



**L'illuminazione in Kray** le impostazioni di base  
di Matt Vanni

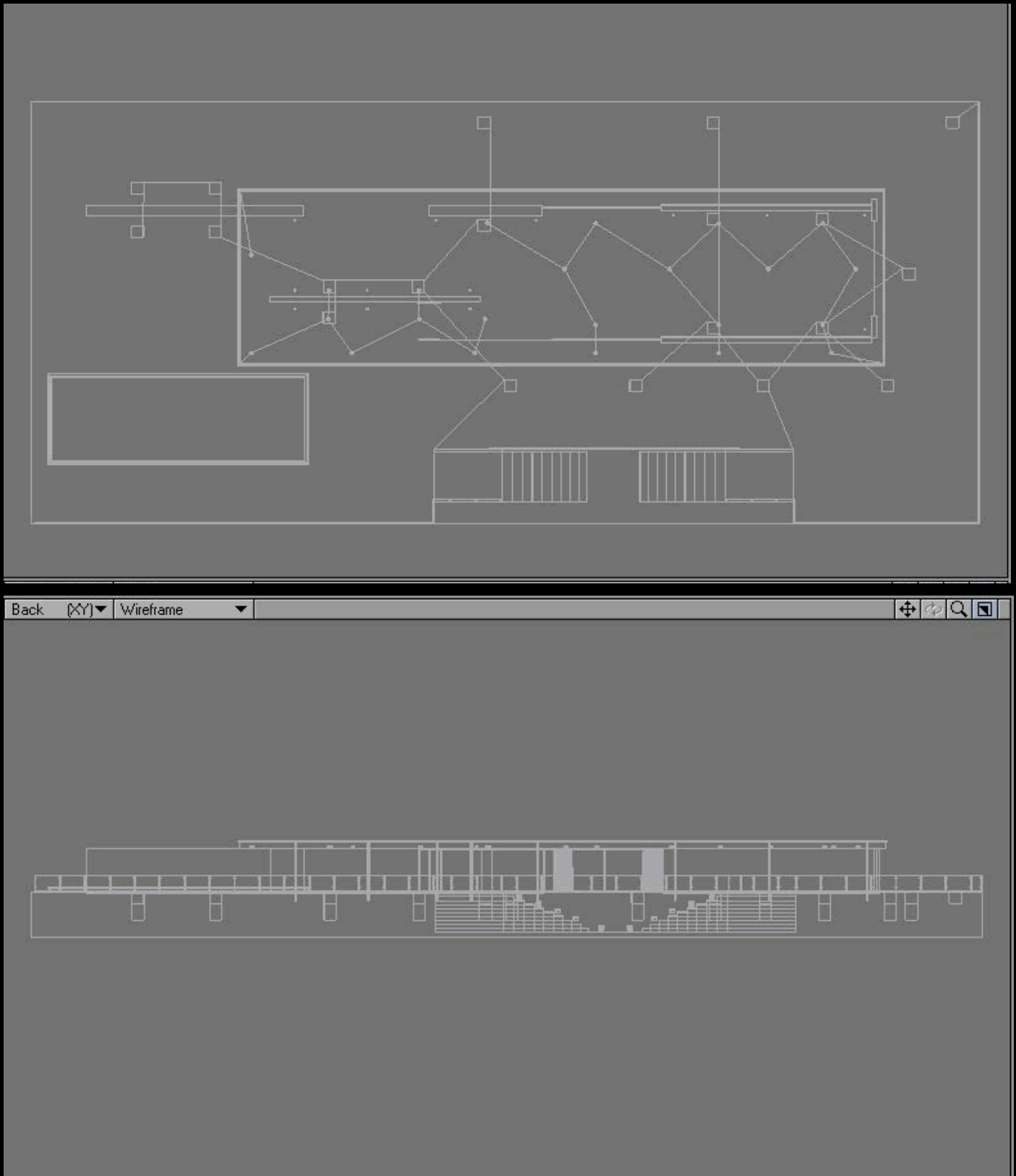


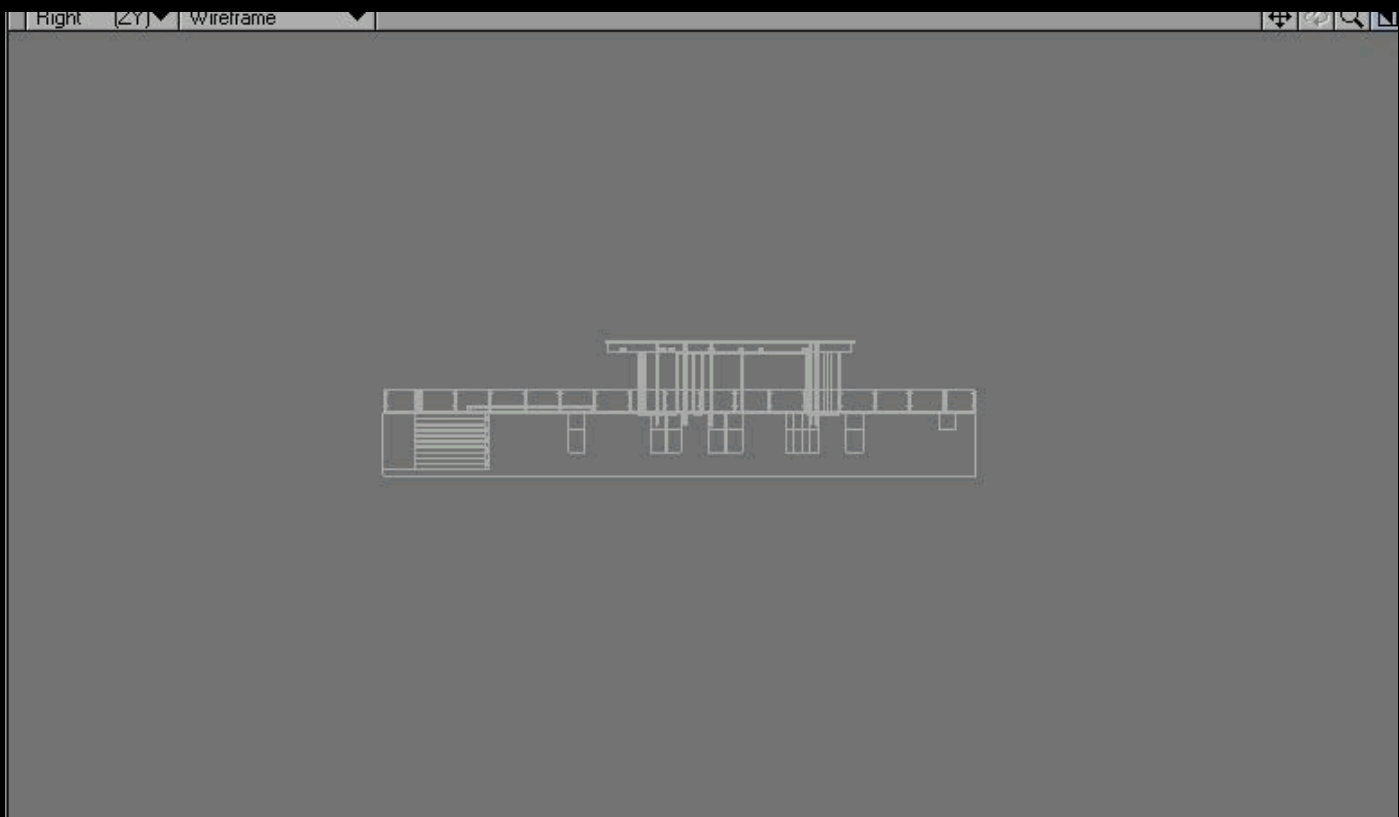
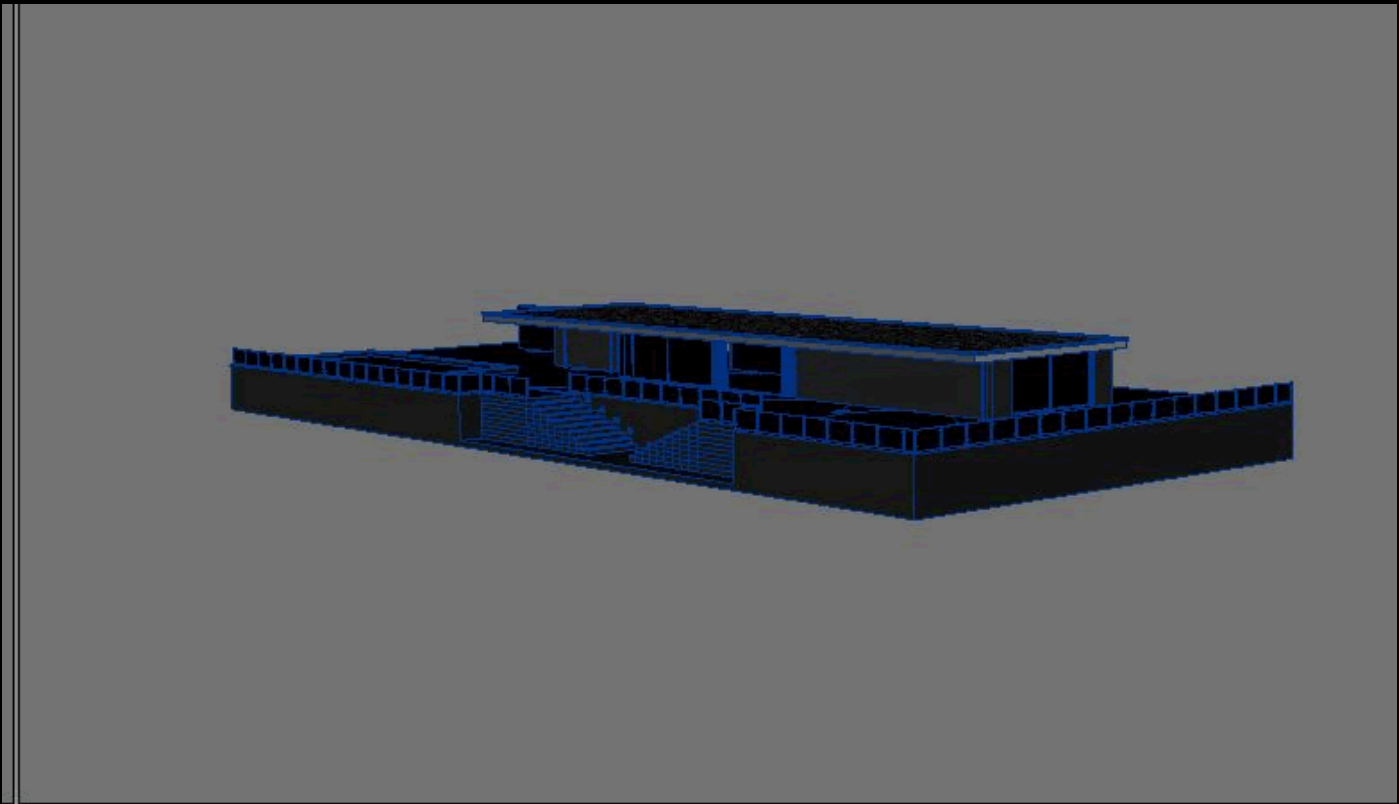
con la collaborazione di Gianandrea Ghirri - LAV

In questo tutorial basico tenterò di spiegare come impostare una illuminazione basica principalmente per esterni.

Prima di tutto vorrei chiarire che non esiste un'unica impostazione per l'illuminazione in Kray; questa è solamente una guida di riferimento: in ogni software, lavorando giorno dopo giorno, s'imparano vari modi di impostare e raffinare il workflow.

In questo render ho usato un riferimento trovato online, [un edificio molto semplice](#) geometricamente parlando, basato su una conosciuta opera architettonica (fig1)





Per quel che mi riguarda, io preferisco **segmentare l'edificio in modeler**, nei layer componenti, soffitto, pavimento, pareti, e vetri (fig2)

