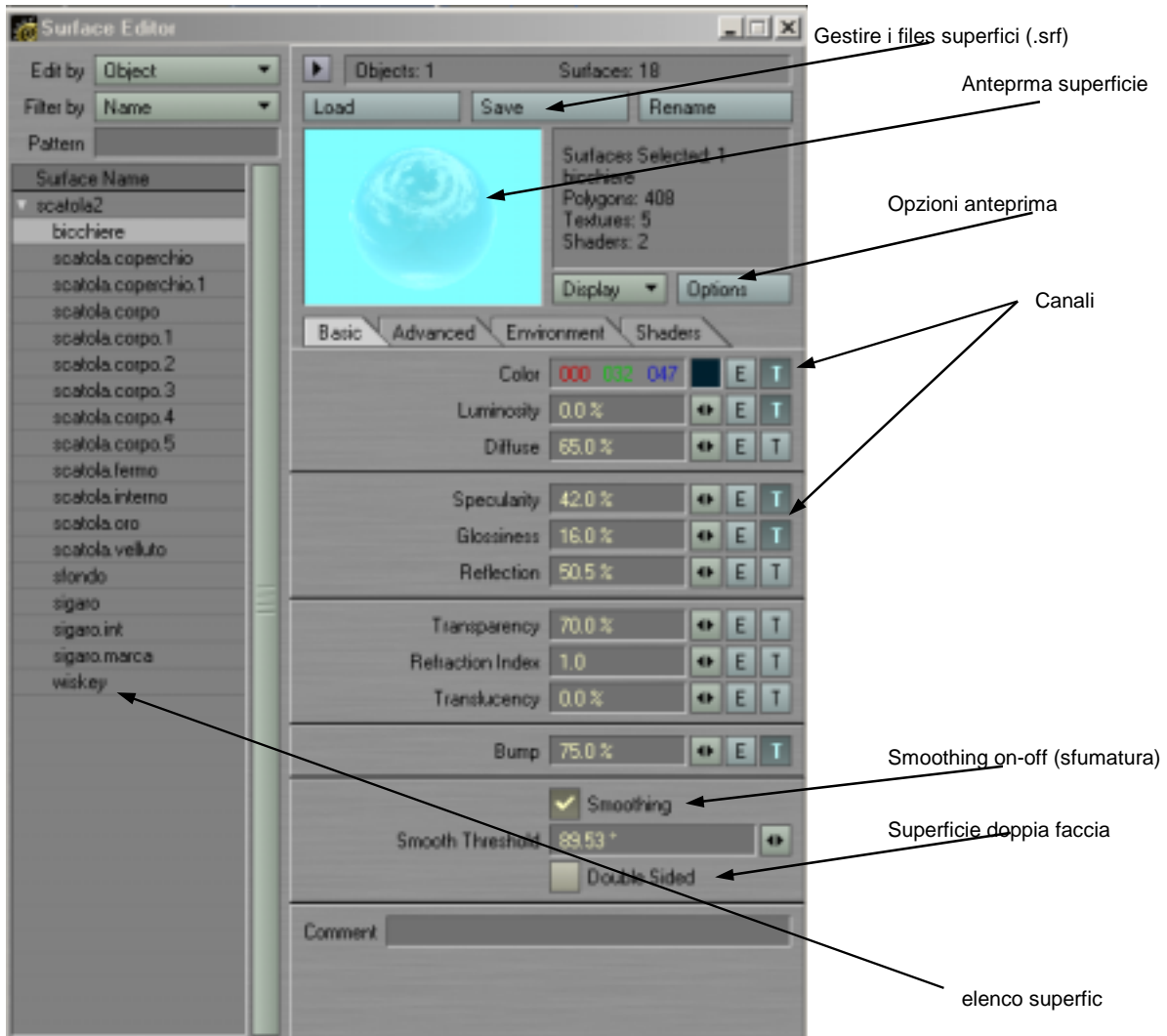


Le superfici

Iniziamo a vedere più da vicino i pannelli non modali del programma. Ricordo che il pannello non modale è quella finestra del programma nella quale si possono settare parametri importanti, aprire altri pannelli contestuali o anche animare gli elementi.

Qui sotto è presentato il pannello forse più utilizzato in LW, il pannello *Surface editor* (Items->Surface editor) con il quale si attribuiscono i materiali alle superfici degli oggetti, nel suo tab *basic*:

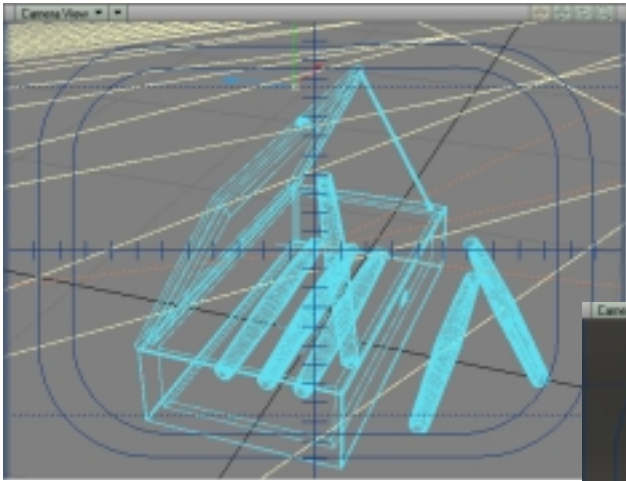


A sinistra abbiamo l'elenco superfici: in questa scena abbiamo solo un oggetto multimesh (o multilayer o multilivello, in pratica un oggetto multiplo che rappresenta già una scena creata in modeler). L'oggetto è salvato col nome scatola2.lwo e contiene una scatola di sigari con le seguenti superfici: scatola.coperchio, scatola.coperchio1, scatola.corpo, scatola.corpo1-5, scatola.fermo,scatola.interno, scatola.oro, scatola.velluto.

Dato che in modeler abbiamo creato un oggetto multioggetto .-) è bene seguire la consuetudine (derivata dal linguaggio di programmazione C) di nominare tutte le superfici che appartengono ad un unico sub-oggetto con la notazione *nomesub-oggetto.nomesuperficie*. In questo modo gestiremo più logicamente il progetto anche col passare del tempo.

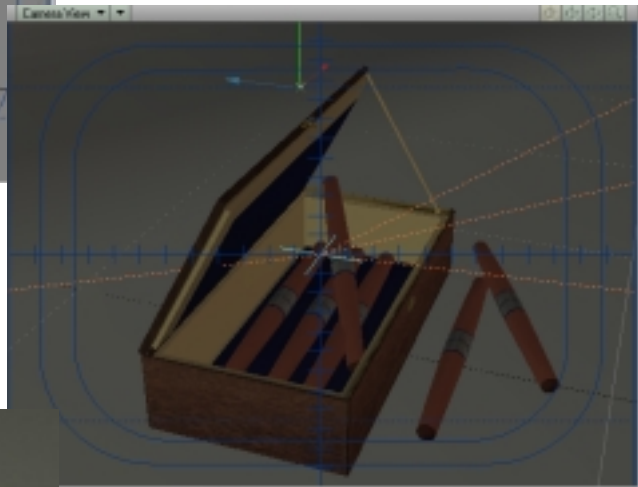
Allo stesso modo nominiamo le superfici del suboggetto *sigaro* come sigaro, sigaro.int e sigaro.marca.

Adesso abbiamo nella finestra a sinistra tutte le superfici che abbiamo attribuito al nostro oggetto multilayer (per essere sicuro di potermi spiegare vi faccio vedere qui sotto l'oggetto di cui si parla, nella vista 'camera' di layout, in modalità wireframe e in quella texturized):



Come si può vedere ci sono sei sigari, una scatola (corpo e coperchio), i fermi in metallo e un reggicoperchio... se si dice così. Ho scelto di modellarli insieme, in un unico oggetto in modeler, per semplicità: così si tengono meglio d'occhio le relative proporzioni.

Subito sotto vi mostro un rendering di prova, così ci sarà più facile parlare poi delle qualità delle superfici.



Abbiamo una scatola di legno ricoperta in pannelli di radica, due fogli di velluto tra il violetto ed il blu, la superficie dei sigari (la foglia di copertura ed il trinciato interno), la marca del sigaro e poco altro.

Quando parliamo di superfici di un oggetto intendiamo riferirci alle qualità ottiche del materiale: il colore, la quantità di riflessi, la sensazione di rugosità, l'arrotondamento e la luminosità. Ogni materiale è diverso da tutti gli altri per queste, ed altre, caratteristiche.

LW ci permette di avere a disposizione quello che io ritengo il più potente strumento di definizione delle superfici oggi disponibile.

Il livello colore – colori, sarebbe meglio, definisce i valori cromatici, ovviamente.

Il livello luminosity indica, pensa un po', i valori di luminosità, cioè anche le aree di emissione di luce propria.

il livello diffuse, diversamente da quanto avviene in molti altri programmi 3d, dove è un doppio del livello colore, indica la quantità di luce che viene restituita dalla superficie all'ambiente, cioè quella parte di luce che non viene assorbita. Possiamo pensare alla luce diffusa come a raggi luminosi che vengono *sparpagliati* dalla superficie.

il livello specularity indica il valore di riflessione della sorgente di luce (la luce riflessa, generalmente, è dello stesso colore della sorgente). Alti valori, come si può immaginare, sono propri di sfere metalliche o di vetro.

Il livello glossiness è attivo solo se esiste un valore di specularity diverso da 0. Un alto valore di glossiness indica un materiale tipo metallo (riflessioni puntiformi e dure), un basso valore indica un materiale tipo plastica (riflessioni molto diffuse e morbide)

Il livello reflection indica quanto del mondo circostante sarà rappresentato sulla superficie (uno specchio avrà il valore di reflection=100).

Il livello transparency indica, stupore!, se è possibile vedere attraverso la superficie. Si badi che i raggi di luce vengono deviati dalla geometria delle superfici (in effetti, in LW, è possibile far passare un raggio di luce bianca attraverso un prisma e farne uscire un fascio di luce con tutti i valori cromatici, dal rosso al violetto).

