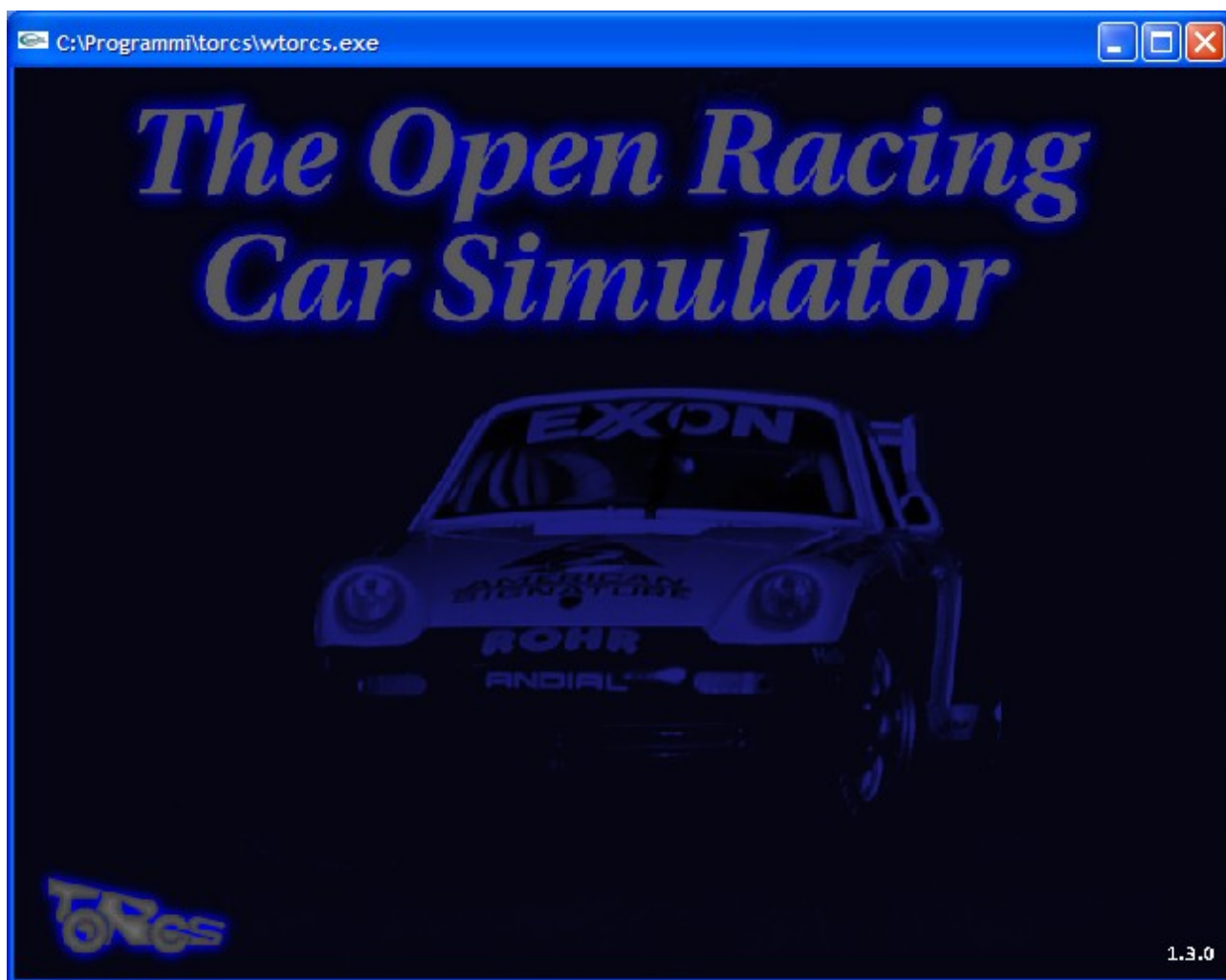




torcs.sourceforge.net

presenta:

TORCS



GUIDA RAPIDA DI RIFERIMENTO

realizzazione a cura della redazione di

www.nontipago.it

portale di servizi e programmi gratuiti selezionati e illustrati



Indice degli argomenti:

# Introduzione e requisiti	pag. 3
# La configurazione dei giocatori	pag. 4
# La configurazione grafica	pag. 8
# Le opzioni audio	pag. 11
# I circuiti	pag. 12
# Suggerimenti	pag. 16
# Domande frequenti	pag. 17
# Gli autori del progetto	pag. 27

Introduzione e requisiti

TORCS acronimo di The Open Racing Car Simulator è un simulatore di corse automobilistiche multiplatforma che funziona sui sistemi operativi basati su Linux (x86, AMD64 e PPC), FreeBSD, MacOSX e Windows.

Il gioco è gratuito e il codice sorgente di TORCS è autorizzato sotto licenza GPL ed è quindi liberamente fruibile, inoltre nel sito ufficiale esiste la possibilità di collaborare attivamente al progetto.

In particolare si possono sviluppare piloti virtuali definiti robot: in pratica una porzione di codice di programma che pilota un'auto (tutti gli avversari non controllati dai giocatori sono in effetti robot). Nell'ambito di TORCS è possibile sviluppare robot e partecipare con essi a veri e propri campionati disponibili in rete.

TORCS è stato realizzato e viene sviluppato costantemente con il contributo di molti programmatori dall'idea iniziale di Eric Espié, Christophe Guionneau, Currently Bernhard Wymann e Christos Dimitrakakis. L'elenco completo dei partecipanti al progetto è disponibile nell'ultima sezione di questa guida e nel sito ufficiale.

TORCS dovremo giocarlo un po', prendere confidenza con il sistema di controllo delle auto e comprendere quale auto si adatta di più alle nostre caratteristiche di guida per poi cimentarci in campionati, ma potremo ottenere da questa simulazione ottime soddisfazioni perché l'involuppo delle caratteristiche tiene conto di molti interessanti fattori sia di tipo cinematografico che dinamico. Aderenza delle gomme, accelerazioni assiali, comportamento delle masse non sospese, aerodinamica, e molto altro rendono il gioco realistico e divertente.

I requisiti non sono eccessivi e il gioco può essere utilizzato anche su un portatile, ma naturalmente sarà bene accordare le opzioni grafiche con la potenza del sistema sul quale intendiamo installarlo.

Requisiti hardware per il gioco:

Minimi:

550MHz CPU, 128MB RAM, OpenGL 1.3 scheda grafica compatibile con 32 MB RAM.

Suggeriti:

800MHz CPU, 256MB RAM, OpenGL 1.3 scheda grafica compatibile con 64 MB RAM.

I requisiti software dipendono dal sistema operativo utilizzato:

Per tutti OpenGL/DRI drivers recenti, funzionanti e correttamente configurati e OpenAL (per una qualità del suono migliore potete prelevare OpenAL dal sito ufficiale TORCS).

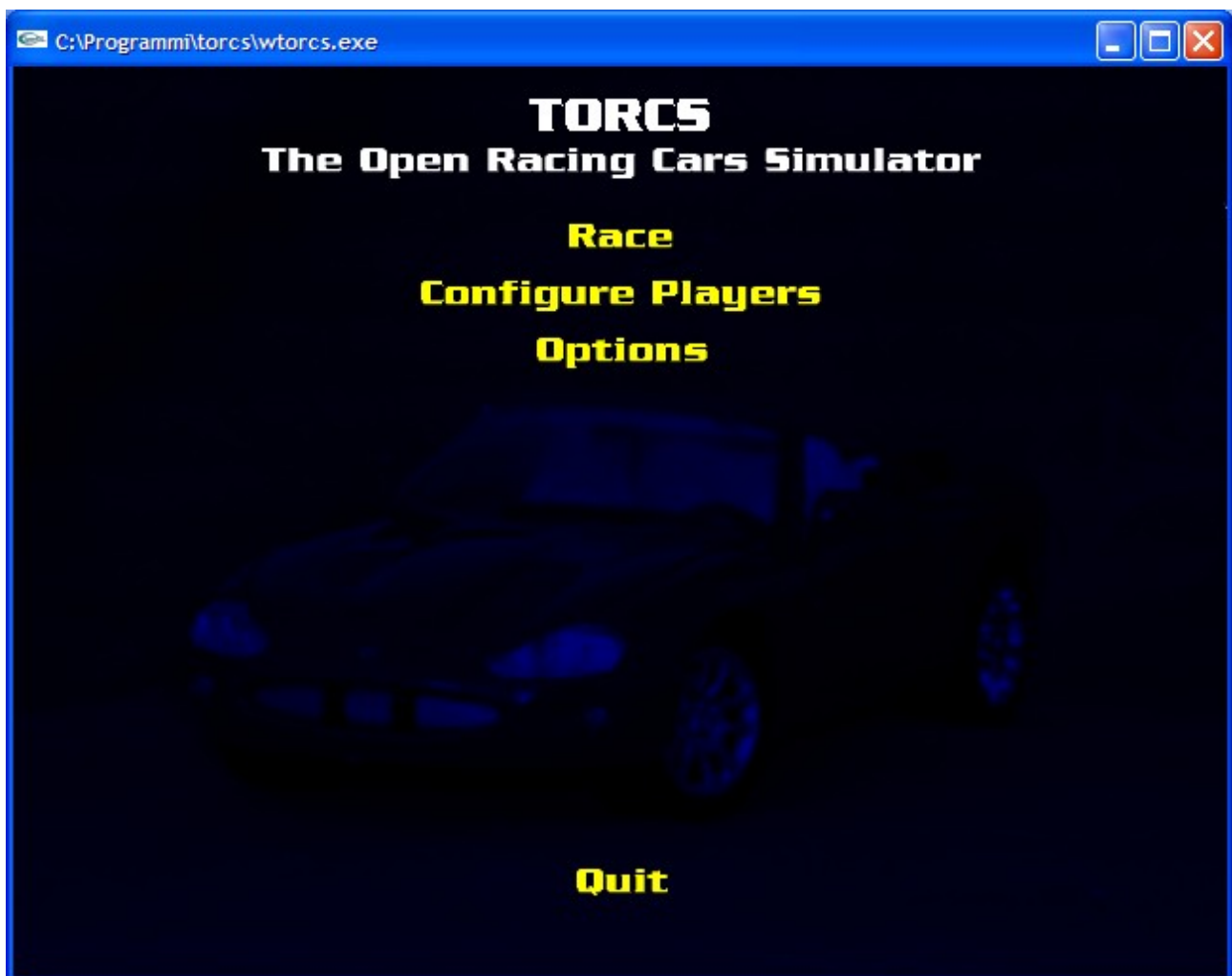
In relazione al sistema operativo utilizzato avete inoltre bisogno o no di altre librerie. Per evitare problemi preferite [FreeGLUT](#) a GLUT. Accertatevi di utilizzare pplib version 1.8.3. Per AMD64 necessitate il settaggio "-fPIC" del compilatore durante la compilazione di pplib (export CFLAGS="-fPIC", export CPPFLAGS="-fPIC", export CXXFLAGS="-fPIC").