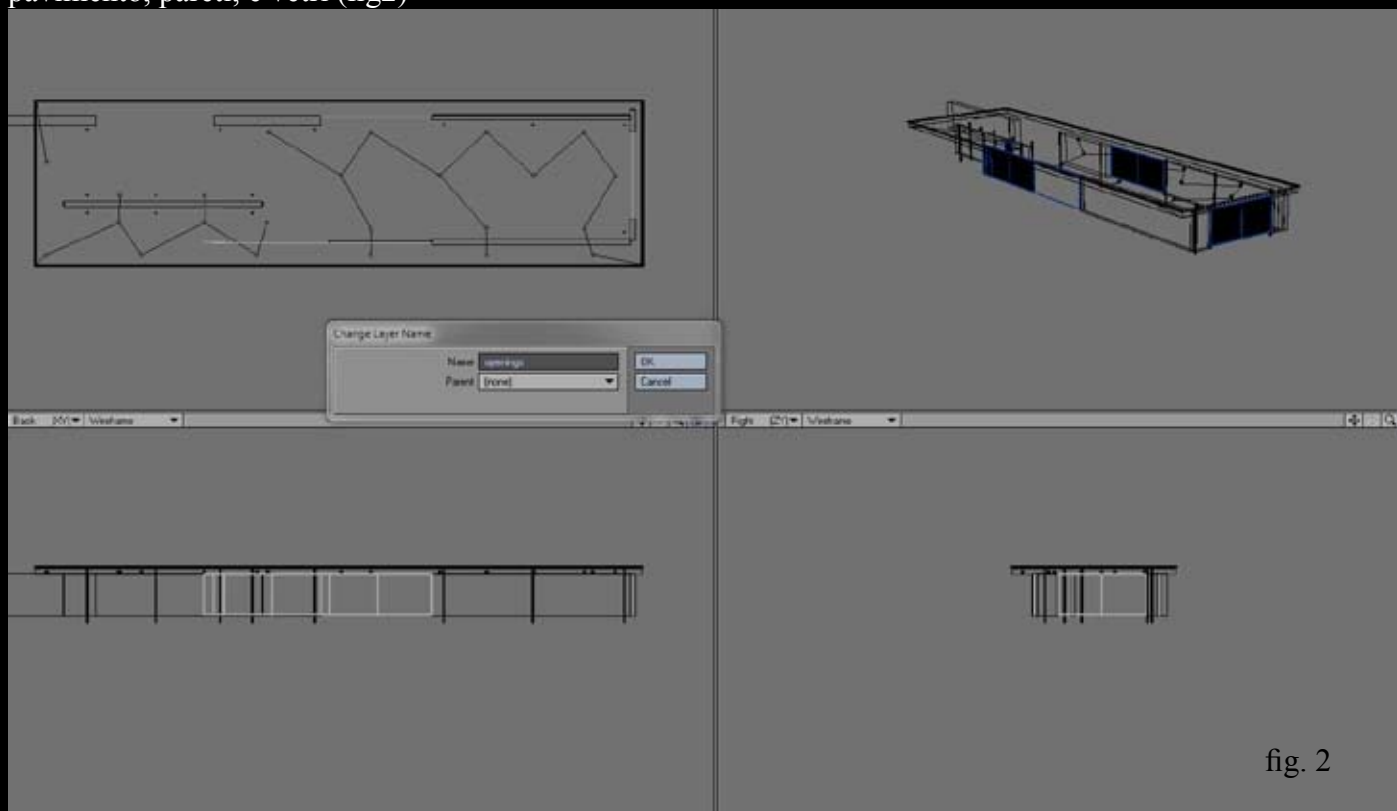


fig. 2

Il vantaggio di separare perlomeno i vetri quando si lavora con kray è questo, si possono **rimuovere le ombre** solamente sul vetro per risparmiare i tempi di render, ed anche si possono rimuovere questi oggetti o layers dal calcolo della GI (fig2b)



fig. 2b

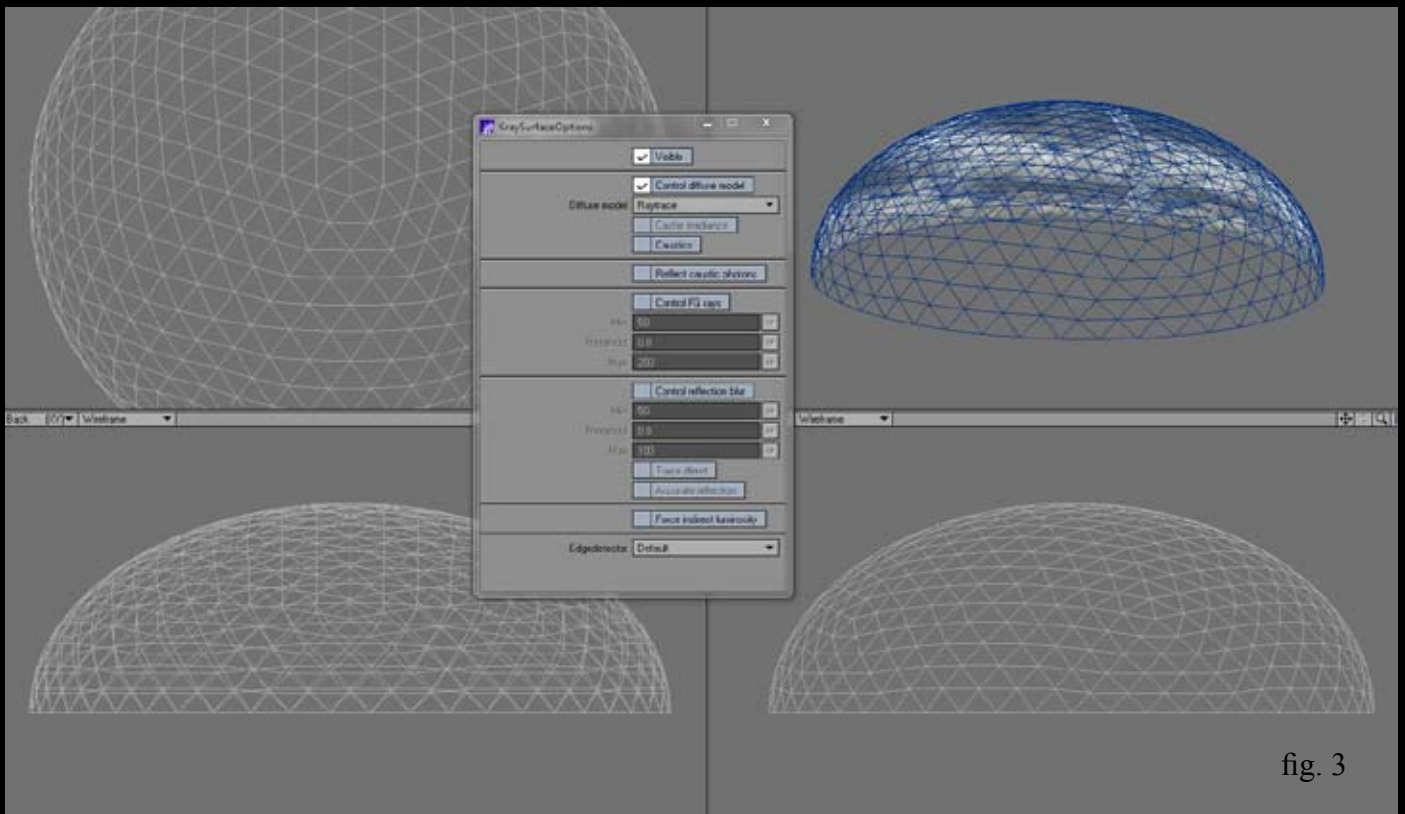


fig. 3

Dopo avere caricato l'oggetto in layout si può mettere un *dome* qualsiasi per illuminare una scena o solamente come proiezione di un'immagine che rappresenti il cielo. In questo caso ho usato la semifera *dome* solo come proiezione, dato che questa volta ho preferito provare (e sono contento del risultato) il nuovo plugin di kray, *physical sky*

