

DooM2d

Alex Watson

COLLABORATORS

	<i>TITLE :</i> DooM2d		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Alex Watson	April 14, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	DooM2d	1
1.1	DooM2d - Main page	1
1.2	Introduction	3
1.3	DooM2d History	4
1.4	DooM2d's incredibly high minimum spec...	6
1.5	Playing DooM2d II	7
1.6	DooM2d II Legal disclaimer - or - why not to sue me	8
1.7	DooM2d custom maps	8
1.8	Example .map file	13
1.9	All about me and my favourite things	14
1.10	About the included DooM2d II levels	15

Chapter 1

DooM2d

1.1 DooM2d - Main page

```

+++++
+++++
+++++
#####
#####
#####
##   ###
o##  ###
###  ##  #o+++++
###  ##  #####+++++
##o   ## #####++#####
##   ##  ##  #####+#####
##   ##  ##  +##+#####
#####  ##  +#####++  ##  ##+#####  o###o###  ##
o#####  ##  +#####++  ##  ##+#####  ##  ##  ##
o#####  ##o###+#####  ##  ##+#####o  ##  ##  ##  o##
o#####+#####  ##  o##+#####  ##  #####
#####++#####  ##+#####  ##  #####
+++++#####  ##+#####++  ##  ##  ##
+++++++##  #o+#####  ##  ##  ##
+++++++  ++##+#####  #####o  ##  ##
+++++++  ++++++#####  #####  ##o  o##
+++++++  ++++++++  o#####  #####
+++++++  ++++++++  o##  #####
o+++++++  o+++++++  #####o
+++++
+++++
+++++

/-----\
> Amiga Format Iridium Edition <
\-----/

```

! THE ULTIMATE TEXT BASED BLOODFEST !

DooM2d 2

Welcome to DooM2d's guide! If you haven't played it yet, then you'll have to know ←
that
it is, in fact, a text game. But don't quit yet! It may give you ten- OK, five
minutes of fun and it isn't like those infocom games. It owes more to DOOM
than Monkey Island. Oh yes. Not that you would guess by the name, which of
course is purely by coincidence nearly the same as a similar but totally different ←
game
of a similar name when you remove some letters and then spell it differently.

This special AF edition contains some different things in the program and an ←
exclusive
level! The guide is also updated.

Introduction

What this is, why and stuff.

Playing

How to play and stuff.

Custom Maps

Making your own levels! (EASY!). And stuff.

About levels

The levels and stuff included here

About me!

Who's this then? Not forgetting stuff.

Legal stuff

Legal stuff. And why not to sue me.

Requirements

What it needs (or not)

Sorry if this guide is sightly dodgy in places, I wrote it in a hurry and added ←
bits
later without much thought, and it was written mostly when the game was finished ←
in July
'98, although due to an annoying bug I left the game unreleased until now ;(

PS: If you have only a 68000/'020 Amiga, its a good idea to start the program now,

as it takes ages to load in levels (I didn't notice until recently; It's fine on my
'040. I'm not good at assembly and since most of the waiting is processing strings
in pretty basic ways, it wouldn't be worth it. I will soon make it so it loads them
as you play the levels.

1.2 Introduction

Once upon a time, many moons ago (well, 1996), someone was thinking "Hmmm...what sort of game could take the world by storm once more..." or something along those lines. So DooM2d was born. A few months later, and a genuinely reasonable game was realized!. So, DooM2d, written in dear old AmigaBASIC, struggled through life with the most basic of programming, no flash graphics or sound an an old A500.

Eventually though, after months of dedication and the development of not only Amiga, but PC and BBC(!) versions too, and a multiplayer prototype, interest waned, especially with the advent of Blitz Basic. The creator was lured to the attraction of graphics and sound. But one day in May '98, he remembered the nostalgic old game, and that programming didn't have to involve hair-tearing over forgetting to OR-in the correct bitplane value on a CopList, and started work.

Three weeks later, after abandoning an attempt to replicate the original, a new, completely re-coded version was running. DooM2d was back with a vengeance!

DooM2d is text adventure, of sorts. But you don't wander around in a forest picking up vaguely useful items and using them on other everyday objects (no, I do like them!). Instead, in the first DooM2d, you walked around a strange medievell-type world picking up items such as pistols, rockets, chainsaws and exploding sheep and then using them on people that get in your way. And that's a lot of people. The problem with the original DooM2d though, was that due to constraints of using AmigaBASIC (mainly) was that all levels had to be integrated with the program code, it was slow, and unstable. So DooM2d, now, can load levels from a simple script file, and there is no speed problem (unless, perhaps, you try running it on a PC - nuuurrrrr!). To enjoy DooM2d, you really need to realize that most of it's appeal comes from the fact that it is'nt graphical - leaving you to imagine it. That and being able to put real locations and people (that you don't like!) in it.

For more info on making levels, read
this!

Don't worry if you are no good at programming or whatever,
all you need is a bit of imagination; and I emphasize a bit!