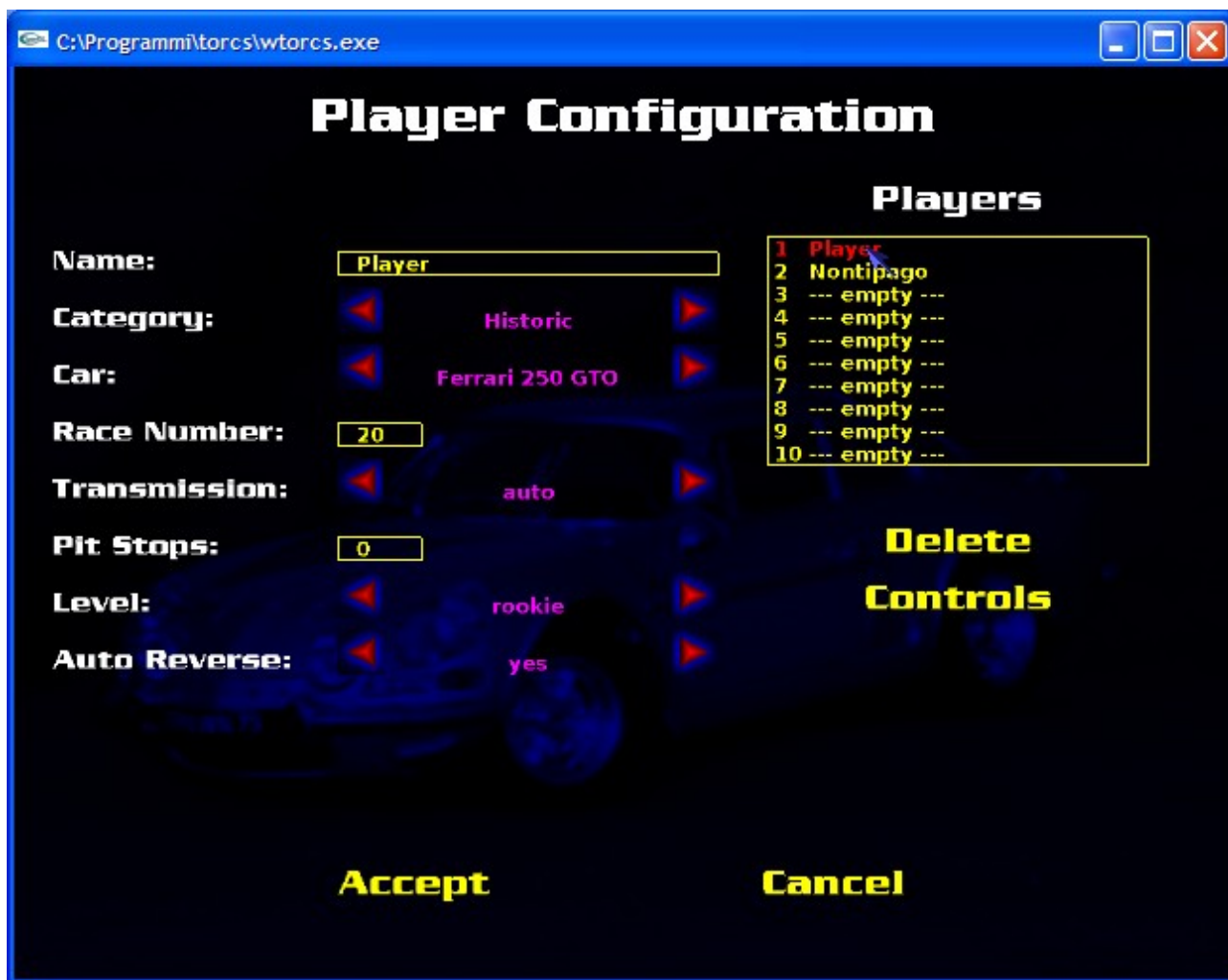
Per Windows in genere l'esecuzione del wizard di installazione è sufficiente, per Linux preferite, se possibile, l'installazione mediante il gestore di pacchetti Synaptic (Ubuntu) o equivalenti in modo da installare automaticamente tutte le dipendenze e le librerie richieste.

La configurazione dei giocatori

Come sempre iniziare un gioco rappresenta la parte più difficile perché siamo curiosi e impazienti di cominciare ed evitiamo accuratamente di documentarci in modo sufficiente: questo paragrafo eviterà errori grossolani e vi farà entrare velocemente nello spirito del gioco.

Dopo aver avviato il gioco un semplice click nella finestra di benvenuto permette di accedere al menù principale nel quale le prime cose da fare sono configurare il profilo del o dei giocatori (Configure Players) e le opzioni (Option) selezionando le opzioni con il mouse.





Questo è un esempio di configurazione del giocatore. In questa sezione si possono preconfigurare sino a dieci giocatori, ognuno dotato di caratteristiche proprie in particolare dopo la scelta del nome che deve essere univoco si può scegliere la categoria di appartenenza:

Track-RWD-GrA: Auto a trazione posteriore con più di 500 hp di potenza

Track-RWD-GrB: Auto a trazione posteriore fino a 500 hp di potenza

Track-4WD-GrB: Auto a trazione integrale fino a 500 hp di potenza

Track-FWD-GrB: Auto a trazione anteriore fino a 500 hp di potenza

Historic: Auto leggendarie degli anni 60' e 70'

Nascar: Auto omologate per i circuiti NASCAR la formula GT degli Stati Uniti

Offroad-4WD-GrA: Auto a trazione integrale da fuoristrada (auto da rally)

Offroad-RWD-GrA: Auto da fuoristrada a trazione posteriore

In funzione della categoria scelta potete selezionare la macchina che desiderate dall'elenco nel quale vengono ovviamente rese disponibili solo i modelli che soddisfano i requisiti di omologazione della categoria selezionata. Quindi le auto per ciascuna categoria sono diverse e diversa e anche il numero delle possibili scelte.

Altre opzioni riguardano il tipo di trasmissione che può essere automatica (da preferire in particolare all'inizio in modo da mantenere la concentrazione sulla guida) e manuale, e il numero di fermate ai box (pit stop) che desiderate fare durante la gara: questo garantisce una vettura più leggera perché la quantità di carburante viene dosata in funzione delle soste ai box automaticamente.

Per quanto riguarda il livello di gioco esistono quattro possibili scelte in ordine di difficoltà crescente:

Rookie: l'auto non si danneggia (0%) e possiede alta aderenza (+40%)

Amateur: possibilità di danni limitati (50%) e aderenza elevata (+35%)

Semi-Pro: possibilità di danni elevata (80%) e aderenza media (+30%)

Pro: possibilità di danni totali (100%) e aderenza normale (+0%)

I danni influenzano progressivamente il comportamento dell'auto e la sua guidabilità e l'aderenza influisce sia nel comportamento in curva che in accelerazione e frenata: entrambi sono parametri importanti e al crescere della difficoltà è necessario prestare sempre maggiore attenzione ad una guida pulita evitando di danneggiare la macchina con uscite di pista e collisioni contro ostacoli e avversari.

La retromarcia (reverse) se abilitata è utile per rientrare più rapidamente in pista dopo un testa coda e si innesta automaticamente quando a macchina ferma viene premuto il freno.

L'ultima opzione riguarda la configurazione dei controlli i tasti da associare alle varie funzioni del programma possono essere infatti variati.

La configurazione dei tasti potete ottenerla in ogni momento con il tasto F1 ma è sensibile al contesto quindi le informazioni variano in funzione di dove vi trovate.

Quelli principali sono:

- **freccia su** accelera
- **freccia giù** frena e se siete fermi innesta la retromarcia
- **freccia destra e sinistra** sterzo
- **F2 F11** cambiano il tipo di visuale
- **1 5** visualizzano o nascondono gli strumenti e le indicazioni dei box
- **0** seleziona la modalità di visualizzazione arcade (utile nelle fasi iniziali)
- **spazio** per saltare il precorsa
- **ESC** per fermare la corsa e aprire il menù del gioco
- **p** per mettere in pausa congelando lo schermo

Il programma prevede anche la possibilità di utilizzare il mouse per la corsa, oltre che per le scelte dei vari menù, ed è anche possibile utilizzare un joystick: in questo caso se il vostro joystick è configurato sarà sufficiente assegnargli le funzioni desiderate selezionandole e muovendo il joystick o premendone i bottoni subito dopo.

Se utilizzate Linux come sistema operativo sarà bene fare riferimento a siti quali [the Vojtech Pavlik's joystick driver site](#) e [Linux Gamers FAQ](#) per ulteriori informazioni sull'utilizzo e la calibrazione dei joystick.

Allo stesso modo è possibile anche utilizzare volante e pedaliera collegati al PC.