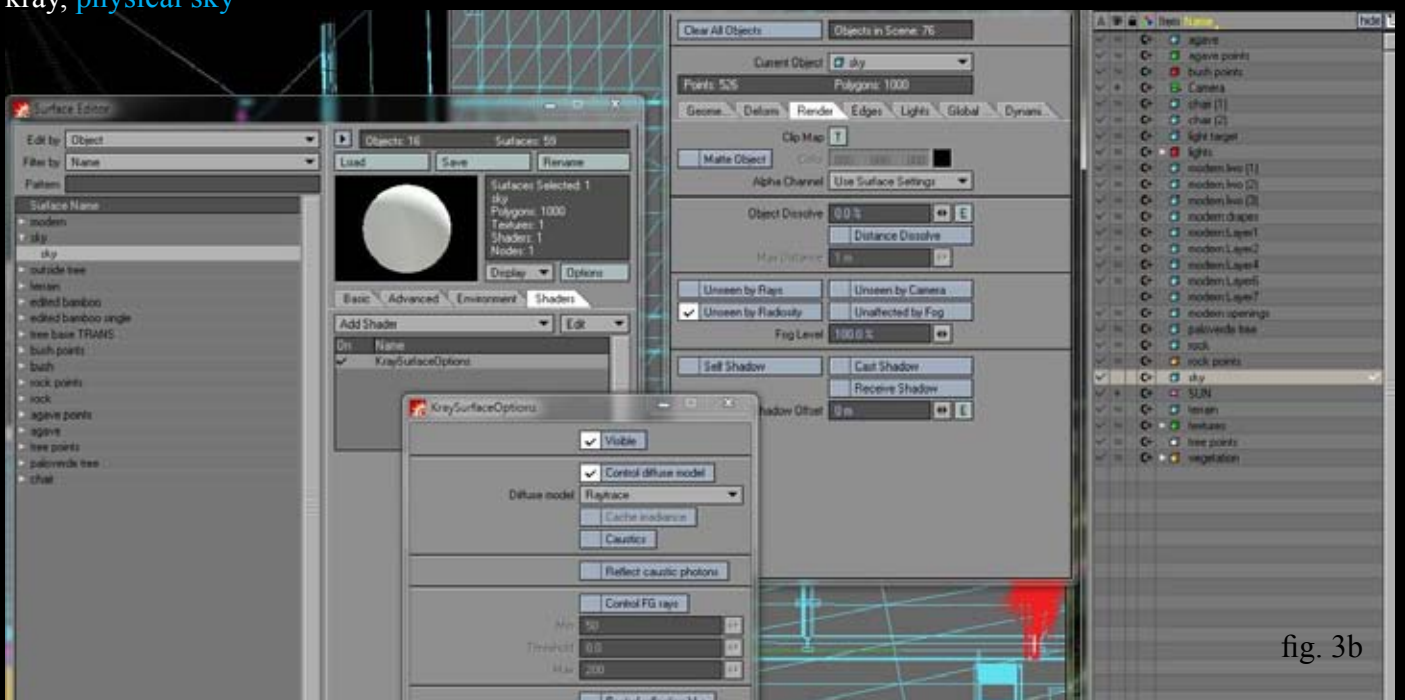


fig. 3b

Il cielo non lancerà ombre sulla scena, per questo ho rimosso **Self Shadows**, **Cast Shadows** e **Receive Shadows**. Questo è utile se si usa una luce lontana come Sole, altrimenti questa luce non illuminerà la scena, perché ovviamente la sua luminosità sarà bloccata dall'oggetto 'cielo'. Per risparmiare tempo di render il cielo non è stato calcolato nella Global Illumination, per cui ho usato l'add-on **KraySurfaceOptions** pannello surface.

In questo pannello ho attivato **Raytrace** come diffuse model invece di **photon mapping**: questo significa che sebbene il complesso del render sarà computato come photon mapping, il cielo (ed altro, come vedremo dopo) saranno computati con il semplice raytracing, dato che gli oggetti esclusi non apportionerebbero alcuna miglioria nella resa finale (almeno in questo render esterno, dato che la cosa dipende della qualità della scena)