Il refraction index indica proprio l'indice di rifrazione (diverso dal vetro al diamante, salvo che quest'ultimo non sia un fondo di bottiglia).

Il livello translucency indica quanta luce passa attraverso la superficie, proprio come attraverso un foglio di carta.

Il livello bump, importantissimo, permette di attribuire ad una superficie un aspetto rugoso che simuli la tridimensionalità di pori e rughe (una buccia di arancio, ad esempio – in effetti le superfici perfettamente lisce non esistevano neanche nei programmi 3d del 1986).

Il surface smoothing crea delle superfici smussate in cui ogni poligono sfuma nell'altro. E' fondamentale.

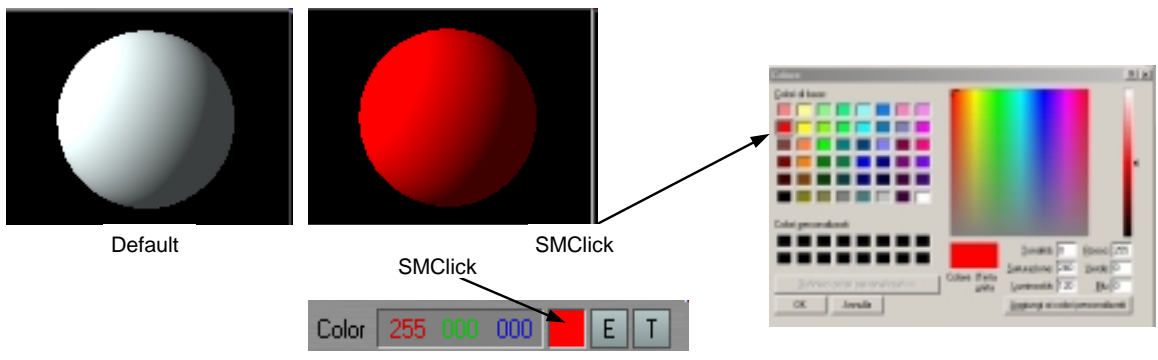
La cosa bella, resa possibile dalla potenza dei calcolatori di oggi, è che i valori della superficie vengono mostrati nella finestra di anteprima in tempo reale: aggiungete delle rughe? Le vedrete subito all'opera.

Ogni livello presente nella finestra principale presenta i seguenti punti di controllo:



Lasciamo per un attimo il nostro progetto della scatola di sigari (ci torneremo in seguito) per caricare un semplice oggetto sferico (Clear scene, Items->Add item->load object). La sfera avrà una unica superficie non nominata (default).

Per nominare le superfici, in modeler, basta selezionare tutti i poligoni che vogliamo costituiscano la superficie e poi premere 'q' sulla tastiera. Si aprirà una finestrella da cui potremo nominare in modo univoco e poi riconoscere sempre la superficie creata. I nomi della superficie, come i valori a lei attribuiti, saranno sempre salvati nel file oggetto .lwo. Da ciò deriva che ogni volta che voi modificate i valori della superficie in layout dovrete poi obbligatoriamente salvare l'oggetto modificato con modeler. Pena la perdita delle modifiche.



Hey! Abbiamo dato un colore alla sfera!



SMClick (Icona T) fa apparire il pannello dei Layer del Texture editor.... Spaventati?

Operazioni con i layers

Tipo di layers (immagine, procedurale, gradiente)

Modo di fusione
Nomi e qualità dei layers

Proiezione (planare, sferica, cilindrica...)

Nome dell'immagine (se il layer è Image map)

Anteprima del layer selezionato

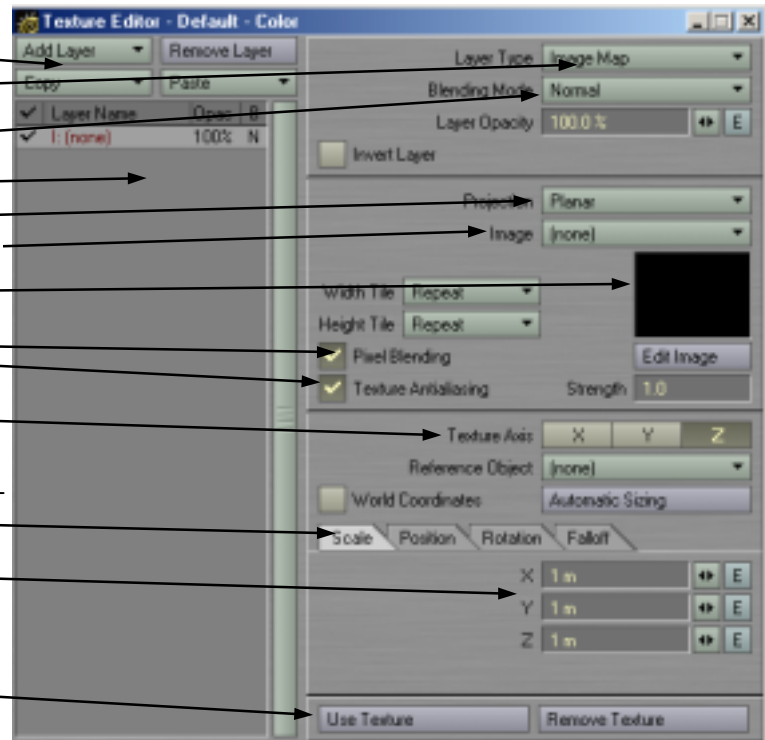
Opzioni per evitare scalettature

Asse su cui la texture è mappata

Dimensioni, posizioni, rotazione e decadenza della texture

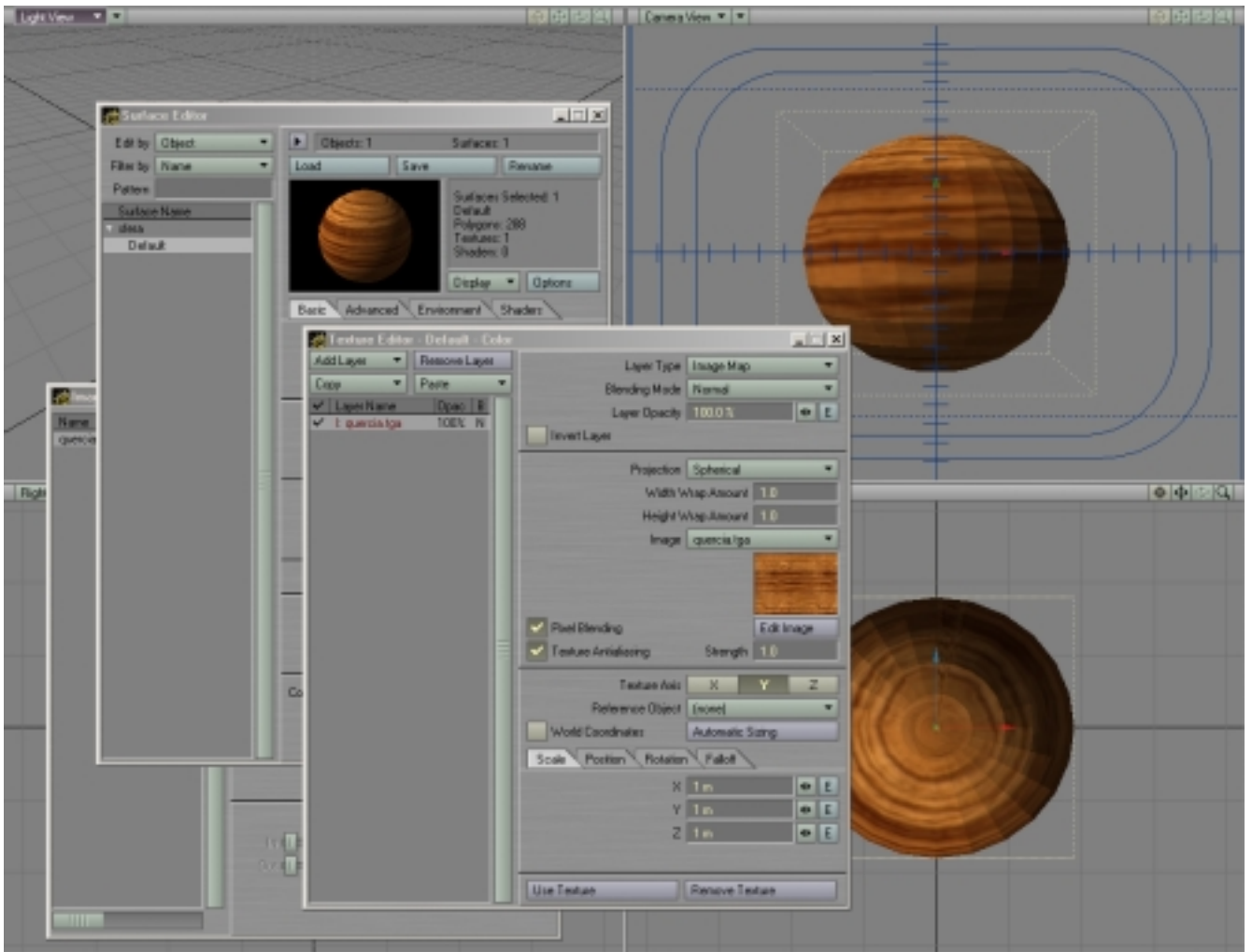
Valori di cui sopra

Lo stesso che YES-NO



Ho intenzione di applicare una semplice immagine sulla sfera, un'immagine di legno. Se ho lasciato selezionato di default il *Layer Type* su *Image Map*, devo solo cliccare su *Image->Load image*, caricare l'immagine voluta (quercia.tga), scegliere *Projection->spherical*, *Texture Axis->Y* e cliccare su *Automatic Sizing* per dare alla mappa di immagine la dimensione corretta (noterete che i valori x,y,z sottostanti varieranno). Il resto può essere lasciato invariato.

Nella pagina successiva vedete parte dello schermo di layout mostrante le modifiche apportate.