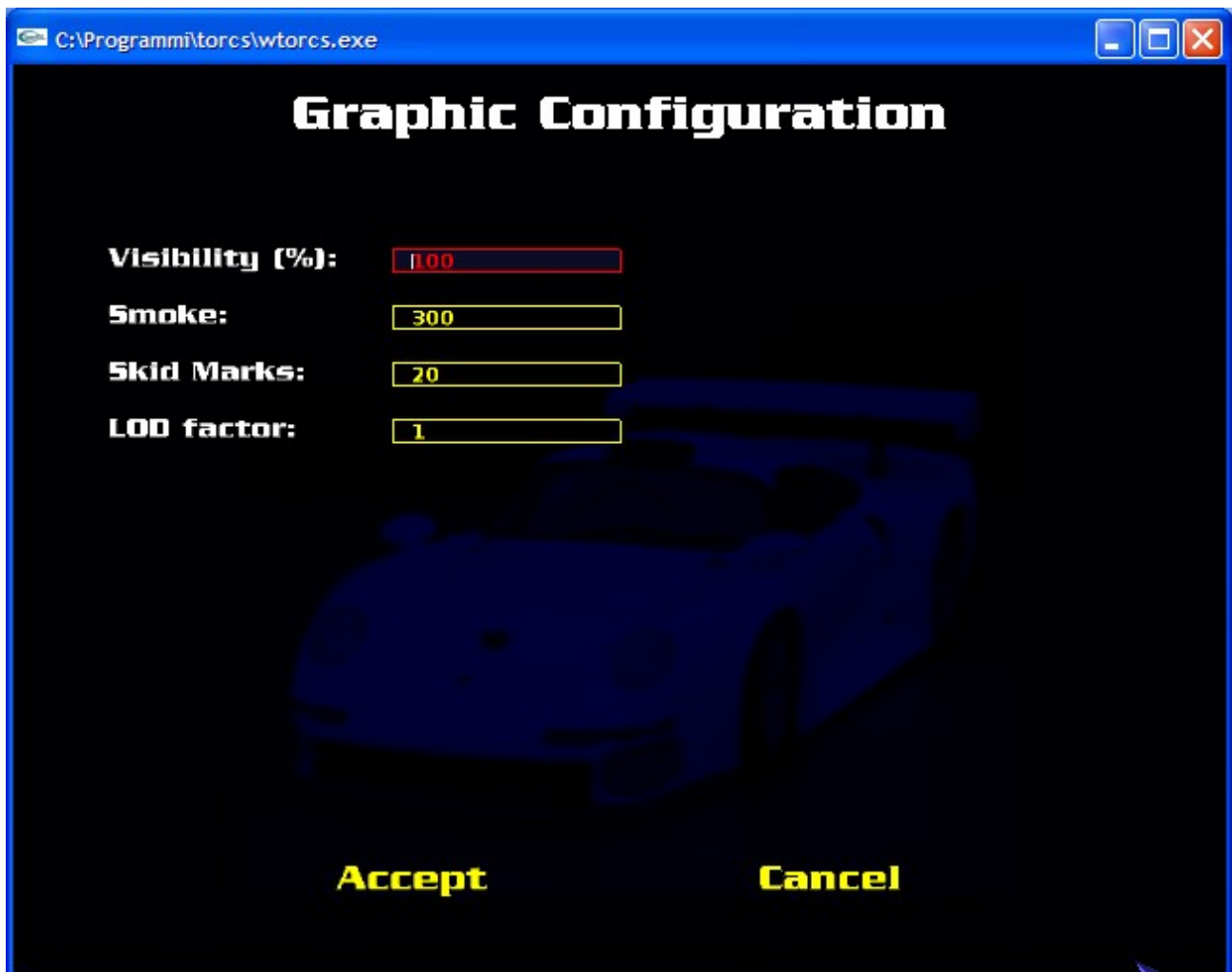
Questa schermata propone la configurazione dei tasti di default.

Questa configurazione è ben strutturata e se non avete particolari esigenze o comunque all'inizio vi suggeriamo di lasciarla tale e quale.

Se intendete giocare in modalità multiplayer dovreste invece configurare tasti diversi per ciascun giocatore oppure selezionare per uno la tastiera lasciando la possibilità agli altri di utilizzare il mouse o il joystick almeno per le funzioni principali.

La configurazione grafica

Dal menù principale scegliendo Option si accede a quattro menù di configurazione che provvedono ad adattare il gioco ai nostri gusti e alle possibilità del nostro sistema.



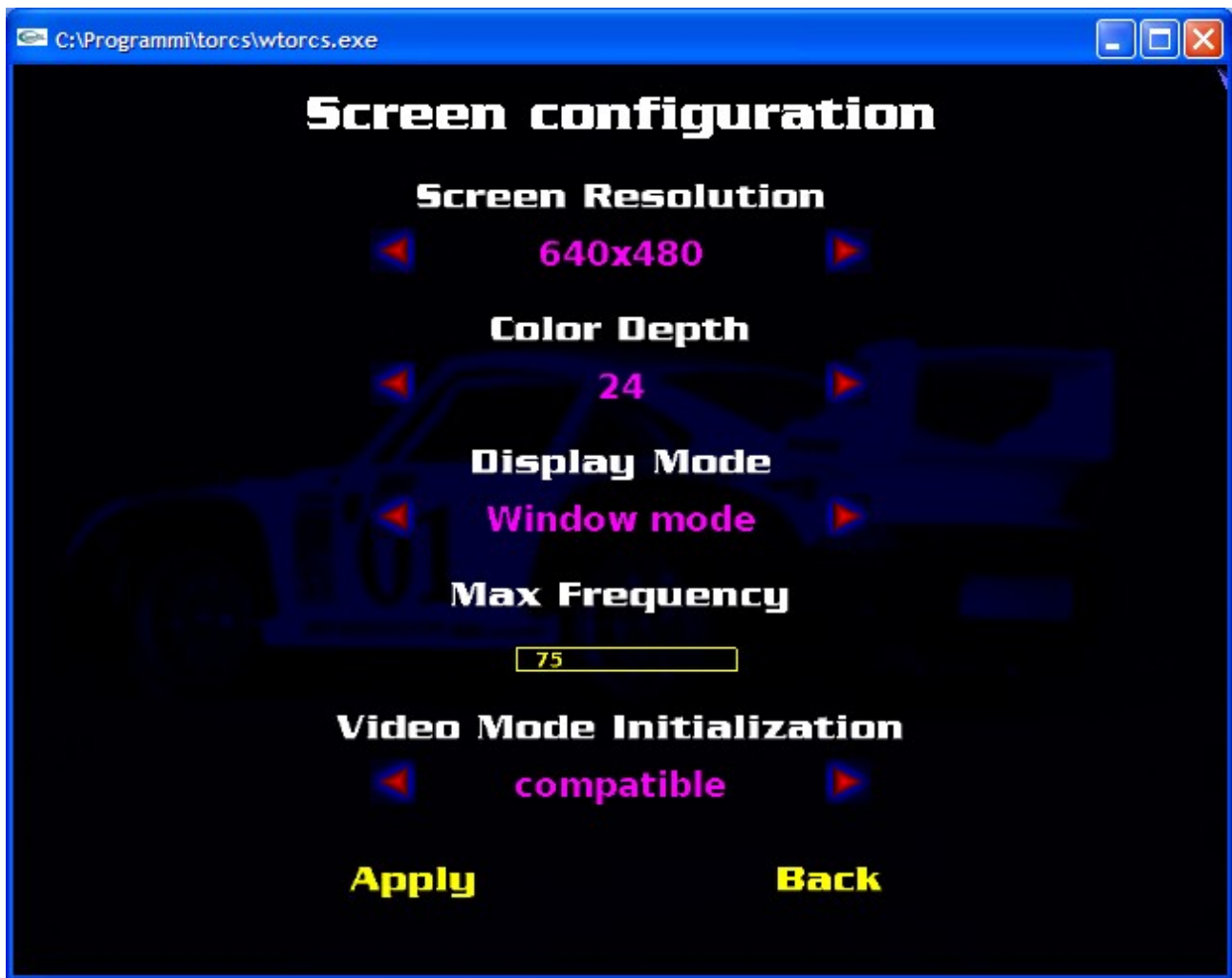
Nel menù della grafica troviamo:

Visibility: la profondità della visione durante la gara, che deve essere sempre un valore positivo e non superiore al 100%, può essere diminuita o aumentata in funzione delle capacità grafiche del computer e della velocità del processore: una visibilità con molta profondità facilita la guida ma può divenire pesante da gestire per il PC quindi è bene eseguire qualche prova e diminuirla se il carico grafico diviene troppo elevato.

Smoke: per il fumo valgono esattamente le stesse considerazioni già fatte per la visibilità.

Skid Marks: come sopra da tarare in funzione della velocità del sistema

LOD Factor: LOD=livello di dettaglio delle auto come per tutti gli altri non può essere un numero inferiore a zero e va dosato in funzione delle prestazioni hardware.



