

## I plug-ins

Uno degli aspetti eccezionali di LW è costituito dai plug-ins.

Il plug-in è un frammento di codice che viene scritto da programmatori esperti dell'ambiente Lightwave per ampliare le possibilità offerte dal programma. In pratica il plug-in è una serie di comandi che sfruttano le funzioni interne a LW per ottenere risultati non ancora presenti nel programma o anche una nuova estensione del programma che offre opzioni e procedure completamente nuove.

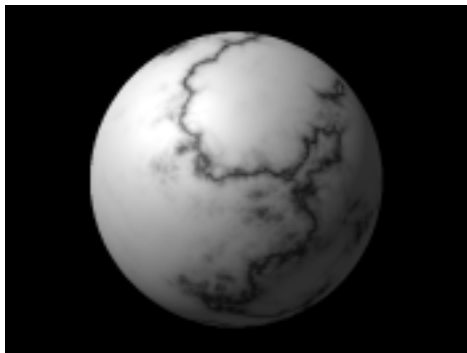
Fin qui niente di completamente nuovo e originale: famosi sono, per esempio, i plug-ins di 3D Studio Max (programma concorrente di LW) oppure i particolari plug-ins di Photoshop (i cosiddetti *filtri*). Oggi quasi ogni programma in ambiente Windows si avvale della tecnologia.

La cosa veramente straordinaria, pressoché unica dell'ambiente LW, è che questi frammenti di codice, funzioni o procedure che dir si voglia, sono programmabili all'interno del pacchetto software in dotazione standard. Ed ancora più eccezionale ed interessante per noi, utenti finali, è che la comunità di grafici, programmatori, creativi ed artisti LW sviluppa in continuazione queste preziose aggiunte e le mette molto spesso a disposizione *gratuitamente* in rete. Avete capito bene!

Provate a collegarvi col sito [www.flay.com](http://www.flay.com) e guardate nel motore di ricerca interno (spuntando solo le opzioni *plug.in*, *free* e *Lightwave 7.5*)...Centinaia di programmi gratuiti!!!

Ma a cosa servono questi così???

Poniamo che qualcuno trovi la procedura per creare, senza sfruttare immagini fotografiche, una buccia di mela... Oppure che si riesca a programmare una funzione che permette di mappare qualunque superficie con un bel materiale marmoreo... come quella della figura qui sotto:

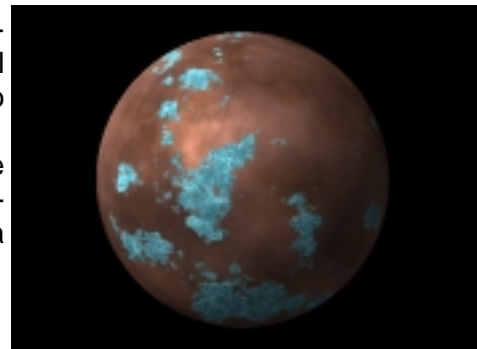


E' inutile oltre che dispendioso continuare a creare una cosa che, invece, può essere messa a disposizione di tutti.

I plug-in occupano ogni spazio disponibile in LW. Esistono, cioè, plug-in che operano come shaders di superficie, in altre parole come nuovi materiali che noi possiamo utilizzare per ogni nostra superficie. Famosa è la collezione di shades procedurali di Gordon Miller.

A volte collezioni particolarmente pregevoli di plug-in, o il particolare software che serve per produrli, vengono messi in vendita separatamente. E' il caso del preciosissimo DarkTree della *Darkling Simulations* ([www.darksim.com](http://www.darksim.com)) un software stand-alone che genera texture procedurali fantastiche per vari programmi 3D. LW gestisce le superfici prodotte con un plug-in gratuito che la Darksim mette a disposizione sul suo sito. Nel medesimo sito si possono trovare centinaia di presets di superfici prodotti con il software... e messi a disposizione *free* per il popolo degli utenti LW.

Guardate, ad esempio, la splendida superficie di rame ossidato prodotto con DarkTree e caricato in LW attraverso il gratuito plug-in Symbiont della stessa *Darkling Symulations*.



