se ubacilo malo neuravnoteženosti; Schlick je isto kao i Henyey-Greenstein, samo aproksimirana verzija.

$$p(\theta) = \frac{1}{4\pi} \frac{1-g^2}{(1+g^2-2g \cos \theta)^{3/2}}$$

$$\int_0^\pi p(\theta) 2\pi \sin \theta d\theta = 1 \quad (4)$$

$$\int_0^\pi p(\theta) \cos \theta 2\pi \sin \theta d\theta = g$$

$$R(\theta) = R_0 + (1 - R_0)(1 - \cos \theta)^5 \quad (5)$$

Kajiya-Kay i Marschner su bazirani na tangentama za dlake i kosu.

“Lightning Pass Filtering and Density|Final Pass Filtering, Density and Emission Strength”, odnosno gustoća i filtracija propusnosti svjetlosti. Konačno filtriranje svjetlosne propusnosti, gustoća i snaga emisije su dva seta postavki koji većinski odlučuju o izgledu čestica.

Postoje dvije vrste finijih kontrola za gustoću čestica, a to su “Lightning Pass Density” koji izračunava mape za prigušivanje što se koriste kod osjenčavanja čestica, i “Final Pass Density” koji iscertava čestice direktno na ekran.

Postoje dvije glavne vrste filtera, a to su “Shadow Filter” koji koristi informacije iz prigušnih mapa, i “Draw Point Filter” koji odlučuje kako se čestice renderiraju kada više njih padne u isti pixel.

Shadow Filter – “Nearest” (najbliži) je korištenje vrijednosti najbližeg pixela

“Bilinear” koristi linearnu jednadžbu kako bi rekonstruirao vrijednost iz četiri obližnja pixela.

“Bicubic” koristi kubičnu funkciju filtracije da bi se izračunao prosjek obližnjih pixela.

Draw Point Filter – “Nearest” (najbliži) je kada se puna vrijednost pixela stavi u obližnji pixel na izlaznoj slici

“Bilinear” koristi linearnu jednadžbu da bi prepolovio vrijednost čestice i ubacio u čestice koje su pod utjecajem na izlaznoj slici.

“Bicubic” koristi kubičnu funkciju filtracije kako bi se izračunao prosjek obližnjih pixela na izlaznoj slici.

“Final Pass Density” je jedna od postavki koje su potrebne da se odredi renderirana gustoća za svaku česticu. Ova vrijednost je zapravo koeficijent gustoće koji je izražen znanstvenom formulom. Ako su vrijednosti manje tada se čestice čine manje gustima, dok povećana vrijednost diže gustoću.

“Density Exponent (10^x)” je druga postavka za određivanje gustoće. Ovo je vrijednost eksponente na desetu.

Npr., da bi se dobila gustoća od vrijednosti 0.000005, to se može napisati kao 5.0×10^{-6} , odnosno u Krakatoa-i je to Density Per Particle: 5.0, Density Exponent: -6. Vrijednost 27500000 bi bila 2.75×10^7 , odnosno Density Per Particle: 2.75, Density Exponent: 7.

“Reflection Strength”, odnosno snaga refleksije kontrolira jakost refleksije koja nastaje iz refleksivnih mapi. Vrijednost 1 znači da je refleksija kao i original, vrijednost 0 znači da se refleksija ne vidi, sve vrijednosti između lagano otkrivaju refleksiju i ako je vrijednost veća od jedan tada će cijela refleksija biti bijela.

“Emission Strength”, odnosno snaga emisije ili isijavanja definira se koeficijentom i eksponentom. Tako mijenjanje postavki iz 1.0/-1 u 1.0/-3 je zapravo mijenjanje vrijednosti iz 0.1 u 0.001.[6]

4. IZRADA PROJEKTA

4.1. Find Target i put kretanja

Prvo se napravi Find Target objekt koji će čestice pratiti. Taj se Find Target treba kretati po putanji koja se kreira pomoću “Splines”. Koristeći postavke Initial Type: Smooth i Drag Type: Bezier nacrtate putanju na sceni, zatim uđete u vertex mod, nasumično odaberete nekoliko točaka i malo ih podignete ili snizite po želji tako da vam putanja bude raznolika po svim osima.