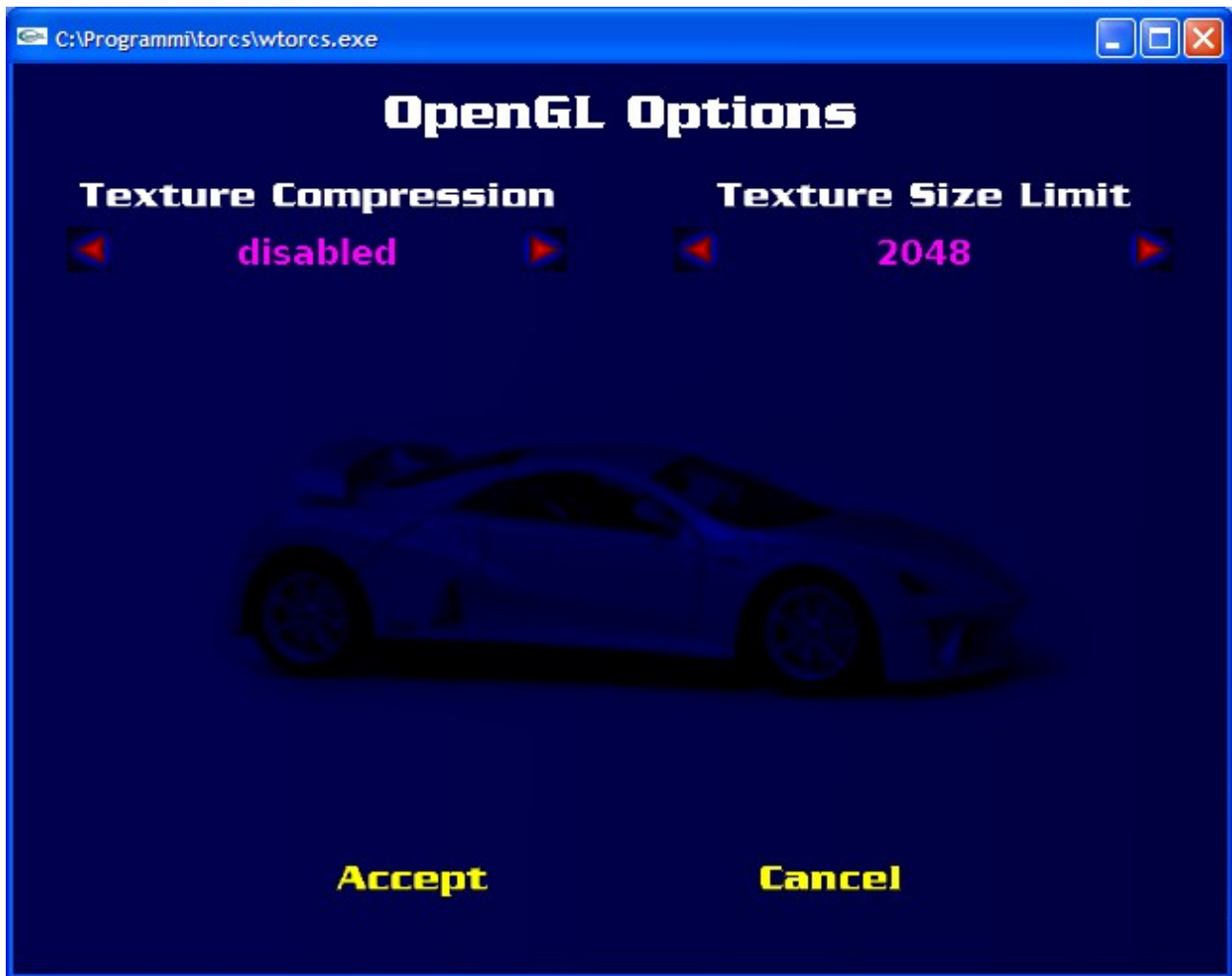
In questa sezione è possibile selezionare la risoluzione dello schermo e la profondità di colore: i valori maggiori rendono il gioco più lento in funzione delle caratteristiche del PC.

E' possibile far girare il gioco in una finestra o in modalità schermo intero: quest'ultima meno pesante per l'hardware.

E' anche possibile scegliere la frequenza di refresh del vostro monitor (che ovviamente dovete conoscere o tramite le informazioni sul sistema o facendo ricorso al menù on screen del monitor oppure per tentativi).

La scelta del modo video è infine limitata a due scelte: compatibile e best.

Nel caso optaste per scelte incompatibili con il vostro computer e il programma si rifiutasse di avviarsi sarà sufficiente modificarle editando con un editor di testo il file "config/screen.xml" e riportandolo ad una configurazione più adeguata. Se desiderate essere sicuri di non avere problemi fate una copia di questo file prima di eseguire tentativi con la configurazione di default o con una sicuramente funzionante e in caso di problemi ripristinate il file con la vostra copia di sicurezza.

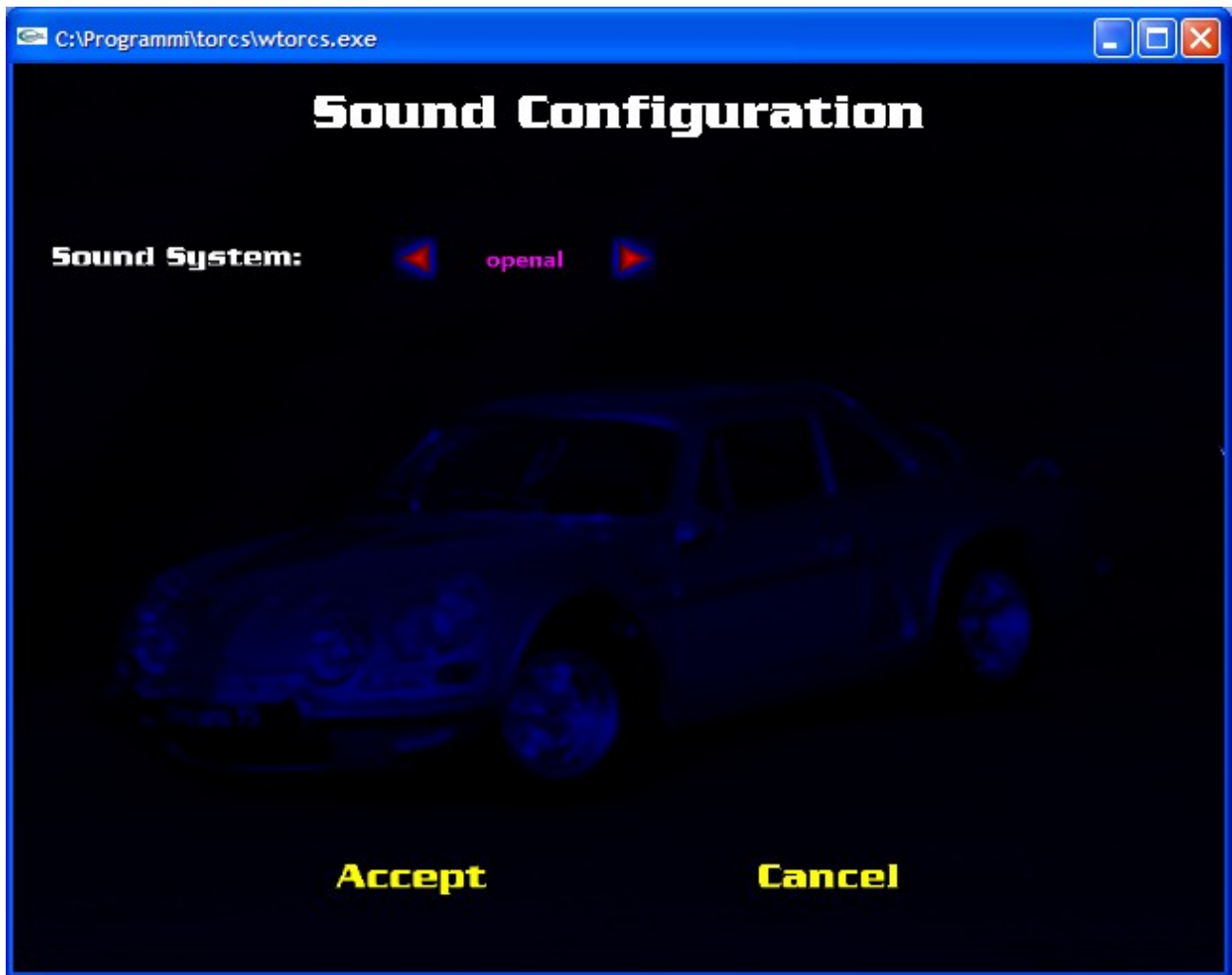


In questa ultima sezione invece si possono definire le opzioni delle librerie grafiche.

Il programma utilizza le OpenGL cioè le librerie libere da copyright ed è possibile variarne alcuni parametri che riguardano le texture cioè i “rivestimenti” dei poligoni grafici che ricoprono gli elementi tridimensionali.

E' possibile attivare o disattivare la compressione delle texture e porre un limite definito alla loro dimensione: tutto ciò al solito influisce sulla resa estetica ma purtroppo anche sulla quantità di dati da gestire e quindi sulla giocabilità quindi è opportuno calibrare sapientemente questi parametri in funzione del proprio hardware come del resto per tutti gli elementi grafici.

La configurazione audio



Per la sezione dedicata agli effetti sonori, ancora non sviluppati come il resto del gioco la scelta si limita alla attivazione e disattivazione degli stessi.