Altri plug-ins sono dedicati, per esempio, alla gestione delle dinamiche e delle animazioni. Altri ancora sono creati come veri e propri filtri per le immagini: come in Photoshop è possibile, per esempio, virare in seppia l'immagine renderizzata, o utilizzare per la produzione i valori cromatici delle più diffuse pellicole per negativi e dia. Ancora, si trovano plug-ins che offrono anche a LW la possibilità di gestire l'algoritmo di Sub Surface Scattering (cfr. pag 1.xx) o di creare cieli complessi e realistici (plug-ins del programmatore giapponese Ogo Hikari).

In sostanza... Cercate il plug-in che fa per voi. E' quasi certo che sia già in rete!

## Utilizzo del plug-in

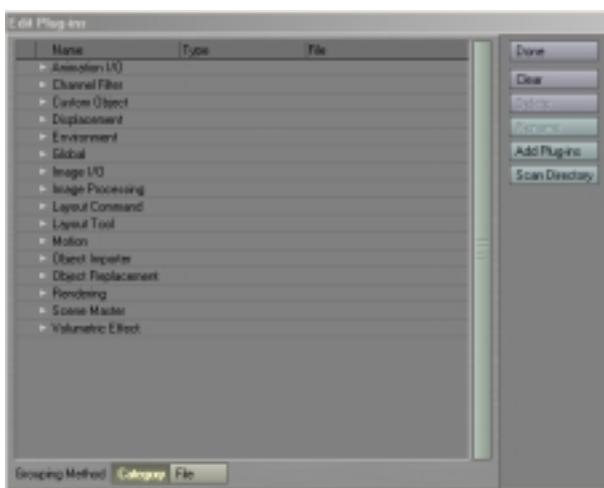
LW, diversamente da molti altri software, utilizza un modo del tutto trasparente per la gestione dei suoi plug-in.

Il codice viene diffuso attraverso la forma di files compilati con estensione \*.p. Essendo file di piccola dimensione, spesso vengono messi in rete senza neppure comprimerli. Altre volte, invece, si trovano compressi in un file \*.zip che contiene, oltre al plug-in, un file readme.txt che spiega molto sommariamente il suo utilizzo. Le notizie più importanti, tutorial o manuali veri e propri, vengono di solito messi in rete sul sito del programmatore.

Una volta scaricato il file sul vostro Hard Disk dovete trasferire il file \*.p in una directory a vostra scelta. Io consiglio di continuare ad utilizzare la directory di default dei plug-in. Se avete usato l'installazione standard di LW, avrete sul vostro Hard Disk una directory C:\Lightwave\plugins all'interno della quale troverete altre sottodirectory come Animate, Model, Texture, Utility ecc. In queste sottodirectory stanno i plugin compresi nel pacchetto all'atto dell'acquisto. Io consiglio di creare una directory C:\Lightwave\plugins\scaricati dove mettere, come avrete capito benissimo, i pezzi di software trovati nelle vostre escursioni in rete.

Il fatto di aver trasferito il software sull'hard disk, *non* integra automaticamente il plug-in col programma principale. LW ha bisogno che l'utente *gli* dica quale plugin vuole inglobare nel software. Per far ciò bisogna seguire questa semplice procedura:

Layout->plug-ins->edit plugins




Subito si attiva una finestra, questa volta non modale, che permette di gestire l'insieme dei plug-ins di LW. Se voi ricordate esattamente come si chiama il file del plug-in che avete scaricato, premete il pulsante *Add Plug-ins*, navigate verso la directory nel quale lo avete posto e selezionate il file (utilizzando il canonico file browser di gestione risorse). Dopo averlo caricato, LW vi informerà sul numero di plug-ins presenti nel file da voi scelto (in effetti più plug-ins possono essere contenuti in un solo file .p). A questo punto siete pronti per utilizzare il vostro nuovo piccolo (o grande!) software.

Attenzione: generalmente la versione di LW in vostro possesso vi costringe ad utilizzare plug-in compatibili! E' per questo motivo che sui siti di origine si trovano versioni dello stesso plug-in siglate \*6.p \*7.p \*7x.p: vuol dire che vanno bene, rispettivamente, per la versione 6, la 7 e tutte le versioni successive alla 7.

Facendo ciò vi accorgete immediatamente che sono già presenti, di default, una serie di plug-ins. Nella finestra di gestione tali plug-ins, come poi tutti quelli che aggiungeremo, sono riuniti di default come *category*, cioè come tipo.

Se selezioniamo *Grouping Method > File* possiamo vedere i plug-ins inseriti in ogni file.p.