Find Target objekt se sada treba privezati na putanju. To ćemo napraviti tako da označimo Find Target i uključimo Path Constraint. Find Target će se automatski premjestiti na početak putanje, kretat će se konstantnom brzinom sve do kraja putanje i tamo će završiti na zadnjem frameu u sceni.

4.2. Ubacivanje zvuka u 3ds Max

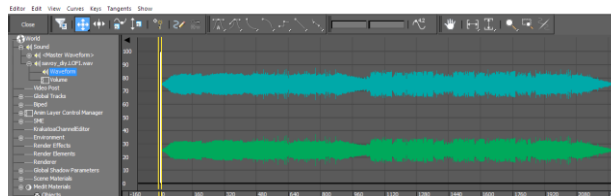
Kako bi mogli animirati kretanje čestica potrebno je ubaciti glazbu u 3ds max. Zvuk koji se ubacuje u 3ds Max mora biti .wav formata, a ubacuje ga se preko Mini Curve Editora.

Koristeći Auto Key opciju koja sprema promjene na svakom frameu na kojem ste promjene napravili, možete animirati kretanje Find Target objekta po putanji,

istodobno prateći glazbu koja se pokreće ovisno o tome kako se kreće po timelineu.

Sve dodatne promjene vezane uz kretanje Find Target objekta na putanji možete promijeniti u Motion postavkama. Bitno je da vam se putanja nalazi pod Target. Ovdje također možete dodati još više putanji.

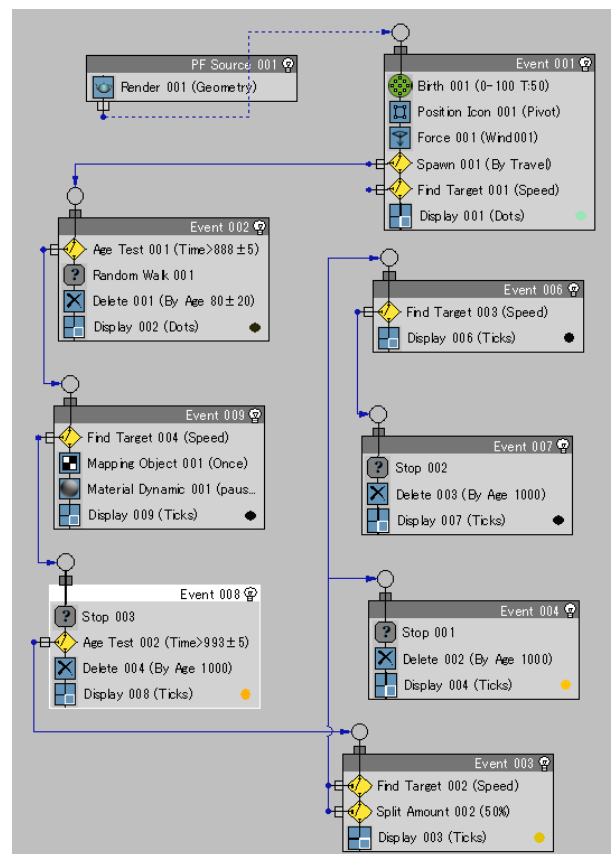
Ako se ne možete odlučiti koju ćete koristiti, ostavljate to za kasnije.



Slika 7. Editor krivulja s ubačenim zvukom

Da bi se Find Target kretao prema glazbi morate animirati % Along Path. “% Along Path” je postavka koja prikazuje na koliko posto od ukupne putanje se nalazi naš objekt. Crveni trokutići po rubovima označuju da je ta postavka animirana.

4.3. Postavke pFlow izvora



Slika 8. pFlow izvor unutar preglednika

Iz izvora se u periodu od 100 frameova stvori 50 čestica. Te čestice izvire iz jedne točke koja je u ovom slučaju pivot izvora.

Na te čestice djeluje sila vjetrova koja nema nikakvu vrijednost snage već samo turbulenciju, frekvenciju i veličinu koje tim česticama mijenjaju smjer gibanja po malo kaotičnom rasporedu. Iz svake od tih 50 početnih

čestica odmah na početku počinje se stvarati novi trag čestica gdje se 19 novih čestica stvara unutar 0,5 unita (3ds Maxova jedinica) iza one prethodne čestice s varijacijom od 5%.