I circuiti

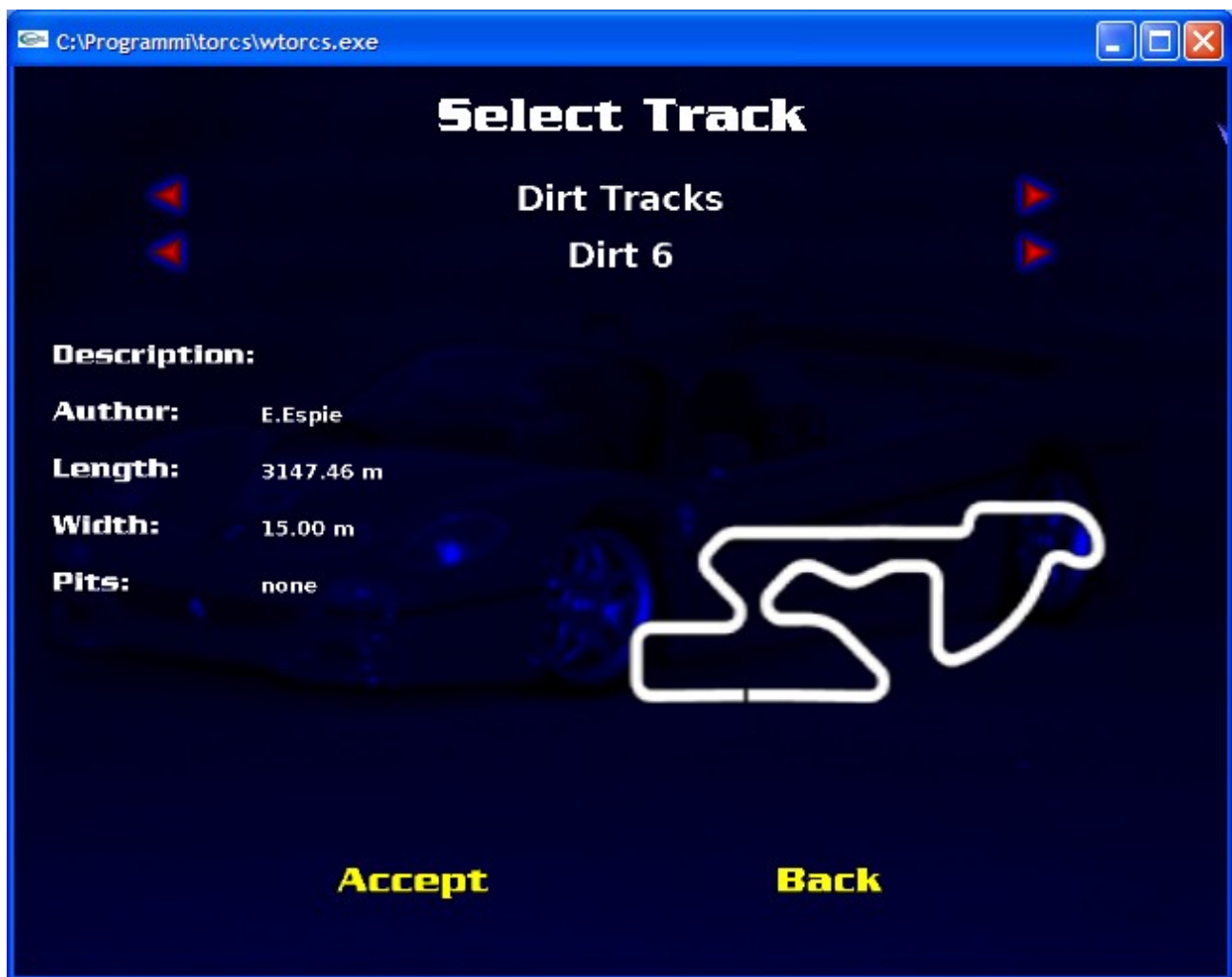
Dal menù race si accede direttamente alle competizioni:



Dalla Quick race la corsa immediatamente fruibile quando siamo impazienti di cominciare alle singole corse di vario genere (comprese quelle per fare pratica) ai campionati che hanno la possibilità di effettuare salvataggi e successivi caricamenti per essere portati a termine in tempi diversi.

Una volta scelto il tipo di gara si ha la possibilità di iniziare subito o di configurare alcune opzioni concernenti la competizione stessa.

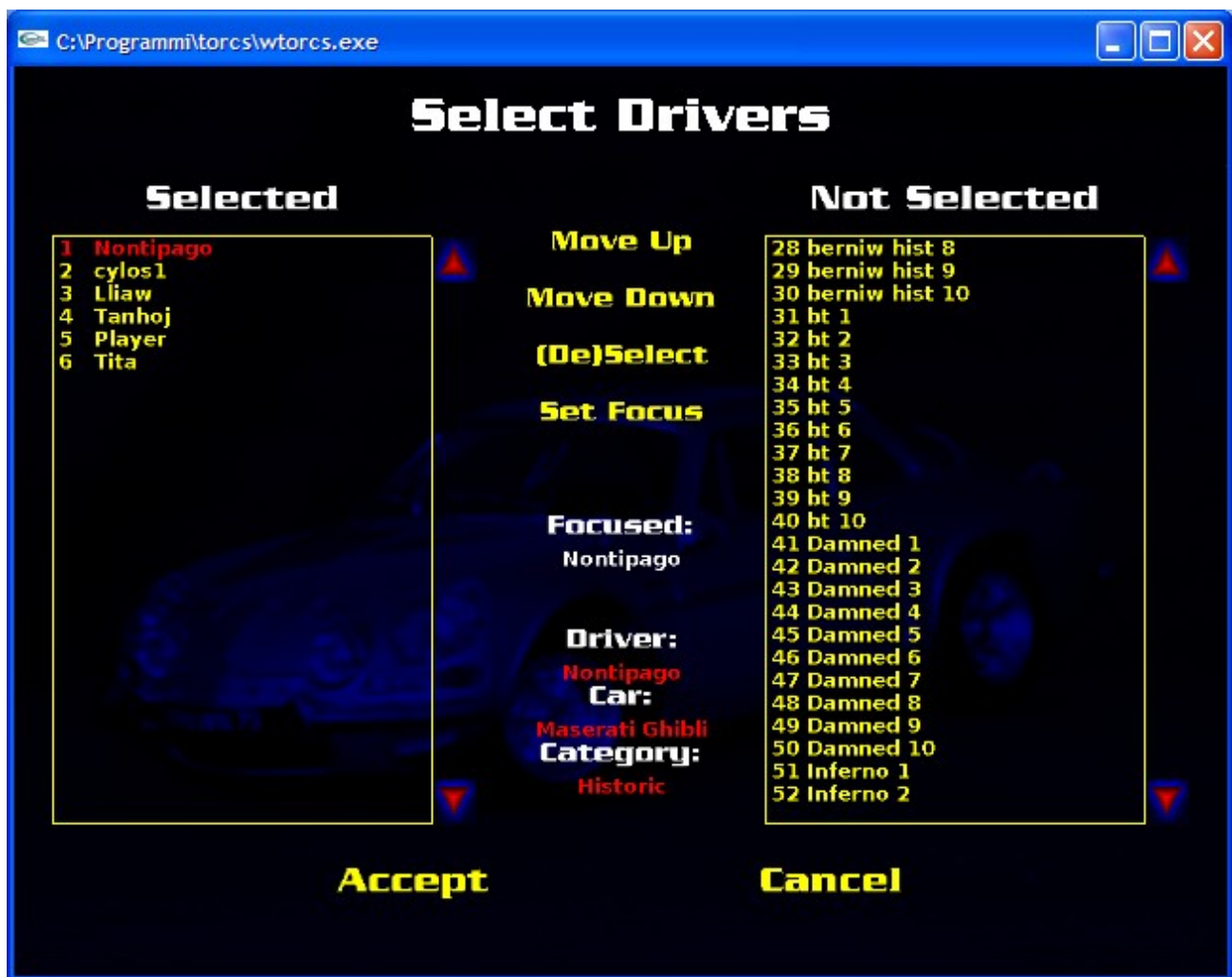
In funzione della categoria scelta possiamo selezionare un sottoinsieme di circuiti (selezione superiore) e fra questi uno in particolare (seconda selezione).



Di ciascun circuito vien presentata una breve descrizione o solamente il nome, la lunghezza del giro e la larghezza della pista (all'inizio sceglietene una larga in modo da facilitarvi le cose).

Viene anche indicato il numero di pitstop, ma soprattutto è utilissima la planimetria del circuito che ci mostra il numero di curve e la difficoltà della pista.

Dopo aver selezionato il circuito di nostro gradimento con un click su Accept si apre un secondo menù



Esistono oltre 70 avversari già prefigurati oltre ai dieci che potete configurare voi nel menù giocatori (configure player) e in questa schermata è possibile scegliere i propri avversari in pista.

Il programma di default ve ne seleziona alcuni che vengono presentati nella finestra di sinistra che però potete deselezionare e quindi scartare sostituendoli con quelli prelevati dalla finestra a destra scegliendoli con il solito click del mouse e selezionandoli con la voce del menù al centro che permette sia di togliere che di aggiungere concorrenti in funzione dell'elemento selezionato.

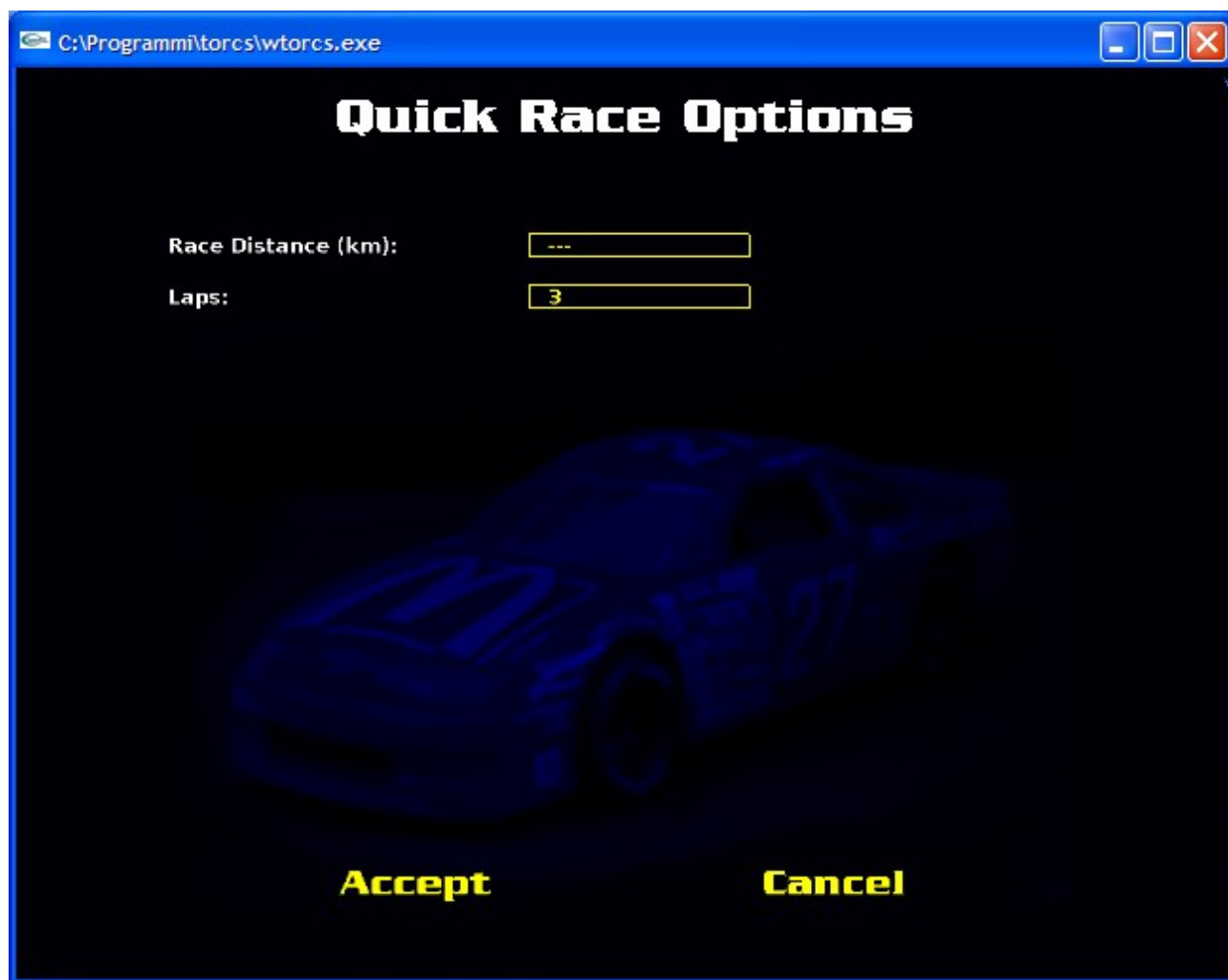
In pratica se è selezionato un pilota presente nella finestra a sinistra lo toglie dalla corsa se invece ad essere selezionato è un pilota nella finestra a destra lo aggiunge.