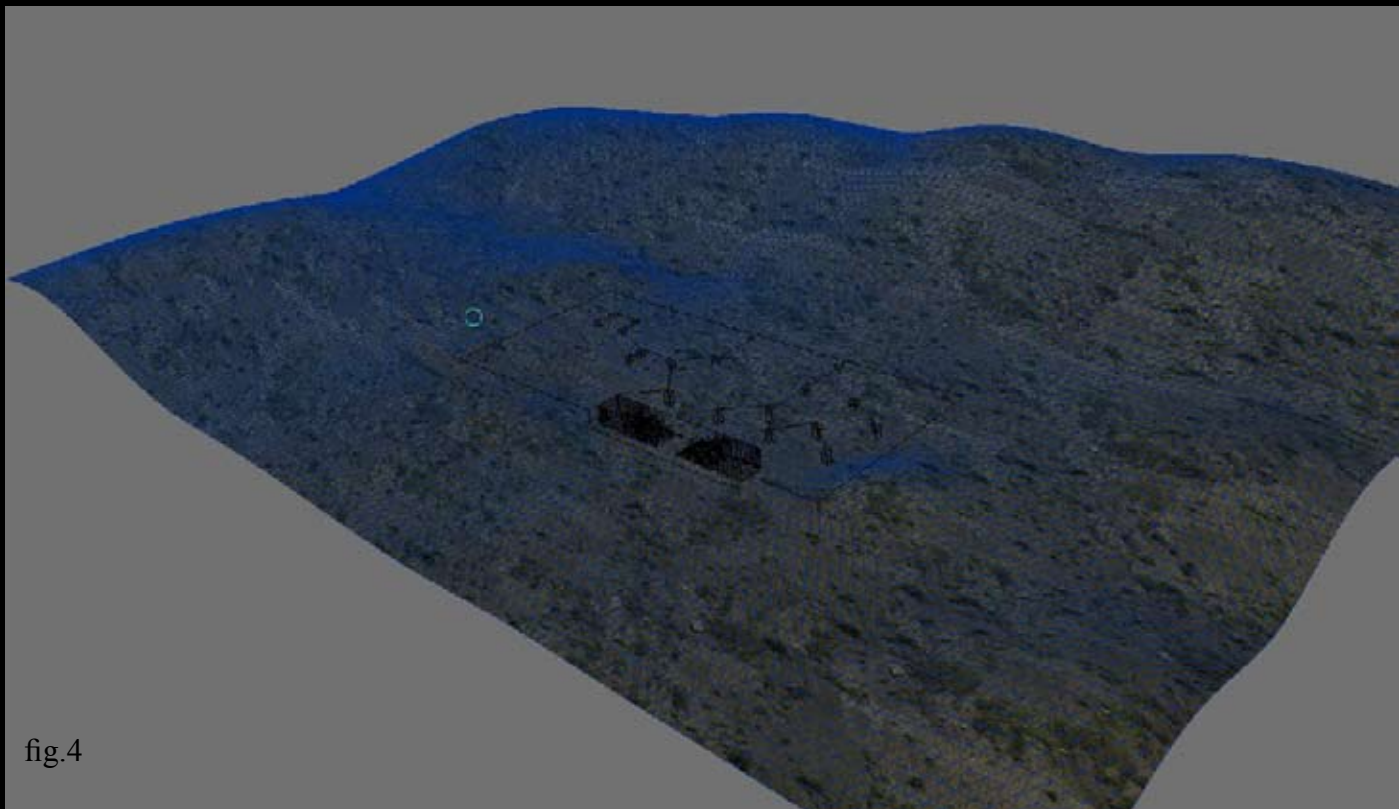


fig.4

**Il terreno in questo caso è abbastanza semplice**, ho cominciato con un rettangolo suddiviso e triangolato, ho usato il layer del pavimento dell'edificio solo come riferimento, poi ho modellato le collinette col comando Dragnet, la texture è una mappa a buona risoluzione di un deserto. La mappa è stata poi estesa con Imagesynth (distribuito in bundle con Modo). (fig4)

Per l'illuminazione ho usato il metodo raccomandato sin dalle prime versioni di kray, prima della comparsa del plugin Physical Sky.

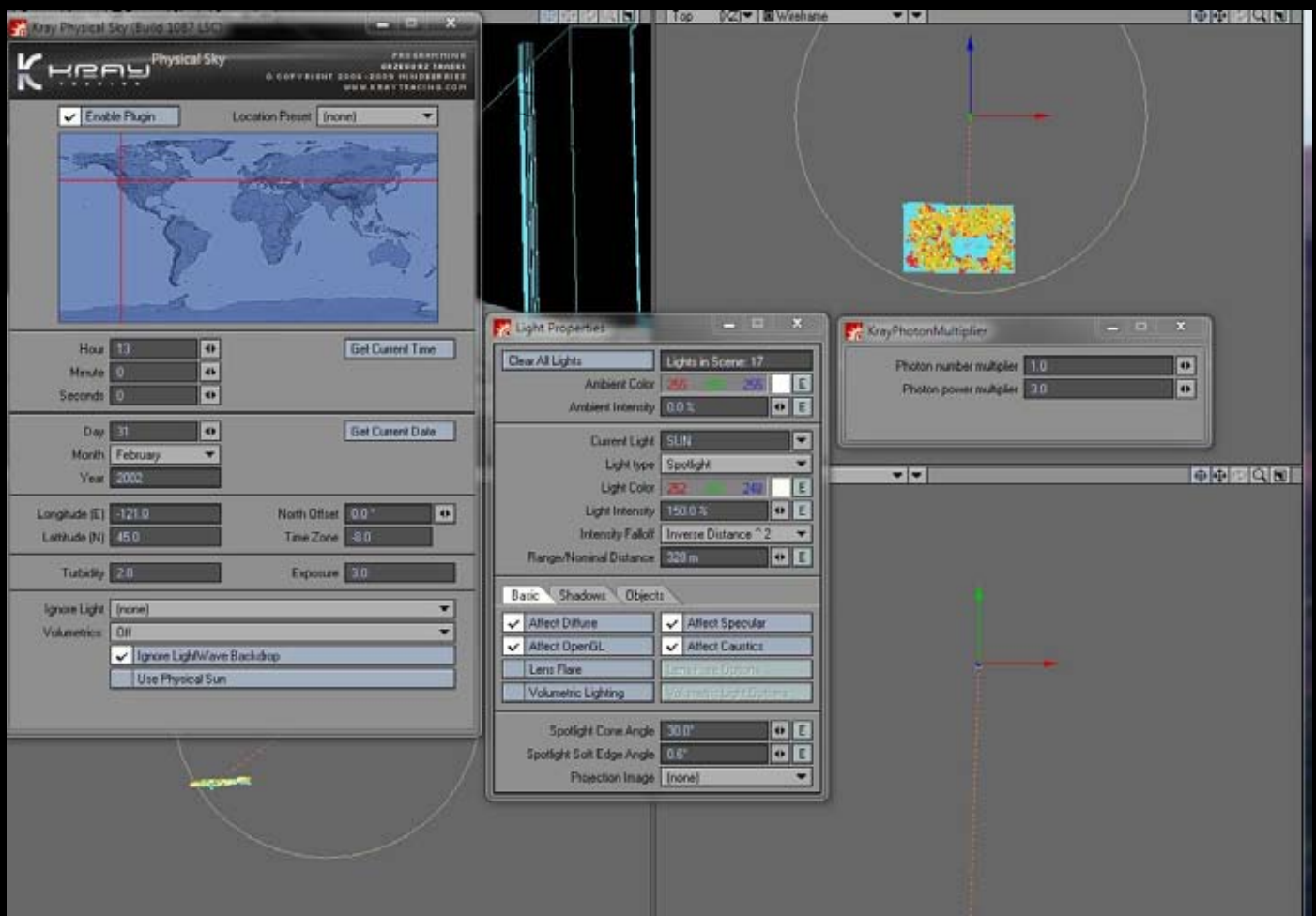
Il Sole è uno spotlight, con falloff tipo  $\text{inverse distance}^2$  (questo tipo di falloff è sempre raccomandato con Kray, perchè simula particolarmente bene la decadenza della luce solare, come avviene nella vita reale).