Štoviše, 10-tak tih čestica poprima smjer i gibanje od čestica iz kojih su proizašle s varijacijom od 2% i odstupanjem od 5 unita, što znači da je dio čestica skladan i kreće se u istom smjeru, dok se jedan manji dio njih raspršuje sa strane. Čestice zatim, s tim postavkama, počinju slijediti Find Target objekt koji se kreće po sceni brzinom od 100 unita i varijacijom od 50%. To znači da čestice mogu poprimiti brzinu vrijednosti od 50 do 100 i kada se 100% približi objektu tada stanu sve dok se objekt ponovo ne udalji. Čestice koje se približe Find Target objektu na vrijednost manju od 1 šalju se u novi događaj i koriste Random Walk koji ih raspršuje u određenom obliku prema postavkama ovisnim o brzini, šumu i viskoznosti. U Eventu 002 čestice počinjemo dijeliti. Kada čestice dođu do framea 889 s varijacijom vrijednosti 5, što znači da se počinje dijeliti u periodu od 884-tog do 894-tog framea, čestice se šalju u sljedeći događaj koji se u ovom primjeru zove Event 009. Također, čestice ne traju vječno već imaju životni rok prema starosti, odnosno svaka čestica umire nakon 80 frameova, s varijacijom od 20, nakon što se rodila. Čestice se zatim nastavljaju kretati prema Find Target objektu i svaka čestica poprima neku vrijednost na koordinati, odnosno one se mapiraju i pridodaje im se dinamički materijal.

Dinamički materijal za razliku od statičkog može animirati svoje atribute. Na frameu 994 s varijacijom od 5 čestice se šalju u sljedeći događaj dok dosadašnje počinju umirati točno 1000 frameova nakon što su se rodile. U novom događaju Event 003 čestice nastavljaju slijediti Find Target objekt i količinski se podijele na dva dijela. 50% čestica kreću u novi događaj Event 006, dok se drugih 50% čestica počinje zaustavljati i umirati nakon 1000 frameova. Ovih prvih 50% čestica u događaju Event 006 počinje ponovo slijediti Find Target objekt, kada dođe do njega zaustavljati se i umirati nakon 1000 frameova.

4.4. Cacheiranje čestice preko Krakatoa sučelja

Ako i smanjimo broj čestice koje se prikazuju u viewportu, velika je vjerojatnost da će nam scena usporavati i zbog toga čestice moramo cacheirati. Drugi razlog tome je što ćemo boju čestica na prvom dijelu animacije mijenjati pomoću Krakatoinog Magma modifikatora koji radi samo na cacheiranim česticama.

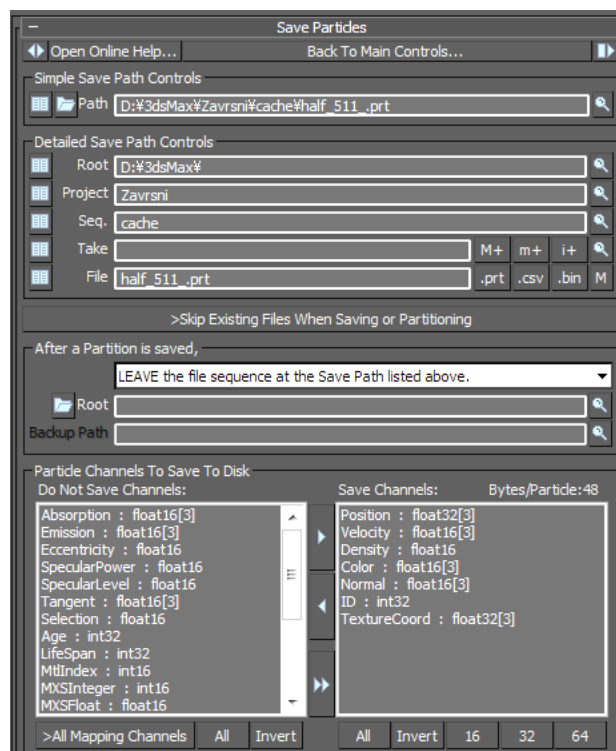
Kanali koji obavezno moraju biti cacheirani su: Position (pozicija), Velocity (vektor brzine), Color (boja), ID (identifikator) i TextureCoord (koordinate na teksturi).