Potete inserire quanti piloti desiderate gestiti dal PC e anche più piloti "umani" in questo caso lo schermo si dividerà per mostrare le singole macchine ai piloti "umani".

I piloti reali devono possedere una configurazione già stabilita nel menù apposito (configure player) ciascuno dei quali dovrà possedere controlli separati.

In pratica un giocatore potrà utilizzare le frecce, un altro i tasti W,A,S e Z oppure il mouse oppure il joystick e ognuno possiederà una zona dello schermo che gli mostrerà la sua visuale.

Dopo aver esaurito la configurazione della lista dei partecipanti verrà presentato, se necessario, un ultimo menù opzioni....



nel quale si potrà scegliere il numero dei giri di pista della gara o, se preferite, la lunghezza in Km della competizione.

Suggerimenti

Provate subito il gioco lasciando le impostazioni di default in modalità facile (rookie) e impostando la visualizzazione arcade tasto 0.

Prendete confidenza con TORCS, abituandovi alla risposta dei comandi, poi configurate i tasti principali secondo i vostri gusti: le quattro frecce si utilizzano con una sola mano ma potete ad esempio dividere i comandi del gas da quelli dello sterzo per utilizzare entrambe le mani se vi trovate meglio.

Selezionate provandole le varie visuali disponibili per comprendere quale sia più comoda e adatta al vostro stile di guida: questa scelta è anche se non sembra molto importante e può farvi guadagnare secondi preziosi.

Provate i vari tipi di corse, i circuiti e i modelli di auto ma poi scegliete un'auto e fate pratica con quella per un pò per entrare nello spirito del gioco e comprendere come rispondono i comandi. Poi potete dedicarvi a testare le auto sui circuiti ad anello con curve sopraelevate tipo Indianapolis: sono semplici e vi permettono di comparare le caratteristiche delle auto.

In gara cercate di utilizzare una guida "pulita" evitando cordoli e fuori pista che vi rallentano moltissimo e traiettorie fluide e scorrevoli che aumentano la velocità. A destra nella visualizzazione standard trovate una crocetta che indica le componenti della velocità e dell'accelerazione istantanea, con il tempo può diventare una indicazione particolarmente utile per scoprire le traiettorie più efficaci in curva.

Provate ad utilizzare l'Alfa 155 una buona macchina facile da controllare soprattutto all'inizio ma dotata di ottime caratteristiche.

Domande frequenti

0. Su questo Documento

0.1 Licenza e copyright.

Copyright (c) 2004 [Bernhard Wymann](#).

Sotto i termini della licenza GNU Free Documentation License è permesso copiare, distribuire e /o modificare questo documento, versione 1.2 o ogni versione successiva pubblicata da Free Software Foundation; con no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. Una copia della licenza è disponibile [qui](#).

0.2 versione del documento.

2006-10-06.

1. Circa TORCS in generale.

1.1 Che cosa è TORCS?

[TORCS](#), **The Open Racing Car Simulator**, è un simulatore di corse automobilistiche, che permette di gareggiare contro avversari simulati dal computer. Permette anche di sviluppare un vostro personale pilota controllato dal computer (anche chiamato robot) programmato in C o in C++. TORCS è "Open Source" (GNU General Public License Versione 2 o superiori).

1.2 Chi sviluppa TORCS?

Il progetto TORCS è stato creato da Eric Espié e Christophe Guionneau. Currently Bernhard Wymann, Christos Dimitrakakis ed altri proseguono nella loro opera di sviluppo del progetto. Nel sito ufficiale del progetto potete trovare ulteriori informazioni [TORCS site](#) .

1.3 E' disponibile documentazione?

Si vi è molta documentazione disponibile sia nel sito ufficiale del progetto [TORCS site](#) ache nella [Bernhard's homepage](#). Potete reperire istruzioni per l'installazione, il tutorial per i robot, istruzioni sull'utilizzo di TORCS e molto altro. Per quanto concerne i circuiti potete reperire documentazione [qui](#), è anche disponibile un editor per piste [track editor](#). Ciò che invece manca al momento è un tutorial per la realizzazione delle auto.

1.4 Le caratteristiche di TORCS?

TORCS dispone di 42 differenti modelli di auto, oltre 30 circuiti e circa 70 piloti controllati dal computer.

Permette di utilizzare joystick o volanti se sono supportati dal sistema operativo e dal computer che utilizzate. E' anche possibile utilizzare come controlli la tastiera o il mouse

ma in questo caso il controllo risulta più difficile.

Gli effetti grafici riguardano le luci, il fumo, gli adesivi, ecc, mentre la simulazione è caratterizzata da un semplice modello di danneggiamento, collisioni, proprietà dei pneumatici e delle masse non sospese (ruote) e dell'aerodinamica compreso effetto suolo, spoilers, ecc.

L'ambiente di gioco spazia da gare semplici ad interi campionati ed è possibile gareggiare contro avversari umani nella modalità a schermo diviso sino a 4 giocatori umani.

1.5 Progetti futuri?

In primo luogo non considerate questa risposta alla stregua di una promessa perché implementare e migliorare ogni caratteristica richiede ormai un tempo decisamente lungo. Contemporaneamente alle operazioni di manutenzione e all'eliminazione dei bug verranno probabilmente implementate: miglioramento degli inviluppi, sonoro con OpenAL, miglioramento delle visuali, replay e gioco in rete.

1.6 Perché non fate...?

Non bisogna dimenticare che noi stiamo sviluppando TORCS nei nostri momenti liberi e gratuitamente. I vostri desideri, le vostre idee e richieste sono benvenute nella [TORCS-users](#) mailing list. SE nessuno realizza la vostra idea siete liberi di contattare il project leader e realizzare voi stessi la vostra intuizione. Ma per favore evitate di inviare messaggi del tipo: "I am sure there are numerous technical reasons why not to use GLUT for games." This kind of argumentation might work for a junior president of a well known country to convince the citizens of the existence of weapons of mass destruction in Iraq, of course "we are sure there are weapons of mass destruction in Iraq"... But here you better put up some arguments;-)

1.7 Cosa si intende con robot?

In TORCS un "robot" è una porzione di codice di programma che pilota un'auto e tutti gli avversari che scegliete sono in effetti robot.

1.8 Perché non sostituire GLUT con SDL?

Faremo presto questo cambiamento ma for fun I leave the former answer.

Ah, simple question, simple answer. Like you can easily see G-L-U-T has four letters and S-D-L has just three letters, that should be enough proof of superiority, is it? Ok, then look at the version numbers, GLUT has a major 3.7, SDL has a childish 1.2.7, so GLUT is $3.7/1.27 \Rightarrow 2.913385826$ times better at least. By the way, I am sure there are numerous technical reasons why not to use SDL for games, got it, bro?

Ok, now back to serious mode. First let us have a closer look at the question, to understand it you need to have an idea what [SDL](#), [GLUT](#) and [plib](#) provide. In a nutshell, SDL provides framebuffer access, OpenGL context, a thread API, support for input devices and audio output. There are loads of additional services available through additional libraries. GLUT provides an OpenGL context, keyboard and mouse input and simple window management functionality. Now you can see that SDL provides much more functionality than GLUT, but where does e. g. the sound in TORCS come from? Yes, it comes from plib. Plib provides sound, a scene graph, a geometry library, joystick input and much more.

Now you can see what we are talking about. We need pLib because of its scene graph, so the only thing we need and SDL could provide is the OpenGL context, input and simple window management. Hmm, sounds familiar, does it? Yes, GLUT fits perfectly, no SDL functionality is needed at the moment. That raises the question why should we replace working code with new code if there is no advantage? Yes, it is simply a crazy idea. Keep in mind, all the libraries are maintained and have an appropriate license. Both FreeGLUT and SDL are not standard, so there is also no advantage.

Conclusione: SDL è una grande e valida libreria della quale non abbiamo per ora necessità. E' anche possibile che ciò si modifichi in futuro, chi può saperlo?