Adesso vi resta solo da premere il pulsante *add plug-in*, se dovete aggiungere solo un file da voi conosciuto, o *scan directory*, se volete aggiungere molti plug-ins presenti all'interno di una directory. In genere questa operazione, molto comoda, viene fatta una sola volta, all'inizio magari della nostra esperienza con LW. Teniamo presente, peraltro, che si può comunque ricaricare per errore un plug-in già inserito nel sistema senza, per questo, creare confusione o crash improvvisi. Una volta che l'interfaccia LW<->Plug-ins è sistemata è però consigliabile evitare di fare cose stupide come, ad esempio, trasferire manualmente files, rinominarli, aggiornarli senza prima cancellare la versione precedente ecc. LW continua ad essere felice ed a funzionare normalmente... Voi un po' meno perché, magari, vi siete visti sparire la possibilità di usare il materiale preferito!

 Allo stesso modo, l'utilizzo di plug-ins in un progetto ci vincola a trasportare, assieme ai files, alle immagini ed ad ogni altro settaggio particolare, anche i plug-ins che, nel sistema destinazione, sono magari assenti.

Una volta caricato nel sistema il plug-in si pone il problema di sapere *dove* il sistema l'ha messo, in quale punto delle varie interfacce grafiche, cioè, il modulo è stato reso disponibile per l'utente.

... Dov'è?...

Non sempre è così semplice rispondere a questa domanda. Solo quando diventerete utenti esperti di LW troverete sempre ad occhi chiusi il vostro modulo appena caricato. Comunque, niente paura! Adesso vediamo due o tre regolette per non perdersi mai nella giungla dei plug-ins.

 bisogna sempre aver chiaro in testa a quale categoria di software appartiene il plug-in caricato


1) quando si scarica il file da internet bisogna sempre appuntarsi il tipo:  
 Shader = generatore di superfici  
 Motion = gestione del movimento degli oggetti in scena ecc.

Non ricordo il tipo!

Aprite *Layout->plug-ins->edit plugins*, selezionate *File*, cercate il nome del file e, in corrispondenza di questo, trovate il tipo del plug-in.

Name	Type	Display
BESM.p		0 LightWave/Plug
BESM	Shader (with UI)	

Il file BESM.p contiene un plug-in chiamato da LW BESM che è del tipo *shader (with UI)*. *Shader* significa *generatore di superfici* e *UI (User Interface)* ci comunica che questo modulo possiede un'interfaccia grafica, una finestra, cioè, che contiene parametri modificabili.

 bisogna sempre aver chiaro in testa dove è possibile trovare *quel* tipo di plug-in

2) Qui si va a logica, dopo avere conosciuto LW più da vicino. Uno shader di superficie sarà necessariamente da qualche parte all'interno dell'editor di superfici (*Surface editor*). Un modulo di tipo *motion modifier* sarà per forza all'interno del pannello che gestisce le specifiche di movimento (*Layout->Items->motion options->IK and modifiers->add modifiers*). LW è logico e razionale...

Non conosco queste cose!

No problem! Andate a vedere nell'appendice X e troverete tutte le corrispondenze tra tipi di plug-ins e le posizioni che hanno all'interno di LW.

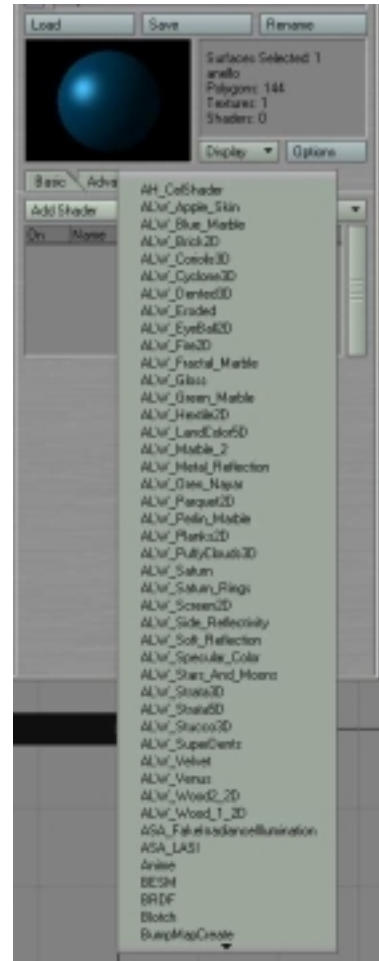


Qui a sinistra, per esemplificare, troviamo il nutrito menù con cui si sceglie il *motion modifier* adatto al nostro scopo. Alcuni di essi sono presenti di default nel pacchetto LW 7.5c, altri sono stati aggiunti in seguito dall'autore.