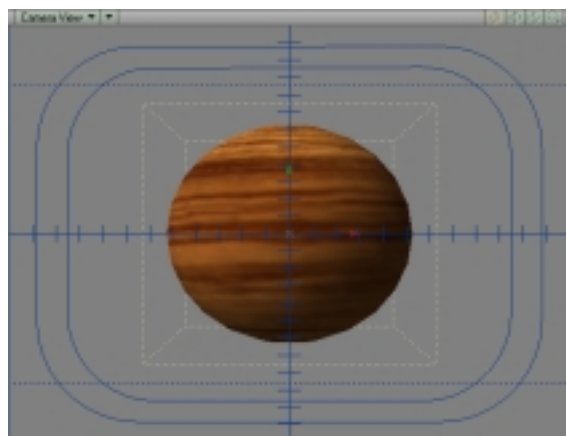
Orrore! Tutti quei poligoni! Una sfera sfaccettata non era nelle mie intenzioni. Devo aver dimenticato di spuntare *Smoothing* nel pannello precedente...

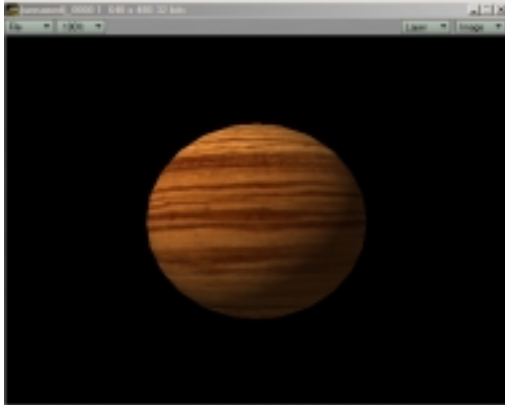


Infatti....->



Adesso il preview della finestra della camera mostra il semplice materiale creato:





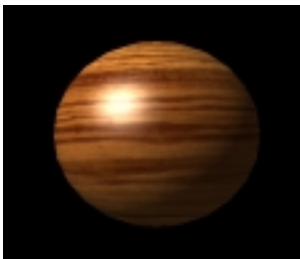
La palla di legno di quercia dà un rendering di questo tipo, con le ombre settate

(F9 per avere il rendering, *Items->Rendering->rendering options* per far apparire il pannello seguente, dove devono essere settati i parametri come mostrato: ne parleremo più dettagliatamente in una lezione apposita.... *Ogni* azione in LW ha un pannello che porta ad altri pannelli i quali conducono a pannellini piccolini che, dopo sette-otto passaggi mostrano che c'è anche quello che non si poteva immaginare .-).

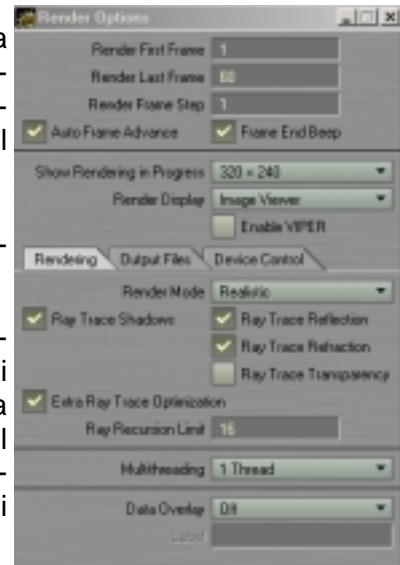
Certo, le venature sono suggestive, ma la palla sembra tutto tranne che una palla di legno levigata, invecchiata, lucidata dal tempo, con un po' di macchie lasciate dalle mosche e con le scabrosità che mia figlia ci provocò quei giorni che aveva deciso di giocare a bocce in soggiorno.

Andiamo a vedere cosa si può fare con il livello di specularità

Settiamo (nel pannello *Surface editor*) il valore della specularità a 80% (era una sfera levigata bene, con un po' di cera sulla superficie). Settiamo il valore del *glossiness* a 30% (la cera era calda e ben disposta sulla superficie). Il cambiamento è aggiornato in tempo reale in tutte le finestre che mostrano il display OpenGL e viene mostrato qui sotto.



E' già qualcosa di più convincente... Mancano ancora le macchie di mosca e le abrasioni. Le macchie non sono altro che aree della superficie su cui la luce viene assorbita in modo diverso che nel resto dell'oggetto. Se le macchie appaiono scure significa che in quel punto *meno* luce viene ridistribuita all'ambiente e quindi il valore *diffuse* è minore. Se abbassiamo il valore *diffuse* nel pannello principale vedremo che tutta la superficie diventerà più scura. E non è questo quello che vogliamo. Noi vogliamo delle macchie di mosca (o qualcosa di molto simile... come macchie di gatto... se avete dei gatti capite quello che intendo).



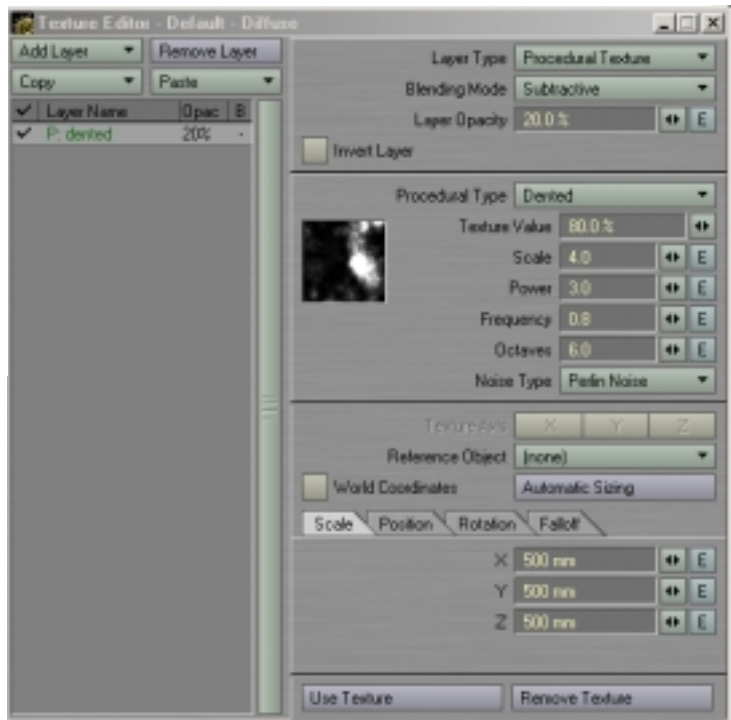
Andiamo allora ad aprire il sottopannello texture, come abbiamo fatto prima per mettere l'immagine *quercia.tga* sul canale di colore, operando però sul livello *diffuse*. Dobbiamo creare macchioline scure e quindi dobbiamo applicare una tessitura frattale (generata cioè da iterazioni ripetute di semplici operazioni matematiche->per avere informazioni di base sull'universo della geometria frattale si inizi con A.Mandelbrot, *Oggetti frattali*, Mondadori).

La geometria frattale, essendo generata matematicamente non ha una risoluzione predefinita e si può operare su una texture frattale con ingrandimenti ripetuti senza perdita di dettaglio. Se ingrandite una foto mappata su una superficie prima o poi, inevitabilmente, vi trovate di fronte a quei brutti quadratoni che sarebbero poi i pixels...Ingrandite una mappa frattale e vi trovate a dubitare del buon vecchio Euclide.

Provate a settare il pannello texture come indicato nell'immagine seguente:

Come si vede ho settato il tipo di texture come procedurale, il modo di fusione a *subtractive* (devo sottrarre la luminosità nelle macchie), la scala a 500 mm (la sfera ha una dimensione di circa 1,5 m) e la texture procedurale deve essere ripetuta più volte all'interno del solido.

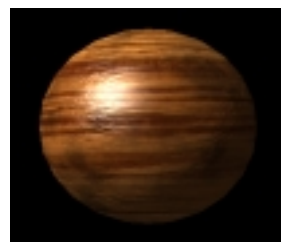
in effetti, lasciando il valore della diffusione a 100% nel pannello precedente, sottraggo il 20% del valore della texture a quel 100% ottenendo così valori di diffusione variabili da 100% a 80%, dato che la texture, rappresentata in scala di grigi, ha valore 0 dove c'è il colore nero e 100 dove c'è il colore bianco puro. Gli altri valori per il momento non ci interessano, per non andare alla deriva.



Se volete potete aumentare il *Layer opacity* per rendere più intense le macchie. Adesso aggiungiamo delle rigature sulla superficie (ricordate... mia figlia...) Andiamo sul canale *bump*, cambiamo il tipo del layer in procedurale e scegliamo STClouds (nuvole tipo SaTurno... non scherzo).



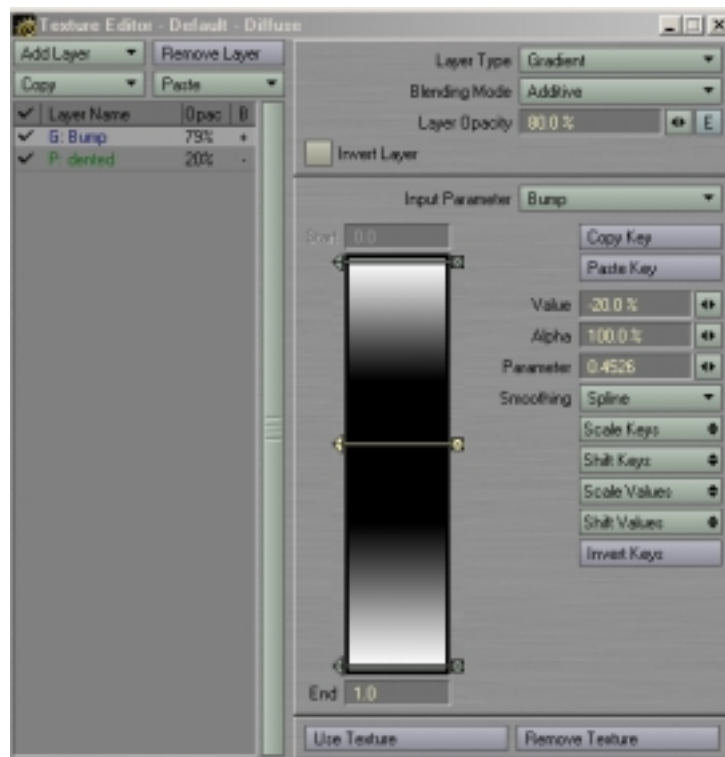
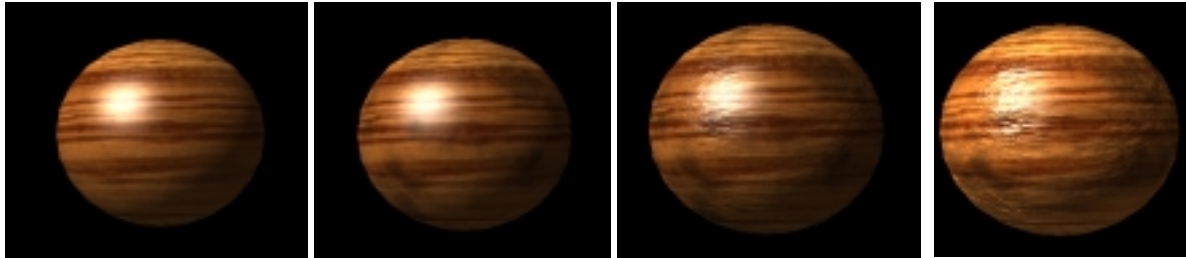
Con i parametri settati come mostrato qui a sinistra avrete un rendering come sotto (ed il valore di *bump* settato a -30 nel pannello principale – i graffi sono *profondi* ed hanno quindi valore negativo)