Pod "Path", odnosno putanjom upisujemo gdje će nam se čestice spremati nakon cacheiranja i kako će se one zvati. Ispod toga program automatski ispunji kućice s informacijama kao što su ime projekta, gdje se projekt nalazi, ime sekvence i kako će se čestice automatski imenovati (slika 9.).

Ako su se čestice na nekim frameovima već prije cacheirale i ne želite da se te čestice cacheiraju ponovo,

možete uključiti "Skip Existing Files When Saving or Partitioning", što znači "Preskoči postojeće datoteke kada se sprema ili radi potpodjela". U sredini možete podesiti što će se dogoditi s datotekama nakon što se čestice cacheiraju. Postoje tri mogućnosti: da se datoteke ostave tamo gdje su spremljene, da se kopiraju u pričuvni direktorij ili da se premjeste u pričuvni direktorij.

Ovo može biti korisno ako želite da se datoteke automatski s vašeg računala kopiraju na server tako da njima mogu pristupiti i drugi.



Slika 9. Sustav za pohranjivanje čestica

Pri dnu se nalaze dvije sive kućice s raznim kanalima spremnim za uključivanje ili isključivanje iz cacheiranja. U lijevoj kućici nalaze se kanali koji se neće spremati u datoteke od čestica prilikom cacheiranja, a s desne strane su oni koji će se spremati. Svaki kanal zauzima određeni broj bajtova u svakoj .prt datoteci kojih ima onoliko koliko ima i frameova. Zbog toga se prati jesu li kanali zauzimati mjesta. Kako se ne bilo dogodilo da na računalu nema dovoljno prostora, uključuju se samo oni kanali koji su potrebni.

Iako 48 bajtova možda ne izgleda mnogo, ali kada se to pomnoži s brojem čestica kojih je oko 500 000, koje kad se umnože zbog potpodjele, to je 500 000 x 10 x 48 = 240 000 000 bajtova po jednom frameu. Ako animacija traje npr. 5 sekundi, to je 30 frameova po sekundi, a sve zajedno 150 frameova x 240 000 000 bajtova = 3.6x10¹⁰ bajtova, što je oko 28 gigabajta - to je 28 gigabajta za samo 5 sekundi animacije.

4.5. Korištenje potpodjele preko Krakatoe

Ponekad nam maksimalan broj čestica koji možemo koristiti unutar 3d programa u realnom vremenu nije dovoljan i želimo ga povećati da bi dobili što preciznije

rezultate. Jedan od načina je korištenje “Partitioning” ili potpodjela čestica. Potpodjela čestica funkcionira tako da cacheira čestice više puta, a svaki put kada cacheira čestice na istom frameu malo im pomakne poziciju.

Npr., ako u jedan frame uspijete ubaciti 50 000 čestica, ako optimizirate broj vidljivih čestica u viewportu možete imati 500 000 čestica. Ako to cacheirate s potpodjelom možete imati i do 500 000 x 10 = 5 000 000 čestica.

“Partition Count”, odnosno broj potpodjela označuje koliko će se puta svaki frame duplicirati.

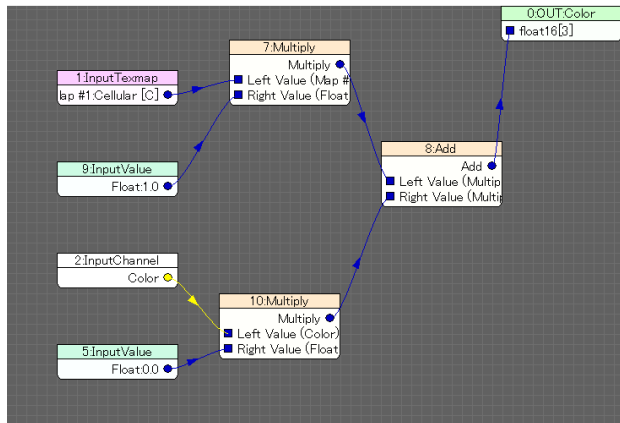
Maksimalni broj može biti 10, što znači da će se za jedan frame spremati 10 različitih datoteka. Odmah pokraj nalaze se opcije za postavljanje raspona potpodjele. Kako cacheiranje zna potrajati, može se odrediti koji dio potpodjele će se cacheirati.