1.9 Esistono campionati ufficiali?

Sì, è possibile trovare campionati offline per robot [qui](#).

1.10 Devo iniziare un mio personale racing sim project?

Assolutamente no! Perché? Con una ricerca su sf.net e Google, potrà trovare dai 15 ai 40 open source car/driving/racing sim projects, e molti di loro, pur promettenti, risultano abbandonati. Invece di iniziare e abbandonare un nuovo progetto è decisamente molto più promettente contribuire ad uno già iniziato e magari abbandonato.

Molte persone sottostimano il lavoro e l'impegno necessari a portare avanti un progetto quindi è decisamente più conveniente collaborare all'ultimazione di un progetto già intrapreso piuttosto che iniziarne uno nuovo con il rischio di abbandonarlo in seguito.

2. Piattaforme e requisiti.

2.1 Su quali sistemi operativi funziona TORCS?

Linux (x86, AMD64 e ppc), FreeBSD, Windows 98/2000/XP.

2.2 Quali sono i requisiti hardware?

2.2.1 Sviluppo dei robot.

Minimi: 400MHz CPU, 128MB RAM, OpenGL 1.2 scheda grafica compatibile con 16 MB RAM.

Suggeriti: 600MHz CPU, 256MB RAM, OpenGL 1.3 scheda grafica compatibile con 64 MB RAM.

2.2.2 Gioco.

Minimi: 550MHz CPU, 128MB RAM, OpenGL 1.3 scheda grafica compatibile con 32 MB RAM.

Suggeriti: 800MHz CPU, 256MB RAM, OpenGL 1.3 scheda grafica compatibile con 64 MB RAM.

2.3 Quali sono i requisiti software?

Avete bisogno di OpenGL/DRI drivers recenti, funzionanti e correttamente configurati. Avete inoltre bisogno di altre librerie. Per evitare problemi preferite [FreeGLUT](#) a GLUT. Accertatevi di utilizzare pLib version 1.8.3. Per AMD64 necessitate il settaggio "-fPIC" del

compilatore durante la compilazione di plib (export CFLAGS="-fPIC", export CPPFLAGS="-fPIC", export CXXFLAGS="-fPIC"). Per le librerie OpenAL e la migliore qualità del suono prelevate la versione dal sito ufficiale TORCS.

3. Installazione.

3.1 Come installare TORCS?

Potete reperire le istruzioni di installazione su [TORCS site](#) e su [Bernhard's homepage](#).

4. Problemi e soluzioni.

4.1 Problemi di ordine generale e approccio risolutivo.

Potete incontrare problemi di questo tipo: configurazione di sistema, di realizzazione e di runtime. Se incontrate problemi seguite questa procedura:

- Cercate nelle domande frequenti perché le vostre difficoltà possono essere già note.
- Ripulite la vostra vecchia installazione di TORCS compresi i settaggi personalizzati.
- Leggete con attenzione le istruzioni di installazione controllando ogni passo due volte.
- Accertatevi di non aver installato versioni multiple delle librerie GLU.
- Provate a lanciare TORCS con l'opzione "-s" che disabilita le texture multiple.
- Accertatevi che le versioni dei vostri driver, librerie e utility ALSA corrispondano.
- Provate la configurazione delle OpenGL installate con tuxracer e bzflag.
- Cercate nella mailing list, potreste trovare la soluzione.
- Se non siete riusciti a risolvere [report a problem](#) e contattate la mailing list.

4.2 Build Problems.

4.2.1 ./configure reports "configure: error: Can't find GL/glut.h".

Download and install [FreeGLUT](#) or install GLUT which is probably included in your distribution (you need also the header files, which may be in a separate package like glut-devel). I suggest using FreeGLUT.

4.2.2 ./configure reports "configure: error: Can't find plib/ssg.h" or "plib/ul.h".

TORCS requires [plib](#) 1.8.3 and its header files. Download and install plib.

4.2.3 TORCS does not compile on Windows.

To compile TORCS in Windows you need Visual C++ 6.0. Follow exactly the instructions in READMEWIN32.txt to build a release. Do not try to build a debug version. If you see this error message you did probably not follow exactly the instructions.

Linking...

Creating library Release/txml.lib and object Release/txml.exp

The syntax of the command is incorrect.

The syntax of the command is incorrect.
Error executing c:Windowssystem32cmd.exe.

TORCS - 1 error(s), 0 warning(s)

4.2.4 FreeGLUT compilation fails with NVidia OpenGL driver.

If you have an NVIDIA driver and FreeGLUT breaks the compilation with complaints about libGL.la, then update your drivers to 5336 (previous libGL.la was broken).

4.2.5 TORCS does not compile with make: 'Make-config' is up to date.

You have some environment variables set like \$TORCS_BASE and \$MAKE_DEFAULT (to follow the robot tutorial), they are pointing now to the wrong directory. Unset or update the variables suitable for the new directories.

4.2.6 Compiling stops with "error: `GL_TEXTURE0_ARB' undeclared".

If the compiler stops with the above message you have probably OpenGL header files installed which do not define the GL_TEXTURE0_ARB and other symbols. If you have updated your NVidia drivers to 6xxx you will experience this error (NVidia removed those definitions...). To fix it for TORCS 1.2.4 you need to install a gl.h which defines the missing symbols. You can get it from your old (5xxx) NVidia drivers or from Mesa (the problem is not the driver itself).

4.2.7 Compiling fails with "The syntax of the command is incorrect. Error executing c:winntsystem32cmd.exe".

You have probably installed your sources in a path which contains blanks or other special characters, e.g. on the desktop. Move your TORCS-1.2.4 folder to a location with a "clean" path, a drive is the best bet (e.g. D: orcs-1.2.4).

4.2.8 ./configure fails in "checking for sin in -lm" on Gentoo.

This is a Gentoo problem, have a look at [this report](#).

4.3 Startup and Runtime Problems.

In several situations TORCS will crash at startup. The cause are often not bugs of TORCS, but a broken installation or buggy OpenGL/DRI drivers.

4.3.1 Broken OpenGL/DRI drivers or '~' file.

```
/usr/local/bin/TORCS: line 54: 31895 Segmentation fault
$LIBDIR/TORCS-bin -I $LOCAL_CONF -L $LIBDIR -D $DATADIR $*
```

There are two known possibilities which cause such a crash:

- You have buggy OpenGL/DRI driver, you should update to a recent version. Start TORCS with the "-s" option to disable multitexturing.
- You have edited a TORCS XML file (e. g. endrace.xml) and your editor left a copy with an appended "~" (e. g. endrace.xml~). Remove those "~" files.

4.3.2 Multiple GLU libraries.

```
linuxModLoad: ... modules/graphic/ssggraph.so:  
undefined symbol: gluBuild2DMipmaps  
/usr/local/bin/TORCS: line 50: 21215 Segmentation fault  
./TORCS -I $LOCAL_CONF $*
```

You have probably two GLU libraries installed on your system and TORCS has been linked to the wrong one, which does not export the "gluBuild2DMipmaps" symbol. First find all your GLU libraries on the system (cd /usr; find . -name "*GLU*"). Then check which one exports the "gluBuild2DMipmaps" symbol (e. g. nm ./lib/libGLU.so | grep gluBuild2DMipmaps). Finally move or remove the GLU libraries which do not define the symbol, such that the linker can't see it. Now recompile and reinstall TORCS.

4.3.3 Not removed old TORCS version.

```
/usr/lib/TORCS/TORCS-bin: relocation  
error: /usr/lib/TORCS/lib/libtgfclient.so: undefined symbol:  
_Z15GfParmWriteFilePKcPvPc
```

If you see such a message you have probably not cleaned up your old TORCS installation. Make sure that you have removed all old TORCS files, then recompile and reinstall TORCS.

4.3.4 TORCS reports missing files and crashes.

```
data/fonts/b5.glf: No such file or directory  
data/fonts/b5.glf: No such file or directory  
data/fonts/b5.glf: No such file or directory  
data/fonts/b5.glf: No such file or directory  
data/fonts/b7.glf: No such file or directory  
data/fonts/b7.glf: No such file or directory  
data/fonts/b7.glf: No such file or directory  
data/fonts/b7.glf: No such file or directory  
data/fonts/digital.glf: No such file or directory  
Can't open file data/img/splash-main.png  
Can't open file data/img/splash-single-player.png  
Can't open file data/img/splash-qrdv.png  
/usr/local/bin/TORCS: line 54:6839 Floating point  
exception$LIBDIR/TORCS-bin -I $LOCAL_CONF -L $LIBDIR -D $DATADIR $*
```

If you see this you have not installed the mandatory data packages or installed it in the wrong location. Please read the installation instructions carefully and reinstall TORCS.

4.3.5 My screen resolution is not supported.

You have a screen resolution which is not available in the selection menu of TORCS (e. g. 1152x864). Exit TORCS and edit the file .TORCS/config/screen.xml, change the x, y, window width and height to your values.

4.3.6 I have a problem with the fullscreen mode.

If you switched TORCS to fullscreen you get a scrolling screen or the screen size is not fully utilized. If you use the original GLUT library you should choose the resolution of your screen in the display settings. To fix the problem you can also download and install FreeGLUT, then recompile and reinstall plib and TORCS.

4.3.7 TORCS graphics is slow and it crashes on "READY!"

Select in the TORCS display menu a color depth of 24 or 32 bpp, even if your display runs at 16 bpp.

4.3.8 TORCS on Windows hangs after "READY!"

Shutdown TORCS and restart it, go to the sound menu and disable the sound or try the other backend. The problem is likely caused by some library we use, but it never happened to us, so we could not fix it.

4.3.9 TORCS does not start and reports sIDSP:.

Make sure that the sound device is available for TORCS and you use the latest drivers. Perhaps you have a sound daemon running in the background and you need to start TORCS with a wrapper (e. g. artsdsp of KDE) or to disable the daemon.