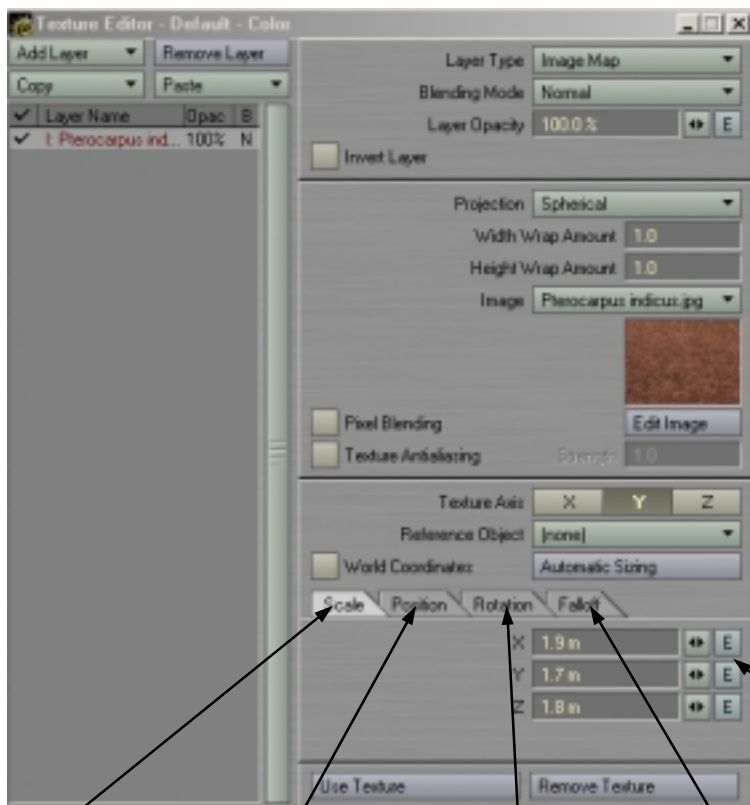
Non è particolarmente realistico ma, insomma.

Mi viene in mente però che le righe che rovinavano la superficie avevano un altro aspetto... un po' più *graffiato*. Per forza: i graffi sono zone dove la luce viene riflessa in modo del tutto diverso.

L'ultima immagine a destra viene ottenuta aggiungendo, per questo scopo, un layer *gradient* al livello generale di *diffuse*:



Il layer *gradient* opera come un filtro su valori settati da altri layer (in questo caso *bump*) operando su suddetti valori delle operazioni aritmetiche di incremento come generate dai valori che noi impostiamo nella scala verticale (operando sulle rugosità: in un range di rugosità che va da 0min a 1max, il valore della diffusione generato dal gradiente settato su tre punti 80, -20, 80 – su una scala 0-100 – genera, in corrispondenza della rugosità, valori risultanti di diffusione $0 \cdot 80 = 0$, $.5 \cdot -20 = -10$, $1 \cdot 80 = 80$... E le nostre rigature saranno massimamente diffuse nel punto apparentemente più alto (almeno all'80%, cioè in rapporto al valore che questo layer ha sul canale diffusione... L'altro layer, *dented*, ha un peso del 20%). Spero di non avervi confuso le idee nella testa, vi assicuro che questa non era assolutamente la mia intenzione.



scala

Posizione

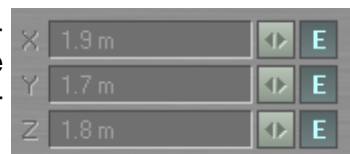
Ruota

Dissolvenza

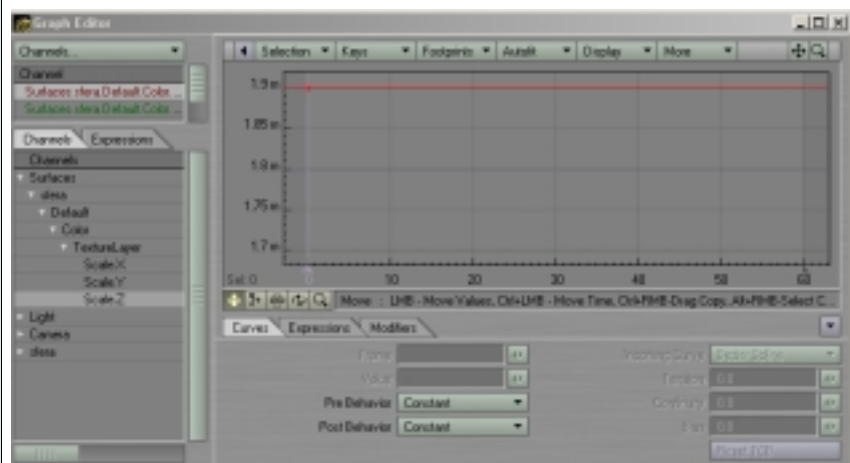
Involuppo

Sempre nel pannello aperto con Layout->Surface editor->T (in qualunque canale) in basso a destra troviamo dei tabs: Scale, Position, Rotation e Falloff (dissolvenza). Impostando per ciascuno di essi i tre valori sugli assi x,y,z o Head, Pitch, Bank (per la rotazione) riusciamo a disporre precisamente la texture sulla superficie. Nel caso di una fotografia, per esempio, possiamo far coincidere un particolare con una posizione in un poligono. Tutto questo è possibile per qualunque degli assi su cui stiamo lavorando. A destra dei valori (x,y,z in questo caso) abbiamo un pulsante di involuppo 'E'. Proviamo a cliccarlo...

Le linee di input diventano inattive e le tre 'E' risultano selezionate (anche se avevo cliccato solo su una di esse... E si apre un'altra finestra... che ha l'aria di essere una cosa completamente nuova e zeppa di opzioni... un'altra!

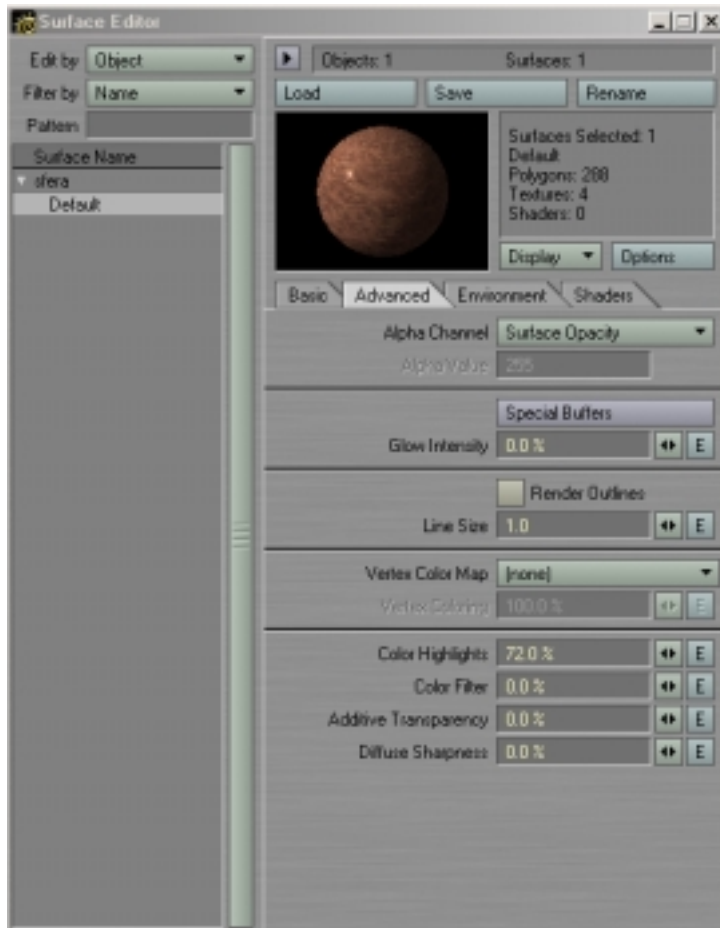


Quello presentato qui a sinistra è il Graph Editor di LW a cui dedicheremo un intero e lungo capitolo. Attraverso questo strumento è possibile animare tutte o quasi tutte le caratteristiche di una scena. In questo caso, avendo cliccato 'E' nel contesto *surface editor->Texture editor->scale*, abbiamo a disposizione uno strumento con il quale variare nel tempo la scala della texture sulla superficie. Come si vede all'estrema sinistra, nel folder *Channels*, siamo posizionati sul canale *Superfici->(oggetto)sfera->(superficie)Default->(canale)colore->Layer(texture di default)->scale x,y,z...*



Torniamo al nostro pannello principale, quello con tutti i canali di superficie (Surface Editor). Per ora abbiamo analizzato solo il folder Basic. Cioè quello che definisce solo i valori di base della superficie. Clicchiamo su Advanced e attiviamo la schermata a sinistra.