Npr., ako nam je broj potpodjela 10 i ako u raspon upišemo 5 do 8 tada će se od 10 potpodjela spremati samo 4 i to peta, šesta, sedma i osma po redu. Ispod toga biramo što će se mijenjati u svakoj potpodjeli. Sve što je plavo će se mijenjati i može se odabrati da se čestice mijenjaju samo kod označenih objekti. Potpodjela se može izvršiti lokalno na računalo ili na nekoliko računala, što onda koristi tzv. Deadline dio programa koji se mora posebno kupiti. Lokalno se može cacheirati sve potpodjele ili samo one koje su unutar raspona.

“Skip Existing Files When Saving”, ako je označeno, dopušta da se datoteke koje su već spremljene ne spremaju ponovo, nego ih se samo preskoči.

4.6. Mijenjanje boje i oblika čestica nakon pohrane podataka

Nakon što se čestice cacheiraju, izvor se može maknuti jer sada čestice izviru iz objekta zvanog “PRT Loader”. Na njega možemo nadovezati neke modifikatore poput “Noise” (šum) koji uzima oblik čestica i raspršuje ga do određenih granica, te “Twist” (savijanje) koji oblik čestica savija po određenoj osi do određenih granica. Šum i savijanje su ovdje oboje bili animirani. Kako bi se zraka čestica bližila kraju kretanja, odjednom bi se raspršila i zatim bi iskoristila savijanje da izgleda kao da se čestice vrte velikom brzinom oko svoje osi.



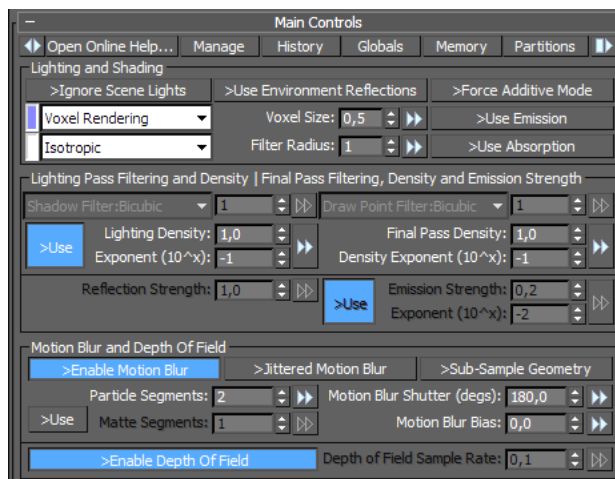
Slika 10. Sustav za mijenjanje boje unutar Magma Flow

Najvažniji modifikator koji je ovdje iskorišten je Magma modifikator koji dolazi kao dio Krakatoa.

Pomoću Magma modifikatora dodana je tekstura na čestice i animirao se lagani prijelaz između boje koje proizlazi iz Magme i one koja je originalno bila u samom pFlow izvoru čestica. Originalne boje čestica iz izvora uspjele su se zadržati zato što smo čestice pohranili s TextureCoord i Color kanalom.

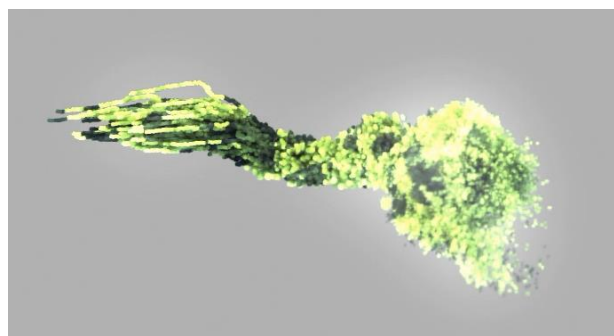
4.7. Postavke renderiranja

Koristi se voksalno renderiranje da bi čestice dobile na popunjenosti i malo čupav i pahuljast izgled. Motion Blur (zamućivanje prilikom kretanja) dodaje realističnost, jer kako se čestice kreću većom brzinom to za sobom ostavljaju veći trag zamućenosti. Uključen je Depth of Field (dubina polja) koji je vezan za kameru koja na sebi ima Krakatoa kameru modifikator. Kamera koristi sferične leće, f-stop od vrijednosti 2 i fokalnu dubinu vrijednosti 500. To su dovoljne postavke da bi se postigla zamućenost čestica, ovisna o udaljenosti od kamere (slika 11.).