4.3.10 The cars are flickering.

Probably your depth buffer has just 16 bit, you should have seen a hint at TORCS startup. Try to run your X-server at 24 or 32 bpp, or update/replace your OpenGL/DRI drivers. In Windows set the quality in the OpenGL properties tab to maximum, such that TORCS has a chance to get a 24 bit depth buffer.

4.3.11 I use ALSA and TORCS does not start up.

You get an error message similar like this:

sIDSP: getBufferInfo: Broken pipe.

Although the sound works with the other applications you have messed up your ALSA installation. Make sure that the versions of your ALSA driver, libraries, tools and utilities are matching. This can especially happen if you run a selfcompiled kernel.

4.3.12 TORCS does not start up and complains about display requirements.

Visual Properties Report

The minimum display requirements are not fulfilled.
We need a double buffered RGBA visual with a 16 bit depth buffer at least.
GLUT: Fatal Error in TORCS-bin: visual with necessary capabilities not found.

If you use the open Mesa/DRI drivers, update your OpenGL/DRI drivers and try FreeGLUT. It seems to be a bug in the visual matching code of GLUT or OpenGL/DRI. The output of 'glxinfo | grep "OpenGL version"' should report at least "1.2 Mesa 5.0.1".

4.3.13 The racetrack mini map is just a blue square.

You have probably more than one GLU library installed on your system and TORCS is linked with a version that does not match the version defined in the header files. Remove all old GLU libraries, make sure that the header files are matching with the library. Make also sure that you have recent OpenGL headers installed which define the GL_UNSIGNED_INT_8_8_8_8 symbol, then recompile TORCS.

4.3.14 TORCS runs but without textures.

This is probably an OpenGL setup problem, check your installation with bzflag and tuxcart. You have to reinstall the XFree86 libraries, FreeGLUT and your OpenGL drivers.

4.3.15 TORCS on Windows crashes after "new race".

The instruction at 0x0043dda5 referenced memory at 0x00000004. The memory could not be written.

TORCS on Windows crashes with the above message after you have chosen "new race". Probably you have not downloaded all required files or installed them in the wrong place. Please check your TORCS installation carefully.

4.4 Other Problems.

4.4.1 How do I capture a movie?

First you have to configure the video capturing in config/raceengine.xml. The captured frames are stored in the output directory, so you need to have write permission and plenty of space (~ 1GB for one minute, depends of framerate and resolution).

During the simulation you can hit "c" to start and stop the capturing. The frames are stored in the output directory and named this way: TORCS-ssss-ffffff.png, ssss is the sequence number and fffffff the frame number in the sequence.

Now you need a tool to compose the frames to a movie. If you use mencoder (of the mplayer 0.92) you have first to convert the images to jpg's with the png2jpg tool from [src/misc/png2jpg](#). Now encode the video with mencoder -ovc lavc -lavcopts vcodec=mpeg4:vhq:vbitrate=1800 *.jpg -mf on:fps=25.

4.4.2 Can I develop a robot using Windows?

Yes, you can. Here is a rough guide:

- Download the TORCS source and manage to compile and run it.
- Download e. g. the bt robot package, it is a little enhanced version of the robot developed in the tutorial, so there is good documentation available to understand it.
- Copy the sources into a new directory, e. g. when you choose the name "myrobot" create a directory /src/drivers/myrobot.
- Copy all the bt files into the myrobot directory.
- Now rename all "bt*" files into "myrobot*".
- Edit myrobot.def and also change all bt strings into myrobot.
- The same for myrobot.dsp, .xml, perhaps more, you get the idea.
- Change the "bt" module entry point in myrobot.cpp to "myrobot", also make the strings for the robots names and description match with those in myrobot.xml.
- Add the myrobot project to TORCS, open the project with vc++ and try to compile it.
- If you want to deploy the files you have to do that manually, create a directory in "runtime" where the other robots are and copy the required files there (myrobot.dll, .xml, setup subdirectories etc.).
- Now TORCS should pick up your robot.

4.4.3 Is my joystick/steering wheel/gamepad supported?

If your operating system supports your device it will also work within TORCS. To learn how to setup your input device visit the TORCS site and read the "how to drive" section.

4.4.4 How do I set up a multiplayer game?

TORCS supports split screen multiplayer games with up to four players. Setup:

- Set up the players in the "Configure Players" screen.
- If more than one player uses the keyboard, read [this](#).
- Now choose a race, select the players in the configuration screen.
- Start the race, stop it with "p" to pause.
- Now create the split screens with "J", or remove them with "[".
- If you want to change the view of a "mini screen" point with the mouse on it and set it up as usual (F1-F11, ">", "<", ...).
- When you have set up all views press "p" to end the pause and start the race.
- Have fun;-)

4.4.5 I cannot adjust the joystick/wheel centering.

Before you start TORCS make sure that the joystick/wheel is correctly set up/calibrated for the system. You can use the "jscal" and "jstest" utilities for that, you should find them in your Linux distribution. If this works fine center your axis (e.g. if you have noncentering axis like sliders), the start up TORCS and set up your players and controls. It is important to center the axis before you go to the calibration.

5. Problem reporting.

5.1 When do I report a problem?

You should report a problem after you have carefully read the available documentation and did not find a solution for the problem.

5.2 How and where do I report a problem?

First subscribe to the [TORCS-users mailing list](#) (otherwise your messages need to be approved, so it can take long till they become distributed). Send us as much information about the problem as you know. At least provide the following:

- A good description of the problem.
- The TORCS version, did you compile from source?
- The console output of TORCS.
- Your operating system (uname -a).
- The output of "glxinfo -l" (Linux, FreeBSD).
- The /var/log/XFree86.0.log file (Linux, FreeBSD).

Please do not file a bug in the bug reporting section on sourceforge, it is not a support area for users (if you know for sure that you have found a bug in TORCS, then it is ok to report it in the bugtracking system).

6. Links.

6.1 TORCS.

- The main TORCS site www.TORCS.org.

- The [TORCS-users mailing list](#).
- TORCS official championships www.berniw.org/trb.
- TORCS development snapshots TORCS.free.fr.
- Bernhard's site www.berniw.org.
- Christos Dimitrakakis TORCS mods www.idiap.ch/~dimitrak/TORCS.html.
- Charalampos Alexopoulos track editor katergo.rege.org/projects/trackeditor.
- TORCS Drivers Championship <http://speedy.chonchon.free.fr/indextdc.html>.

6.2 Libraries and Drivers.

- FreeGLUT freeglut.sourceforge.net.
- Plib plib.sourceforge.net.
- Mesa www.mesa3d.org.
- DRI Project dri.sourceforge.net.
- XFree86 project www.xfree86.org.
- Precompiled drivers for XFree86 www.xfree86.org/~alanh.
- DRI driver limitations and problems www.xfree86.org/4.3.0/DRI10.html.
- ALSA www.alsa-project.org.
- OpenAL www.openal.org.
- Several updated joystick drivers user.it.uu.se/~johannd/projects/ff/index.shtml.

6.3 Other.

- TORCS in research students.cs.byu.edu/~kroundy/research.
- Linux Journal TORCS review www.linuxjournal.com/article.php?sid=7080.
- TORCS review at linuxgames.com
www.linuxgames.com/?dataloc=/reviews/TORCS/.
- TORCS review of Mousehole www.freegamesweb.co.uk/game.php?ID=75.
- TORCS review at Unixreview
www.unixreview.com/documents/s=9846/ur0508d/ur0508d.html.
- TORCS at the university www1.coe.neu.edu/~mourant/cse270.

Gli autori del progetto

TORCS è stato realizzato e viene sviluppato con il contributo di molti programmatori:

People:

Project Leader	Bernhard Wymann	
Developer	Christos Dimitrakakis	Simulation, camera, sound code, samples, tracks, robot.
Developer	Charalampos Alexopoulos	Trackeditor project, trackgen improvement
Developer	Patrice Espié	Windows port of version 0.0.21, trackeditor components
Artist	Eliam SpeedyChonChon	Open Wheel Car, reworked car, track, TDC (TORCS Drivers Championship)
Artist	Olaf Sassnick	Reworked and new cars
Packager	Thierry Thomas	FreeBSD package, testing and bugfixes
Tester	Eugen Treise	XSLT for result formatting, testing
Artist	Andrew Sumner	New/reworked tracks and cars. MacOSX port
Developer	Brian Gavin	SDL port
Developer	Mart Kelder	Bugfixing
Artist	Christophe Macours	Sound samples
Developer	Rémi Coulom	K1999 robot
Artist	Andrea Alfieri	Alfa Romeo 155 V6 Ti DTM car model
Artist	asdas asda (ViPeR)	Viper, Xj-220, Acura NSX, Lotus GT1, McLaren F1
Artist	Patrick Wisselo	Rally cars
Packager	Bernhard Kaindl	SuSE package (and a lot of help)
Packager	Gernot Galli	Debian package
Developer	Henrik Enqvist	In code profiler
Packager	Per Ovind Karlsen	Mandrake package

Packager	Matthias Saou	RedHat package
Packager	Neil Winton	TORCS 1.2.1 Windows package
Artist	Butch - K/Cendra	60's cars
Artist	Vincent Moyet	Race cars
Developer	Jens Thiele	Billy robot
Developer	Paul Bain	Astigot robot
Artist	Jean-Christophe Durieu	McLaren F1, Acura NSX, Lotus GT1, Buggy, 360 Modena skins... Baja bug model, logos and very nice drawings

TORCS creators:

Eric Espié
Christophe Guionneau

Authors of free code used in TORCS:

plib	Steve Baker
expat	James Clark
SOLID	Gino van den Bergen
GIFont	Brad Fish
easymesh	Bojan NICENO

Applications used for TORCS:

AC3D	<u>3D Modeler for the cars and tracks</u>
Blender	<u>3D Modeler for the cars and tracks</u>
Gimp	<u>Texture creation and manipulation</u>
Stripe	<u>Software tool to convert a polygonal model into triangle strips</u>