Il folder Advanced è molto più semplice: contiene i seguenti parametri



* Alpha Channel

* l'intensità di **Glow** (bagliore) L'effetto *glow* in questo pannello è pesato, ma per essere attivato bisogna aprire un altro pannello: Layout->Scene->Effects->Processing, attivare spuntandolo il pulsante *enable glow* e settare l'intensità (relativa alla scena). In questo pannello qui a sinistra settate l'intensità relativa al *glow* dell'oggetto

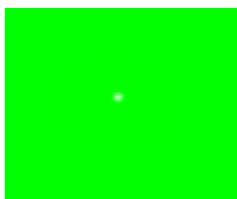
* **Color Highlights** (se si vuole che l'alone di specularità, dominato dal colore della luce incidente, sfumi gradualmente nel colore della superficie illuminata)

* **Color filter** (se la superficie è trasparente questo valore indica quanto un oggetto visto attraverso la trasparenza modifica il suo colore, *virando* verso il colore della superficie che qui si sta modificando. In pratica, immaginate un bicchiere di vetro colorato attraverso il quale si vede una biglia di cromo: quanto dovrà il cromo acquisire del colore della superficie di vetro? Quanto dovrà sembrare azzurrognolo se il vetro è azzurrato? Oppure, il che è lo stesso, quanto dovrà essere colorato il raggio di luce passando attraverso la superficie e quanto dovrà colorare.

* **Additive transparency** con cui la superficie dell'oggetto trasparente viene aggiunta (ADD) ai colori degli oggetti visti attraverso di essa (e si aumenta la luminosità).

* **Diffuse Sharpness** viene usato quando tra la parte illuminata e la parte in ombra di un oggetto si vuole provocare un taglio netto (come l'alba sulla Terra, vista dalla Luna!)

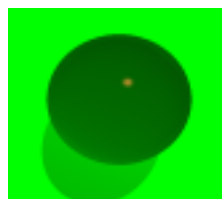
Qui sotto vedete i renderings di una sfera trasparente al 92%, colorata in rosso puro posta davanti a un pannello colorato in verde: come si vede, variando i parametri Color Highlights e Color Filter si modifica il punto di luminosità, che da bianco (il colore della luce) diviene rosso (la superficie), il colore del pannello da verde diventa quasi nero ed il colore e l'intensità dell'ombra tende a scurirsi. In effetti nel primo caso la sfera non si vede nemmeno: ogni raggio proveniente dal pannello verde passa attraverso la sfera senza cambiare. Ricordate: i valori di trasparenza, colore, intensità di luce e tutto il resto rimangono immutati. Cambiano solo i due valori indicati.



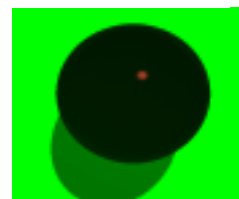
Highlights = 0
Cfilter = 0



H = 50
CF = 30



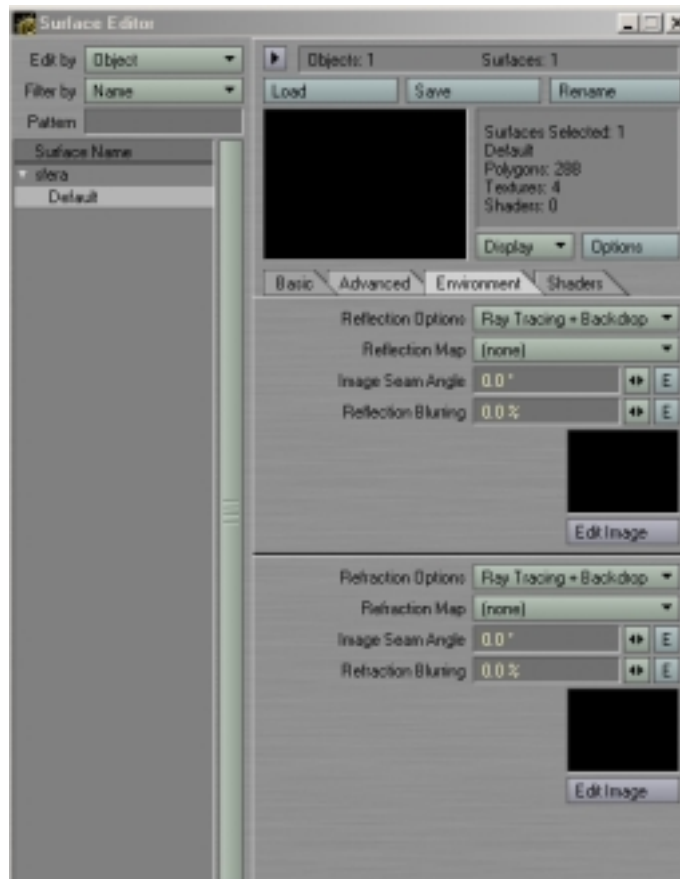
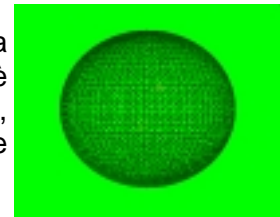
H = 80
CF = 50



H = 80
CF = 80

Ancora due parole su *Render Outlines*: spuntiamo questo pulsante se vogliamo che l'oggetto sia disegnato non attraverso le poligonali ma attraverso le linee degli spigoli (delle quali si può impostare lo spessore) come fosse un wireframe.

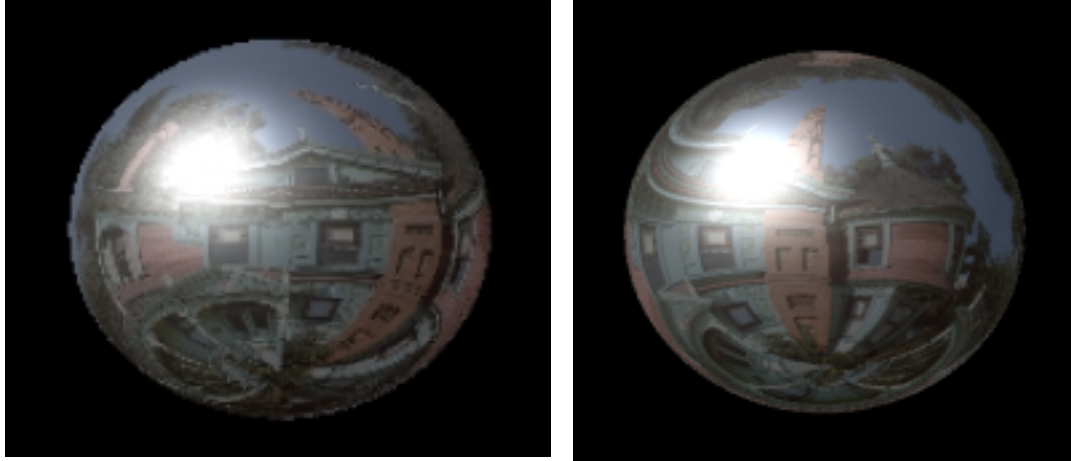
Ovviamente, non essendoci poligoni, l'oggetto non lancia ombre. La grande densità di facce è dovuta al fatto che la sfera, in realtà, è subpatched (i poligoni sono stato trasformati in sub patch in modeler, premendo il tasto TAB). E' questa un'opzione interessantissima. Utile ai progettisti, suppongo.



Il terzo Tab dell'editor di superfici è nominato *Environment*, cioè Ambiente. In questa sezione, molto semplice, si impostano le opzioni di riflessione e di rifrazione. Nel tab *basic* si opera su questi due parametri editando le loro texture. Qui si lavora, più in generale, sul motore di rendering.

Nel caso della riflessione e della rifrazione il motore di tracciamento del raggio può essere molto complesso e esigente, in termini di tempo e di potenza.

Se mettete in scena un bicchiere davanti ad una finestra e impostate il motore di calcolo come ray-tracing puro, e magari avete un processore non molto potente... ok, potete andare a comprarvi il giornale, fare una passeggiata in riva al mare, qualche spesa ecc. ecc. prima che il rendering sia concluso. Per ovviare a questo collo di bottiglia, e nell'attesa di processori da 200 teraflops, il mondo della programmazione 3d ha pensato ad una scorciatoia. In sostanza la riflessione è una mappatura sferica di un'immagine ambientale sulle superfici. Non è necessario ricreare l'ambiente e l'illuminazione di una stanza ove ambientare la nostra scena se abbiamo un'immagine fotografica di quella stanza, o di una molto simile. Se l'abbiamo o ce la possiamo procurare basterà usarla come mappa sferica di riflessione/rifrazione. In questo modo i tempi delle elaborazioni si accorciano enormemente.