Slika 11. Postavke za renderiranje korištene u projektu

Rezultat rada je vidljiv na sljedećoj slici:



Slika 12. Primjer konačnog rezultata simulacije

5. ZAKLJUČAK

Danas je u filmskoj i VFX industriji kvaliteta jako bitna i ta kvaliteta je veoma ovisna o tome koliko nešto izgleda realistično ili koliko nam se nešto čini kao nešto što bi moglo postojati. S takvim razmišljanjem treba raditi i s česticama. Jesu li to bila realistična simulacija nekog fizičkog događaja ili apstraktna animacija koja je

tu da dekorira i zabavlja, rad će se više doživjeti ako je sličan onome što doživljavamo u stvarnom životu. Zbog toga su stvari poput DoFa, Motion Blura i svjetlosti potrebne i moraju sudjelovati u izradi efekata.

Krakatoa sustav za renderiranje čestice je još vrlo nov sustav, iako već korišten u mnogo filmova i igara zbog trenutačne popularnosti. U kombinaciji s moćnim alatima poput pFlowa mogu se raditi animacije vrhunske produkcijske kvalitete na prosječnom PC-u uz malo strpljenja i puno znanja i prakse.

3ds Max nažalost nema mnogo mogućnosti što se tiče zvuka, ali uz Curve Editor ili Displacement mape možete postići zanimljive rezultate gdje će vam se čestice kretati u ritmu glazbe.

6. PRILOG

Primjer postavki od izvora čestica:

PF Source 001

Render 001 (Geometry)

Type: Geometry

Visible %: 100,0

Render Result: Single Mesh

Event 001

Birth 001 (0-100 T:50)

Emit Start: 0

Emit Stop: 100

Amount: 50

Subframe Sampling +

Position Icon 001 (Pivot)

Lock On Emitter –

Inherit Emitter Movement -Location:

Pivot

Force 001 (Wind001)

Force Space Wraps: Wind001

Influence%: 1000,0

Sync By: Absolute

Spawn 001 (By Travel)

Spawn Rate And Amount: By Travel

Distance: Step Size: 0.5

Spawnable %: 100,0

Offspring #: 19

Variation %: 5,0

Sync By: Absolute Time

Restart Particle Age +

Speed: Inherited: 10

Variation %: 2,0

Divergence: 5,0

Size: Scale Factor %: 100,0

Variation %: 0,0

Find Target 001 (Speed)

Control By Speed +

Test True If Distance To: Target Point Is Less

Than 1,0

Control By Speed : Use Cruise Speed +

Speed: 100,0

Variation: 50,0

Accel Limit: 8000,0

Ease In %: 100,0

Sync By: Absolute Time

Target: Icon

Sync By: Absolute Time

Follow Target Animation +

Point: Random

Docking Direction: Type: Icon Spherical

Distance: 10,0

Icon Size: 30,0

Color Coordinate +

Display 001 (Dots)

Type: Dots

Visible %: 100,0

Selected: Ticks

Kraj prvog eventa (Event 001)

7. LITERATURA

- [1] Particles AMS Glossary (American Meteorological Society), srpanj 2012.
- [2] http://www.bbc.co.uk/bitesize/ks3/science/chemical_material_behaviour/particle_model/activity, srpanj 2012.
- [3] <http://docs.autodesk.com/3DSMAX/16/ENU/3ds-Max-Help>, srpanj 2012.
- [4] <http://omlc.org.edu/classroom/ece532/class3/hg.html>, kolovoz 2012.
- [5] http://en.wikipedia.org/wiki/Schlick%27s_approximation, kolovoz 2012.
- [6] <http://www.thinkboxsoftware.com/krak-main-controls/>, rujan 2012.

Kontakt mentora:

Andrija Bernik, dipl.inf., predavač

Veleučilište u Varaždinu

J. Križanića 33, HR-42000 Varaždin

bernik.velv@gmail.com