Allo spotlight viene attivato un **photon multiplier**, col valore 3: è questo un plugin di servizio in Kray, attivabile attraverso il pannello Motion modifier (selezionare la luce che rappresenta il Sole, premere 'm' e, nel pannello attivato, selezionare dal menu a tendina Motion modifier la linea Photon multiplier).

Questo passaggio aumenta la potenza in Global Illumination dei fotoni senza *bruciare* i livelli di bianchi (questa particolare sezione del lavoro è influenzata anche dal settaggio del *tonemapping*, fatto direttamente nel pannello render di Kray).

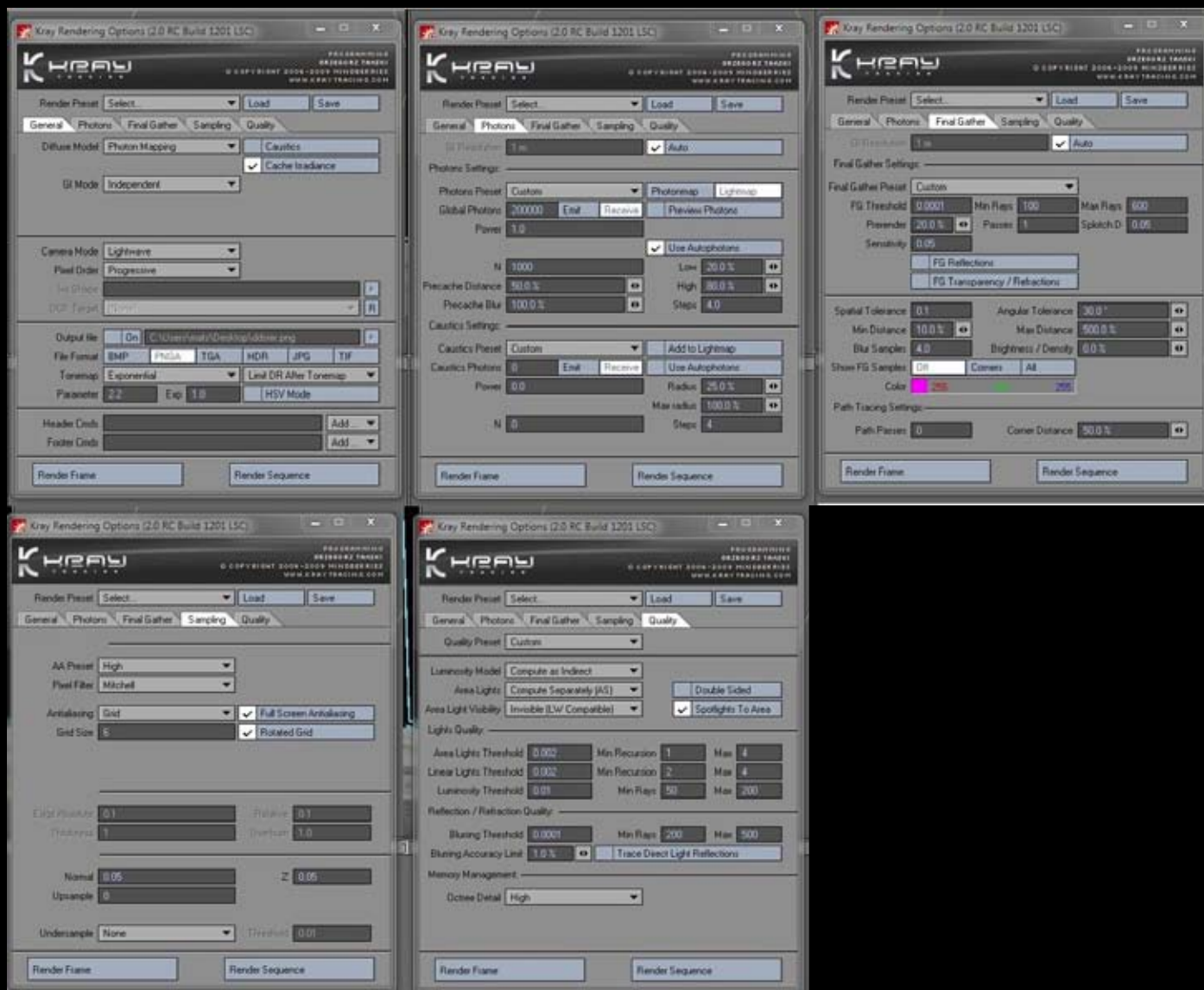
Uno dei molti vantaggi nell'usare **Physical\_sky** è che si possono immettere le coordinate reali della nostra location, in longitudine e latitudine, come pure scegliere una delle numerose località campione presenti nel menù a tendina. Questo è particolarmente utile quando è necessario simulare le condizioni reali dell'illuminazione. Allo stesso modo possono essere inserite data e ora.

In questo render ho scelto di usare come Sole la luce spot principale, abbinandole soltanto il plug Physical sky per produrre la luminosità principale d'ambiente. Per ottenere ciò è bastato deselezionare l'opzione Use Physical sun nel pannello. Diversamente avremmo avuto una resa che sarebbe stata influenzata da due luci, lo spotlight Sole e il Sole realistico (Physical sun) implicito in Physical sky.