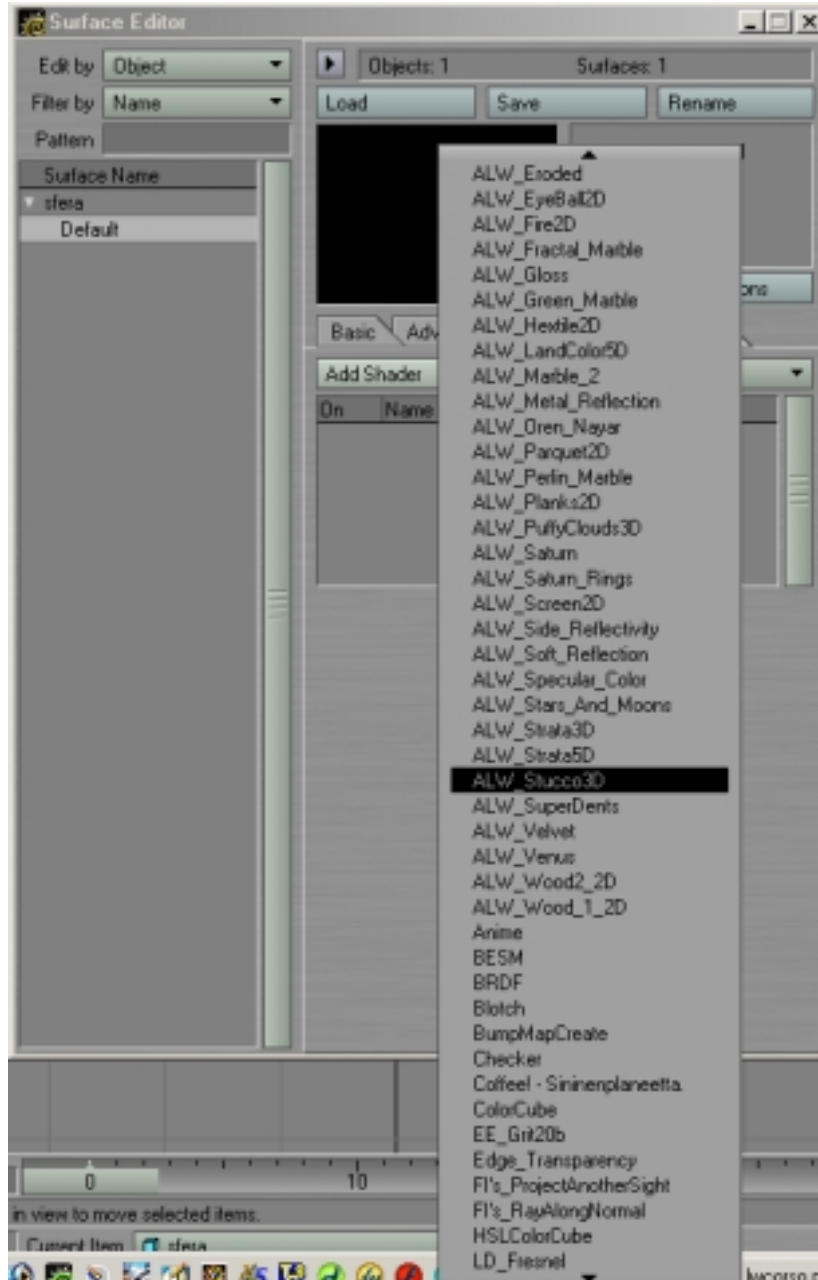


A sinistra una mappatura sferica 'bruta', a destra con *Image Seam Angle* = 90°.

Possiamo scegliere quattro opzioni di riflessione/rifrazione: Sfondo, Raytracing+sfondo, Mappa sferica, Raytracing+Mappa sferica. Io normalmente uso quest'ultima, a meno che sullo sfondo non ci siano elementi troppi importanti per essere tralasciati. Per altri suggerimenti relativi alle riflessioni-rifrazioni si veda il capitolo dedicato al rendering.

Image Seam Angle serve per nascondere l'effetto di sovrapposizione dei bordi che accade quando si arrotola su una sfera un'immagine piana (l'effetto di saldatura, evidentemente innaturale).

Il *Reflection blurring* è il grado di sfumatura e di conseguente *confusione* che si vuole dare all'immagine. Nella grafica 3d è tutto un tentare di eliminare la precisione del calcolo algoritmico per eliminare l'artificialità del digitale.



A sinistra vedete l'attivazione del menù a discesa dell'ultimo tab, *Shaders*. E' la prima volta che ci capita di accedere ad uno spazio così criptico: tutti possono vedere che ci sono un sacco di voci ne menù, tante che addirittura, in alto, è presente una freccetta col significato di *continua*...

LW usa questi menù a discesa anche per permettere all'utente di scegliere tra un numero variabile di opzioni. In questo caso ci permette di applicare all'*insieme* delle opzioni di superficie uno *shader*, cioè un *plug-in* che, attraverso algoritmi proprietari, disegna una superficie particolare, non nativa del Surface Editor di LW, aggiunta quasi sempre da terze parti e spesso successivamente inserita nel pacchetto dalla casa madre. Per una spiegazione più dettagliata di questi oggetti, che sono la vera forza dei programmi moderni, si rimanda alla sezione apposita (I plug-ins).

Se mi sono spiegato, sarà chiaro che questa quarta sezione del Surface Editor è un mondo a sé.

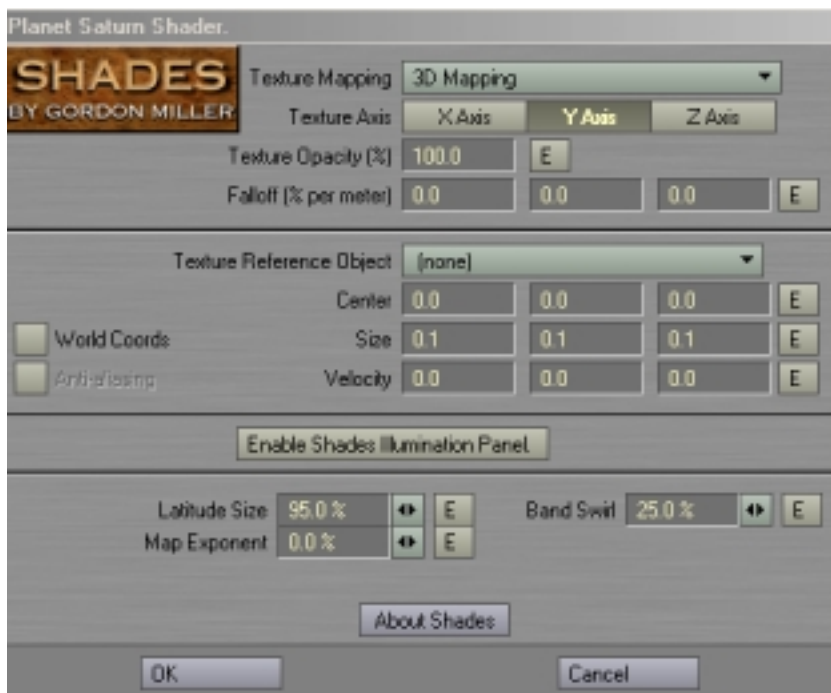
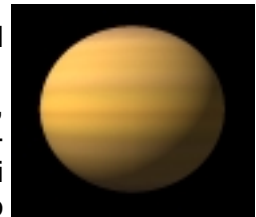
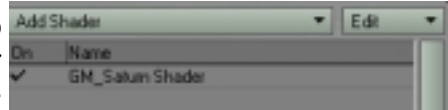
Abbiamo una sfera in scena. Abbiamo una voce del menù che suona come ALW_Saturn, sotto abbiamo un ALW_Saturn_rings... Stai a vedere che selezionando il primo, per la superficie della sfera, ottengo in qualche modo l'apparenza del gigante gassoso...

Vediamo. Seleziono ALW_Saturn dall'elenco, guardo l'anteprima...

A questo punto lo si poteva immaginare: eccolo qui il nostro Saturno senza anelli.

Nello spazio sotto il menù appena attivato abbiamo lo shader caricato: il segno di spunta indica che lo shader è attivo. Comodo! Posso anche disattivarlo senza perdere i parametri che erano stati settati, magari vedendo l'effetto di un altro shader o vedendone altri, insieme, con e senza il primo!

In effetti sotto il menù *Edit* si può accedere a funzioni di copia, taglia, incolla e apri. La maggior parte dei plug-in hanno infatti una interfaccia grafica, una finestra nella quale si possono settare i più vari parametri di cui lo shader ha bisogno. Nel caso in esame, se clicco due volte SM sul nome spuntato ottengo il pannello seguente:

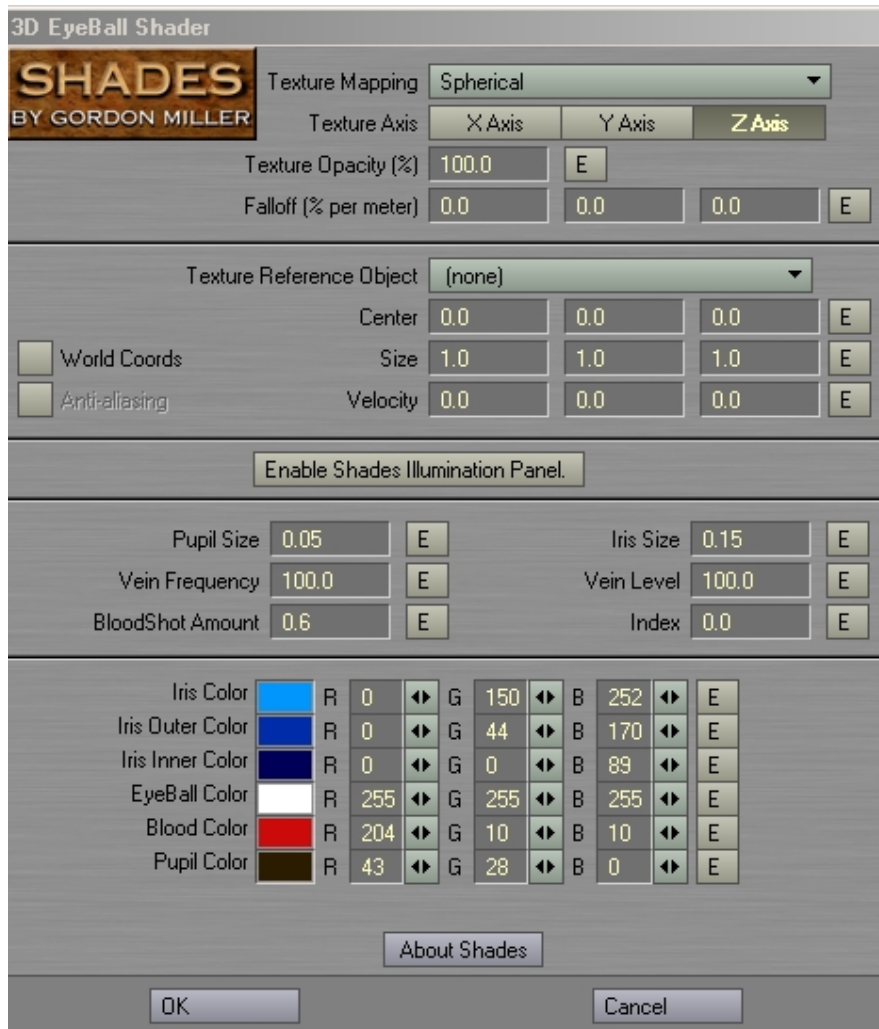


La prima volta che vedi una cosa del genere, la prima volta, cioè, che attivai uno di questi shaders, appena scaricati royalty free da internet, ebbene, la prima volta mi alzai dalla sedia e mi andai a preparare un caffè. Dovevo digerire per bene l'idea che esistesse nel mondo una comunità di utenti che studiava, programmava, pubblicava e rendeva disponibili a tutti gratuitamente simili pezzi di software. ALW_Saturn fa parte di una suite di plug-in, *Shades* di Gordon Miller, intesa ad offrire all'utenza tutti gli strumenti per moltiplicare esponenzialmente la potenza dell'ambiente di sviluppo dei propri progetti.