A destra, invece, vediamo l'enorme quantità di *shaders* (quelli rappresentati sono un quarto di quelli effettivamente presenti) immessi nel sistema dopo qualche lettura, raccolta di informazioni e qualche visita in rete. Sono tutti gratuiti e preziosi!

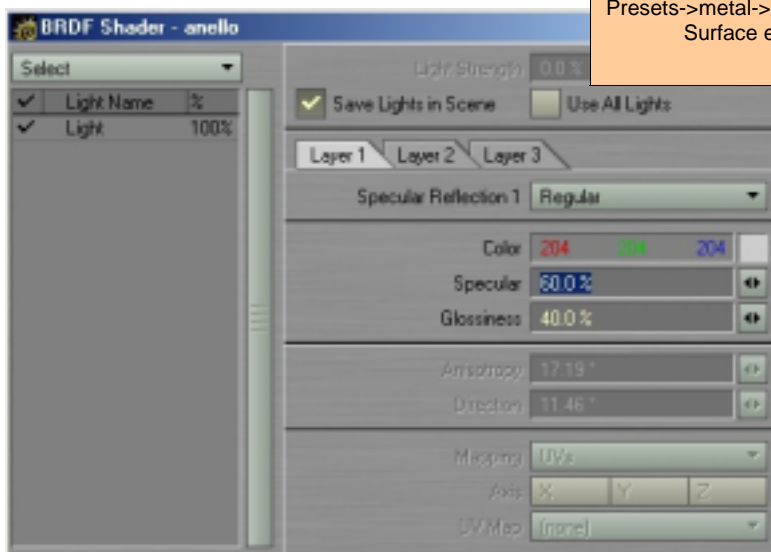
Una volta selezionato con un click il modulo che ci serve esso apparirà nella finestra corrispondente, potrà essere attivato e disattivato, copiato, cancellato e eseguito. In quest'ultimo caso si intende l'apertura della sua finestra (UI) per procedere al settaggio dei vari parametri modificabili.

Proviamo, per esempio, ad andare in Layout. Puliamo la scena, carichiamo un semplice oggetto e applichiamo una superficie metallica. Poi attiviamo il plug-in (o lo shader, se vogliamo chiamarlo così) BRDF che gestisce le riflessioni anisotropiche tipiche dei metalli.



lo shader, se vogliamo chiamarlo così) BRDF che gestisce le riflessioni anisotropiche tipiche dei metalli.

File->Clear Scene  
 Items->Add->Load object CD:\objects\ball.lwo  
 Surface editor  
 Presets->metal->stainless (un semplice materiale tipo acciaio)  
 Surface editor->Shaders->add shader->BRDF  
 Msdoppioclick su BRDF

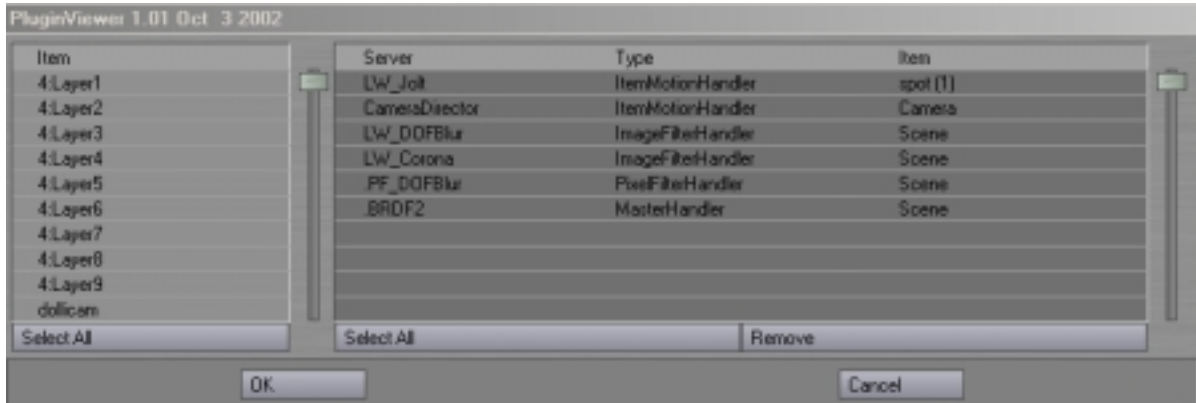


Questa a sinistra è la finestra attivata la *User Interface* del plug-in. Possiamo impostare 3 layer di riflessioni (regular, anisotropic, anisotropic II), i valori della riflessione, dell'angolo di anisotropia, il colore dell'area interessata dalla riflessione ed altro ancora. Una volta finito basta chiudere la finestra per aver subito in preview la superficie modificata.