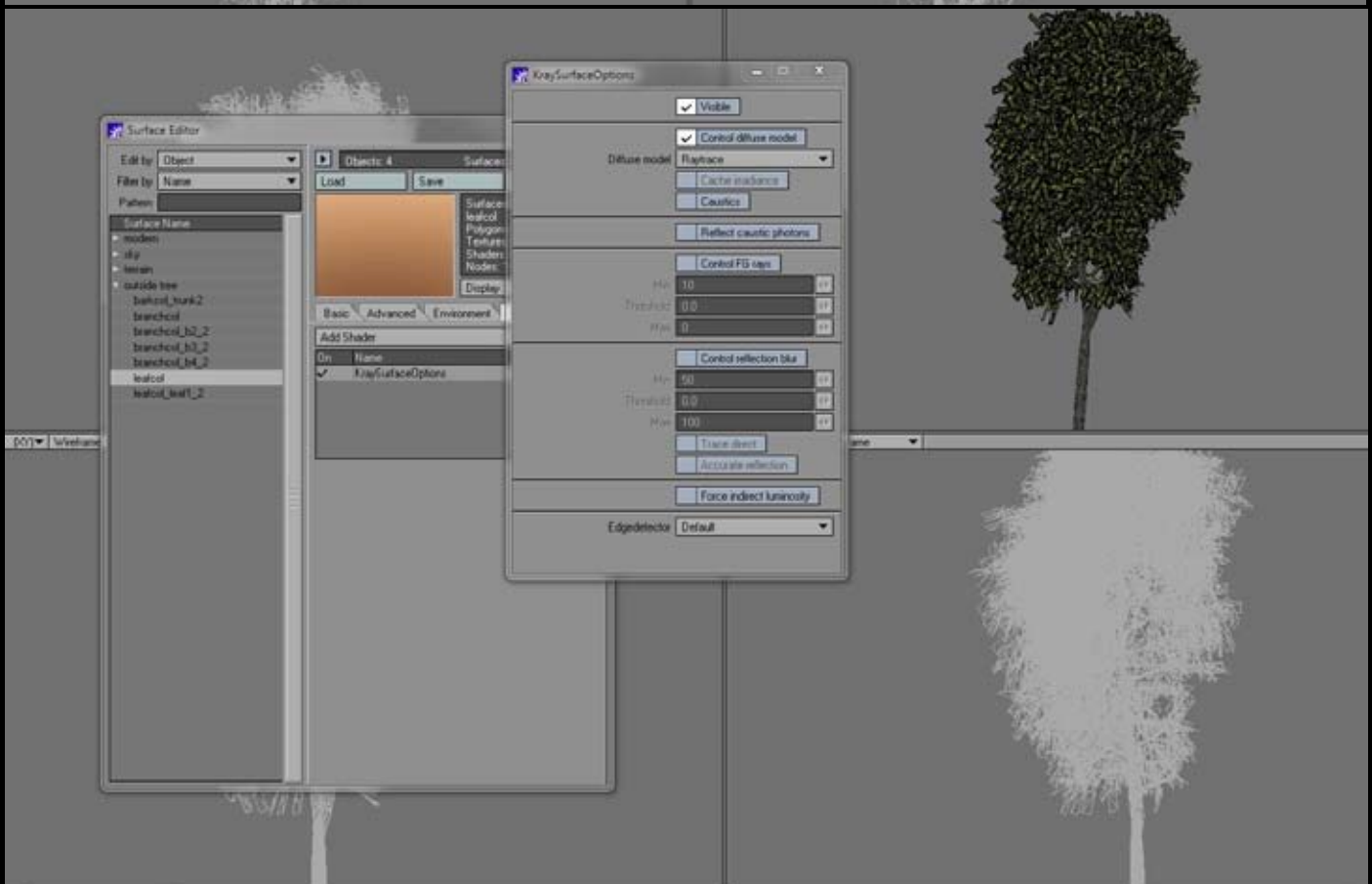
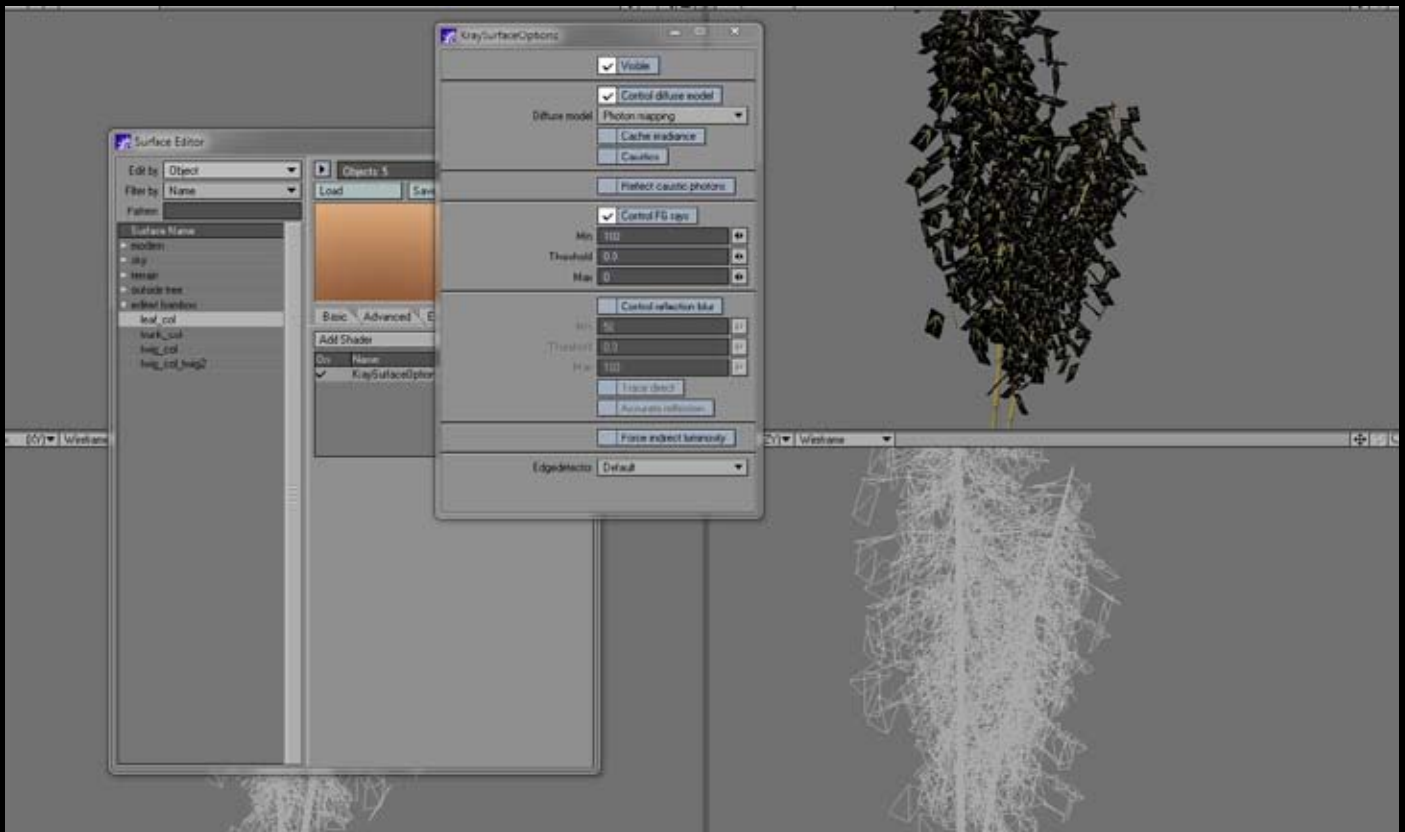
Questi sono i settings usati nel render. Sono partito da settings di bassa qualità (LOW), e aumentando poi via via qualche valore base.