Guardate se scelgo ALW_EyeBall2D:

E se clicco sul nome dello shader:



Si può progettare un occhio. Certo non fotorealistico, se vogliamo essere critici. Ma ugualmente di grande utilità a tutti quegli animatori di fumetti che non devono riprogrammare e ridisegnare un occhio.

Alla prima apertura del programma gli shaders effettivamente presenti sono pochi. Per conoscere il modo di gestire questa potenzialmente enorme libreria di risorse si consulti, come già detto, la sezione dei plug-ins. Inoltre si vedano le appendici relative alle risorse di rete e al database dei plug-ins disponibili oggi per LW.

L'utilizzo delle immagini su color, specularity e bump

Una superficie realistica spesso è anche molto dettagliata. Pensiamo ad un tavolo rustico di legno. Il legno, soprattutto se vecchio, è molto diversificato e complesso: difficilmente una tessitura si ripeterà per tutta la superficie senza variazioni. Il suo realismo sarà quindi dato dalla presenza di venature, imperfezioni, veri e propri buchi... L'unico modo per ottenere ciò, sarà banale, è quello di utilizzare una fotografia di un vero tavolo vecchio.



Questa è una buona immagine: ha una risoluzione sufficiente, una serie di variazioni e di dettagli, una scala adatta a ricoprire superfici di discrete dimensioni ed è *tileless* - si può applicare ad una superficie, cioè, ripetendola come una mattonella senza vedere i punti di giunzione. Ciononostante da sola non può bastare. Vediamo perché.

Andiamo su Modeler

Create->Box o 'X'

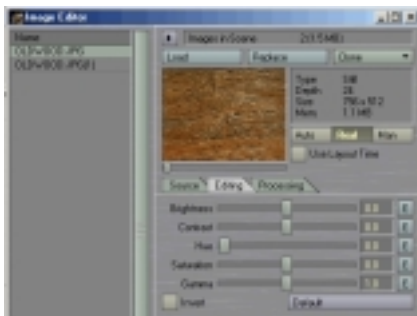
N'->Depth=0

In questo modo abbiamo creato un piano bidimensionale azzerando il valore della profondità Z (*depth*). Meglio per ora semplificare le cose. Salviamo l'oggetto con 's' (nominandolo *box2d.lwojed*) ed inviamolo a Layout:

's', SendObjectToLayout

Probabilmente, una volta che è apparsa la finestra di Layout, vedrete nella finestra della Camera, un quadrato giallo (è giallo perché è selezionato) trasparente. La normale al poligono è rivolta nella direzione opposta all'osservatore. Tornate in Modeler e flipate la normale con 'f'. In Layout il quadrato è ora ben visibile! Apriamo il pannello *Image Editor* e carichiamo il file *oldwood.jpg* (si suppone che la vostra Content directory sia stata impostata come CD):

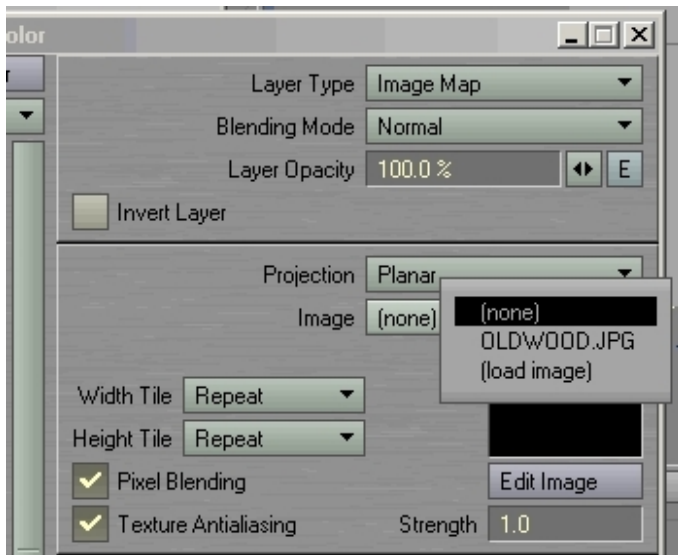
Layout->ImageEditor (o ctrl+F4)->Load->oldwood.jpg



Adesso abbiamo inserito nel progetto l'immagine e la possiamo utilizzare. Apriamo il pannello *Surface editor* e selezioniamo la texture del canale di colore (la superficie è una sola ed è quindi chiamata *default*):

Layout->SurfaceEditor (o ctrl+F3)
Color->T

Clicchiamo nel *texture editor* che si è aperto sul menù a comparsa che ci permette di scegliere l'immagine da mappare, per l'appunto, sul canale di colore:

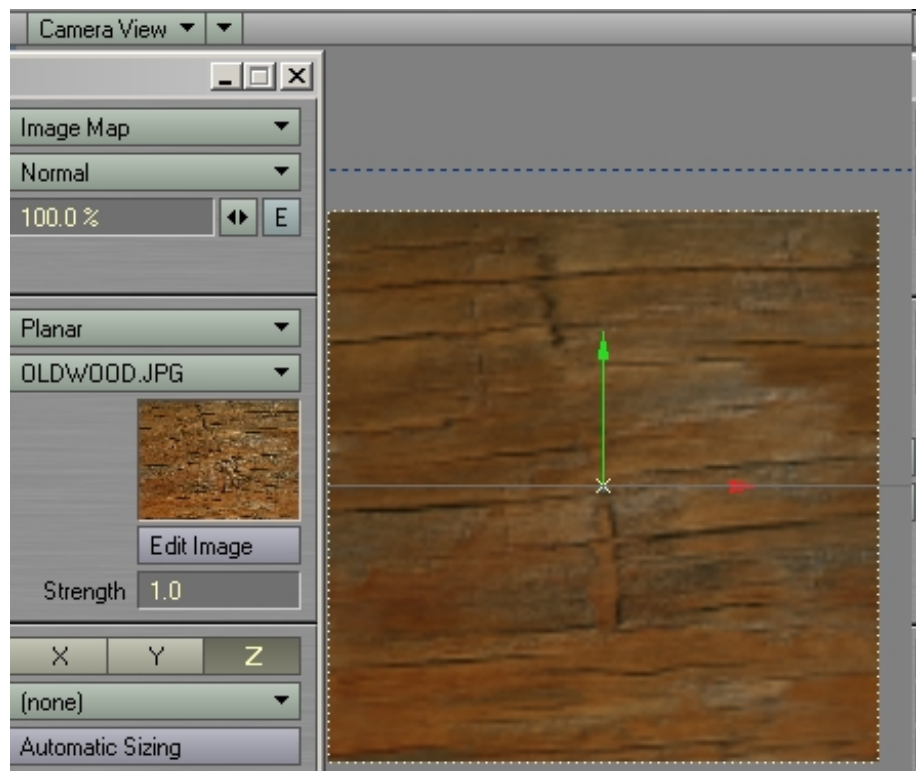


Come si vede abbiamo tre opzioni: *none* indica che non vogliamo usare alcuna immagine (scarichiamo eventualmente quella già selezionata) OLDWOOD.JPG indica il nome dell'immagine che abbiamo già inserito nel progetto con il passo precedente *Load image* permette di caricare una nuova immagine non ancora inserita nel progetto

In questo caso, naturalmente, scegliamo il nome del file OLDWOOD.JPG. Vediamo che la nostra immagine viene subito mappata sulla superficie.

Per vedere le textures di immagine anche nella quadview che è riferita alla camera, bisogna attivare la vista OpenGL nel pannello delle opzioni di display, accessibile premendo il tasto 'd' e spuntando l'opzione corrispondente. Se abbiamo una scheda video sufficientemente potente (32Mb di RAM almeno, con accelerazione hardware e compatibile OpenGL) non abbiamo problemi ad attivare tutte le opzioni ed avere così sotto controllo il nostro progetto con una anteprima in tempo reale. YUP!

L'immagine viene, secondo le nostre aspettative, correttamente mappata sull'asse delle Z. Nel caso in cui l'immagine risultasse distorta possiamo sempre mapparla sulle X o sulle Y. In questo caso, essendo un poligono piano, probabilmente il programma ha saggiamente scelto l'unica mappatura possibile. Il modo di mappatura è *planar*, il che si spiega da solo. E' una immagine che funziona anche per le dimensioni, riempiendo il piano senza doversi ripetere. Se così non fosse possiamo sempre cliccare su *Automatic Sizing*.

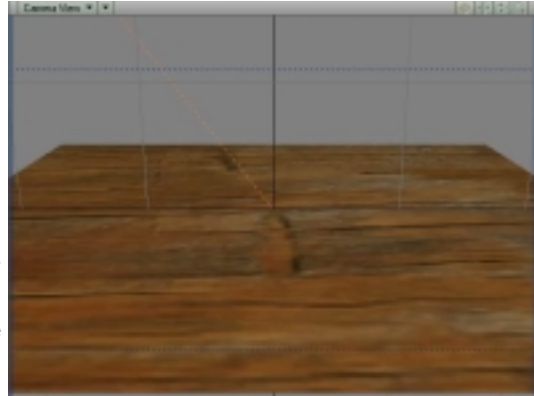


Vista così sembra già una bella cosa. In realtà siamo ancora lontani dalla sufficienza. Vogliamo simulare un piano di legno vecchio e certo non basta disegnare un foglio con sopra stampata una fotografia!