Naturalmente questo porta subito un problema: ogni plug-in ha le sue regole e l'utente deve sapere ciò che il modulo fa per poterlo sfruttare... In rete ci sono decine di pagine di manuali, spesso solo in inglese, peraltro.

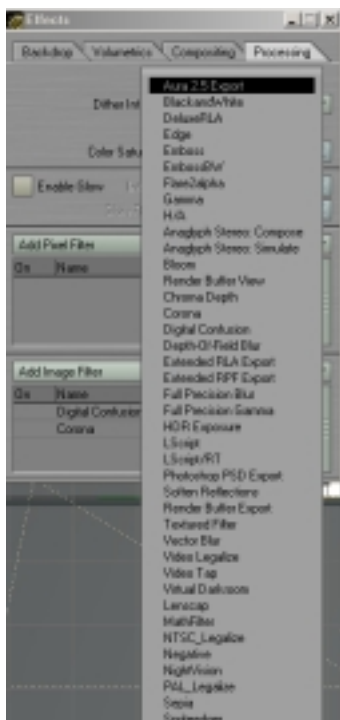
Come si diceva di plug-ins ce ne sono molti. Molto utili sono quelli che aiutano a gestire la complessità della scena.

Un plug-in di grande utilità è *plug-ins viewer* (PluginViewer.p) che, come dice il nome, permette di vedere tutti i plug-ins presenti nella scena (è un mio vizio quello di tenere tutto sotto controllo nel computer...):



Nell'esempio sopra riportato vediamo che c'è un plug-in, chiamato *LW\_Jolt*, di tipo *ItemMotionHandler* (plug-in modificatore del movimento degli oggetti) attribuito all'oggetto Spot(1), la seconda luce spot presente in scena! Gli altri plug-ins presenti in questa scena si possono spiegare usando la stessa tecnica (*LW\_DOFBlur* è, per esempio, un plug-in di tipo *ImageFilter*, filtro di postproduzione come quelli presenti in Photoshop, che applica un *blurring* – una sfocatura – all'immagine, basandosi sul *Depth Of Field*, cioè sulla profondità di campo imposta all'ottica della camera).

Ai plug-ins di tipo *ImageFilter* si accede premendo...



CTRL+F8->add Image Filter

... e qui a destra si vede il risultato: una nutrita serie di filtri che mettono a nostra disposizione importantissimi strumenti. Solo come esempio: un paio di modi per applicare l'effetto di **profondità di campo** (lo sfuocato tipico della fotografia ottica e manuale dei bei tempi andati), una efficiente **camera oscura** nella quale possiamo impostare i valori cromatici, la sensibilità e la grana della pellicola virtuale che stiamo usando – prendendo come parametri quelli delle pellicole professionali più diffuse in commercio, vari **filtri colore** come l'effetto **seppia**, il **bianco e nero** e **negativo**, l'effetto **notturno**, gli effetti **corona** e **bloom** (i punti di luce massima in una scena illuminata con luce naturale), e ancora l'esposizione **HDR**, per esporre immagini da utilizzare poi come fonti di illuminazione di tipo High Dynamic Range (vedi cap.1-l'illuminazione).

LW è un programma talmente complesso che ancora oggi mi capita, sempre meno, per fortuna, di scoprire meandri mai visitati del flusso operativo. Quando mi trovo di fronte ad una novità sono sempre curioso e aperto: non c'è nulla in LW che non si possa imparare.....e se non è documentato da alcuna parte (nemmeno sul manuale di casa Newtek) c'è sempre il sistema del prova e riprova... o quello di iscriversi al gruppo di discussione del sito [www.newtek.com](http://www.newtek.com) dove basta presentare il nostro problema. Subito altri utenti più esperti o più fortunati o più informati di noi ci risponderanno con informazioni notevoli.