Le specifiche dei comandi si possono trovare a questo link: [http://www.kraytracing.com/wiki/Kray\\_User\\_Manual/it](http://www.kraytracing.com/wiki/Kray_User_Manual/it) (tradotto da un utente di LWITA, Talats - Roberto Cifra).



Le piante, gli alberi e gli altri elementi della vegetazione sono complessi da calcolare per il motore di Global Illumination, ma esiste anche qui una via per risparmiare il tempo di render, (qualche volta al 50% se la scena presenta una vegetazione molto densa).

Usando lo shader **KraySurfaceOptions** (nel pannello shaders della finestra Surface options) si può cambiare il metodo di computazione al livello di superficie. Ad esempio, le foglie possono essere disegnate come Raytracing semplice, o attraverso il Photon mapping, ma riducendo il numero di raggi. Abitualmente si usano, in base al tipo di scena, i valori di min 10 e max 0. Se una pianta ha un'importanza particolare, o è molto vicina al punto di vista, o ancora se si vogliono più particolari, si possono sempre aumentare i valori di cui sopra.



Per il paesaggio che circonda la costruzione, ho usato il tool commerciale **Veggipaint**, che può essere usato in collegamento con Layout o Modeler, ma che funziona anche come software standalone. Il vantaggio che presenta il software, nel piazzare ounti sopra una geometria (in questo caso il terreno su cui sorge la casa), è che su possono disegnare semplicemente *spruzzandoli* come vernice sulla superficie.

Questi punti poi vengono interpretati come **istanze** in Kray (grazie al plugin della serie Objects, **Krayinstances**, fornito col pacchetto) semplicemente selezionando gli oggetti che devono essere duplicati in scena. In questo caso ho usato quattro layers di punti, per usarli come riferimento per le instances di Kray: uno per qualche albero del deserto, uno per le rocce, un altro per le agavi e, in ultimo, uno per i cespugli. Come accade anche nel celebre **HDInstance**, in Kray le istanze possono essere ruotate e scalate con un semplice click del mouse.



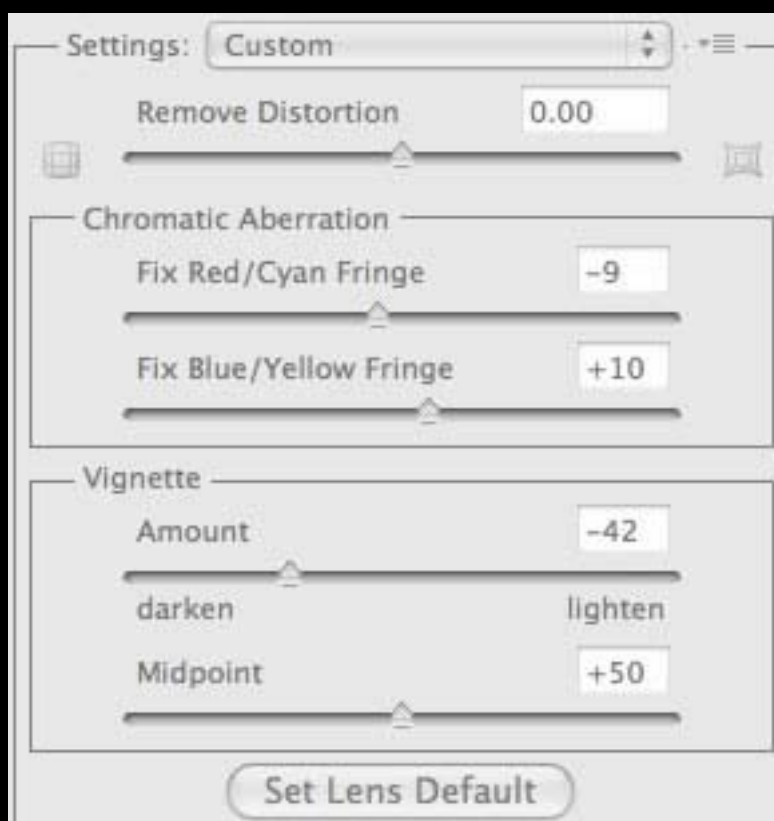
Dopo il render di base prodotto da Kray, generalmente applico una serie di **tecniche di aggiustamento per la correzione del colore** e per simulare l'aberrazione e la distorsione provocate dalle lenti di una fotocamera reale. C'è un modo per aggiungere in Kray direttamente le aberrazioni cromatiche attraverso l'uso dei comandi a tendina sul pannello principale del programma, ma io preferisco farlo in post (usando Photoshop della Adobe), dal momento che in kray non c'è alcuna possibilità di annullare l'effetto senza procedere ad un nuovo render dell'intera immagine.

Come primo passo inizio a modulare i valori del gamma e dell'esposizione secondo le necessità, usando il menu **Immagine > Regolazioni > Esposizione**, per ridurre di poco i livelli: di solito rendo un po' più luminoso il valore gamma mentre diminuisco l'offset di un briciolo, normalmente un  $-0,0020$  o poco più, a seconda dell'immagine da elaborare.



In seguito uso il filtro **Correzione lente**, che può essere trovato nel menu **Filtro > Distorsione > Correzione lente**. Il filtro è normalmente inteso per la correzione dell'aberrazione cromatica e della deformazione prodotta dalla lente. In questo caso, come spesso viene fatto da molti artisti della CGI, preferiamo inserire alcuni di questi difetti delle lenti reali nella nostra immagine digitale: spostando i valori del **rosso/ciano** e del **blu/giallo**, come l'ammontare della **vignettatura** per simulare la radianza che normalmente avviene nelle lenti reali in particolari condizioni di luce, soprattutto ai bordi dell'immagine, che viene lievemente scurita.

Al termine aggiungo un passo di contrasto.





Un ultimo effetto che uso di tanto in tanto per simulare la diffusione della luce è il [Filtro>Distorsione>Bagliore diffuso](#). Stando attenti a che il colore di background sia settato a bianco, possiamo usare questo filtro per simulare aree dell'immagine sovraesposte, con la produzione di un effetto di aloni luminoso o di luce diffusa.



Penso che questo sia tutto, per ora. Anche se questo è di certo un modo personale di lavoro, io sto sempre testando nuove cose e nuove procedure. Nulla è scritto nella pietra: tutto dipende dalla curiosità e dalla sperimentazione continua.

Buona fortuna, quindi, con **Kray!**

Matt Vanni