Dobbiamo occuparci di simulare la tridimensionalità dei dettagli, gli scavi ed i rilievi, i punti di luce, quelli dove, col tempo, si è formata una patina di *cera* naturale che riflette la luce ecc. Per apprezzare i cambiamenti che faremo bisogna innanzitutto scegliere l'oggetto come *target* della camera e della luce (ne abbiamo una sola, per il momento, di default), in modo da poter inclinare il nostro punto di vista tra i 30° ed i 45°.

Seleziona la camera
Layout->Items->motion options (oppure 'm')
Target items->box2d

Layout->move->trascinare il pivot della camera in modo che nella vista camera ci sia qualcosa di simile alla figura seguente



Come si diceva: una stampa fotografica abbandonata su un pavimento grigio... Prima cosa: cercare di dare un po' di rugosità alla superficie utilizzando il canale di *bump*.



I canali del *surface editor* operano in modalità a 256 toni di grigio: il canale *bump*, ad esempio, associa la rugosità più alta al pixel col valore più alto di luminosità. Se abbiamo tre pixel adiacenti con valori RGB 0,0,0 (nero), 127,127,127 (grigio medio) e 255,255,255 (bianco) il canale di *bump* assocerà al primo il massimo valore di rugosità negativa (il solco), al secondo il valore neutro (il piano) al terzo il valore massimo positivo (il rilievo). A conseguenza di ciò la migliore immagine per il canale di *bump* è quella in scala di grigio. Si possono usare certamente anche immagini a colori, si può usare per i canali di *bump*, *specularity*, *trasparenza* ed altro la stessa immagine a colori che abbiamo mappato sul canale di colore ma, in pratica, questo non ha molto senso: noi dobbiamo mappare su questi altri canali una serie di informazioni sulla luminosità e per far ciò la scala di 256 toni di grigio offre lo strumento migliore.

Prima di iniziare ad utilizzare esclusivamente LightWave avevo la directory di immagini per le textures con ogni file triplicato: *oldwood.jpg*, a colori per la mappa di colore, *oldwood_bmp.jpg* per la mappa di *bump*, *oldwood_spc.jpg* per la mappa di specularità... un sacco di dati inutili. Quello di cui avevo bisogno era solo una mappa colore che, nella versione in scala di grigio, variati i valori di luminosità e contrasto, servisse per ogni evenienza. Prima di LW dovevo passare da una sessione di triplicazione dei files e di correzione dei valori di due di questi in Photoshop.

Riapriamo il pannello *Image editor*. Con ancora selezionata la nostra immagine a 24 bit colore utilizziamo la funzione *Clone->Instance*. Avremo subito un'altra immagine, identica alla prima, nominata OLDWOOD.JPG(1). Nessun nuovo file sull'Hard Disk. Solo una copia dell'immagine in memoria. Adesso clicchiamo sul tab *editing*: abbiamo attivato un semplice ma molto efficiente editor di immagini con il quale variare i parametri indispensabili ai nostri fini.



Brightness indica la brillantezza, la luminosità dell'immagine (-1 = immagine nera a +1 = immagine bianca);

Contrast è esattamente il *contrasto*, la differenza cioè di valori tra le luci e le ombre;

Hue è la tonalità;

Saturation è la saturazione, cioè l'intensità di colore (un valore di 0 converte i colori in tonalità di grigio);

Gamma varia la luminosità (da 0 = immagine nera a 2=immagine chiara).

Ricordiamo che queste operazioni non alterano in nessun modo il file dell'immagine sull'Hard Disk.

Noi abbiamo bisogno di tre files (come massimo): una immagine a colori per il canale colore, una in grigio (per il canale di bump) ed una in grigio per il canale di specularità. Cloniamo dunque due volte oldwood.jpg.

- ⇒ Al primo clone, da usare sul canale di bump variamo i valori di
- * contrasto (.4),
 - * brillantezza (-.2),
 - * saturation (-1),
 - * gamma (2).

Abbiamo formato una immagine a toni di grigio in cui la differenza di valori tra chiari e scuri è molto esagerata.

- ⇒ Al secondo clone, da usare sul canale di specularità, variamo
- * contrasto (.27)
 - * brillantezza (.27) in misura uguale e positiva (si va per tentativi)
 - * saturazione (-1)

Abbiamo ora un'altra immagine in cui i toni di grigio sono aumentati ma, nel complesso, si sono abbassate le loro differenze.

Il canale di rugosità aveva bisogno di esagerare i livelli delle asperità, per renderle meglio gestibili. Il canale di specularità aveva bisogno di un'immagine che *spalmasse un po' di cera vecchia*, sapete... quella riflettente..., sulla superficie, per simulare le aree di luminosità diffusa. Vediamo.

Riapriamo il *Surface editor*:

- * Nel canale di *Specularity* clicchiamo il pulsante T per attivare il *texture editor* relativo. Impostiamo la mappa oldwood.jpg (2) e nel *Layer opacity* inseriamo il valore di 50% (al posto di 100% di default).
- * Chiudiamo il pannello e torniamo al *Surface editor*.
- * Nel canale di *bump* impostiamo un valore di 50% e clicchiamo il pulsante T. Nuovamente impostiamo la mappa, questa volta, però, scegliendo oldwood.jpg (1).
- * Impostiamo il valore del canale di *glossiness*, nel *Surface editor*, a 20%: senza aprire l'editor di texture questo serve a rendere la specularità in maniera diffusa (ricordiamo che valori alti sono tipici dei metalli).
- * In *Surface editor->advanced* settiamo *Color Highlight* a 50%. In questo modo facciamo sì che i punti di luce (emessi da una lampada di colore bianco) non siano bianchi oltre il 50% ma prendano in parte anche il colore della superficie.

Adesso premiamo F9 per iniziare il rendering.



Possiamo subito vedere l'effetto delle nostre manipolazioni sulla mappa oldwood.jpg. Ora abbiamo dei punti di massima luminosità proprio nei punti giusti, quelli che abbiamo impostato alterando i valori di *brightness* e di *contrast*. Anche l'immagine mappata sul canale di *bump* ha avuto l'effetto desiderato: il piano sembra ora avere una rugosità ed un livello di rilievo realistico.



Si ricorda ancora che l'applicazione di una mappa di *bump* non modifica la geometria di una superficie. Per modificare effettivamente la componente poligonale con una mappa di bump si veda il capitolo sulle mappe di scostamento (*displacement map*).

Questi sono, in realtà, solo i primi passi. Per aggiungere realismo alla superficie si potrebbe giocare con qualche rumore frattale aggiunto al canale *diffuse* per creare qualche macchia di luminosità negativa e simulare così un po' di sporco o consunzione. Ancora si potrebbe aggiungere una mappa di tipo *gradient* alla *specularity* con riferimento al *Light incidence* (non dimenticandoci di settare la luce principale della scena come luce di riferimento).

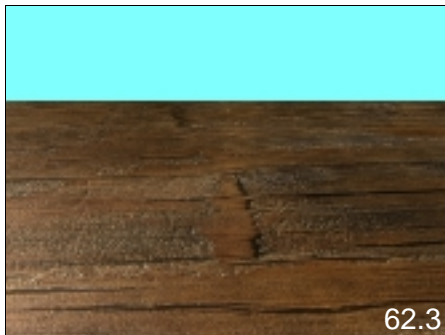
In 62.1 è stato aggiunto al canale *diffuse* un layer frattale di tipo *Turbulent noise*, con opacità del 35% e *blending mode* = *subtractive*.
In 6.2 è stato aggiunto al canale *specularity* anche un layer di tipo *Light incidence* (relativo all'unica luce presente in scena) con valore del 50%.



62.1



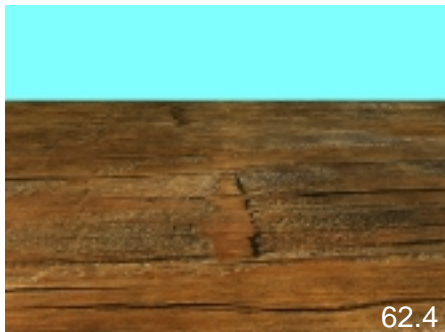
62.2



62.3

In 62.3 si è cambiato soltanto il colore del background per mostrare come l'impressione di eccessivo buio nelle immagini precedenti è data anche dallo sfondo nero.

Naturalmente il realismo di una superficie è strettamente legato alla plausibilità dell'illuminazione presente. Una sola luce, generica come quella di default, non può certo proporre nulla di appetibile. Proviamo a mettere in scena una illuminazione a tre luci, tipica per le illuminazioni da studio (e che verrà spiegata in dettaglio nel capitolo sull'illuminazione).



62.4

In 62.4 abbiamo sostituito l'unica *distant light* con tre *spot light* (una bianca, una a toni rossastri ed una terza a forti toni blu, queste due ultime di luminosità molto inferiore alla prima). L'effetto finale è forse meno suggestivo ed esagerato, ma certamente più banalmente realistico.

Ricapitoliamo dunque: per superfici realistiche – e renderizzate in un tempo molto limitato, sono necessarie foto ben dettagliate, con risoluzione adatta (non è possibile dare uno standard, ma certo più l'immagine contiene pixel meglio è) e in formato non compresso (TGA e BMP) oppure con un livello di compressione molto basso (JPEG compresse a livello 8-9). Bisogna poi creare, quasi sempre, due o più copie assolutamente identiche per usarle sui vari canali, dopo averle rese a toni di grigio ed averle modificate nei valori di luminosità e contrasto secondo le esigenze.

Quindi scattiamo molte fotografie, scansiono il maggior numero di campioni su libri o riviste di qualità, scarichiamo da internet quelle librerie gratuite che ogni tanto si trovano (www.newtek.com, il sito della Newtek, offre gratuitamente per gli utenti registrati molte immagini a buona risoluzione), masterizziamo i nostri CD con tutte le immagini ben ordinate secondo il soggetto rappresentato... Alla fine avremo materiale prezioso con cui riempire i nostri mondi 3D!