Read the
history

if you want.
As DooM2d is obviously a very high-end game, you'd better read the requirements

1.3 DooM2d History

```
v1.0 > * Sometime in 1996
* One level called "Death on Elm Street" - WHY???
* Not released - never actually finished the map (though you can play a ← ←
  replica in DooM2d II)
* Sorry, I can't remember any technical details...

v1.1 > * Early 1998
* The initial conversion to Blitz. Gave up after a few weeks, because ← ←
  AmigaBASIC is so different it would be
  very unstable if I ever managed to change it.
* Oh, by the way, the reason I wanted to change it was that AmigaBASIC ← ←
  kept crashing on my 'new' A1200!
  Micro$oft, eh?

v2.0 > * May '98
* Totally re-written to Blitz.
* Stores map data in a array/structure type thing
* OK, only one map, but it is loaded from a text file.

v2.05> * Late May '98
* Now has upto 99 maps!
* Upgraded logo
* Maps must be numbered map.01, map.02 ...
* Fixed bug that caused stuff like "You get the 34!" occasionally
* Enemies now have various aggression settings

v2.07> * Late May '98
* Now upto 128 maps (and more would require one number to be changed)
* New piece of code lets you name your map whatever you like! (Loads ← ←
  anything ending in .map)
* Fixed some little bugs
* Fixed bug in enemy aggression handling
* Decided I like making up version numbers ;)

V2.1> * Hopefully no serious bugs left

V2.15> * Added <SILLY:> command to maps to enable comments in the game
* Added comments
* Now says one of four things instead of "You have defeated ..."
* 2Kb smaller

V2.2> * Fixed some small bugs, mostly leaving debugging stuff in which churns ← ←
  out numbers sometimes
* Added <STARTENERGY:> command to set starting energy
* Added <STARTWEAPONS:> command to set starting weapons
* Tells you which map is selected on main menu
* Bug fix: if you had collected ammo for the first weapon, quit level ← ←
  and started again, you'd still have the ammo
```

- * Minor speed increase on loading
- * Now prints dots out so at least you know it's doing something ;(
- * Takes less memory running
- * First release!

V2.21> * Stomped Hang-on-quitting bug : it wasn't my fault! It seems that ←
 compiling it with Blitz's new 'deflibs' file
 causes it to crash on quitting (but it didn't start until the coding ←
 was finished!), but it works fine with the
 older library, so I replaced the few new calls with slightly changed ←
 ones, et voila!
 * Knocked 10Kb off as a result of the above
 * Now uses SpeedVER (don't you just hate it when a program has to search ←
 all through a program to find the version
 string?) - the version line appears within 50 bytes of the beginning ←
 of the program.
 * Proper release (at last)

V2.22> * No longer makes all text in map-files lower case
 * Minor speed increase on loading
 * A few tiddly improvements
 * Added a few... features ;)

I have tried crunching it with 'Turbo Imploder v3.1' (from an OLD coverdisk (←
 Amiga Computing #41!)). It made the executable
 smaller from 79Kb to 25Kb. Almost the size of a C program. And it worked
 perfectly, I tried everything. Then I quit. 'Program DooM2d.executable failed ←
 (error blahblah)'. Bugger. It might
 work on older (<040) systems, but it is still unacceptable. If anyone knows of ←
 a cruncher program compatible with all
 Amigas (preferably FreeWare and with a nice GUI) please contact
 me
 !

That's it!

DooM2d has been brought to you courtesy of:

House of Fonz Software (thats me ;)

in association with...

TACsoft productions

And, of course, AMIGA!

DooM2d I has been tested on:

A500, 1Meg Chip, 2Meg fast

A1200, 2Meg Chip <==== Crashed! But only because of AmigaBASIC ←
 (wouldn't even load)

A1200, 2Meg Chip, 16Meg fast, '040/40 + FPU <==== Worked! HOW???

DooM2d II has been tested on:

A1200, 2Meg Chip, 16Meg fast, '040/40 + FPU
A500, 1Meg Chip, 2Meg fast <==== Very slow map file loading

Plans

- > Map builder software, make it a little easier to make maps:
Write in room text and enemy stuff...
Not much easier really, just more friendly & less likely to make ←
mistakes
- > Have a binary format map file - much quicker to load and saves a little ←
disk space (easier to program too)

More wild...

- > Artificial intelligence - enemies move from room to room looking for you ←
...
> If above is done, multi-player (internet!!) games will be possible

1.4 DooM2d's incredibly high minimum spec...

To run DooM2d II at a good framerate, you'll need at least...

PPC/200MHz
32Mb RAM
Picasso '96
Decent graphics card (P-IV or CV64, ideally Permedia2 3D chip)
70Mb HD space
8 speed CD-ROM
Workbench 3.1
21 inch (or bigger) monitor with 1280*1024 resolution
16-bit wavetable sound card with DSP
500 Watt power supply
Network card with 2Mb/s internet access (for multiplayer game)

But at a push, it'll just run on...

Any 68k processor (or emulation)
512Kb Memory (Probably less)

Useful...

Hard Drive
More memory
Faster processor (it takes an age to load in the levels)

Works fine with...

Graphics cards (no graphics!)
 Sound cards (no sound!)
 060 (should do anyway)
 Other stuff

1.5 Playing DooM2d II

First, a note:

It is very very important NOT TO CLICK THE EXIT BUTTON IN THE DOOM2d CLI ↔
 WINDOW!

If you do, the system is likely to crash with a GURU! (Actually, it doesn't do ↔
 this too much in V2.21, but it will loop
 continuously using 90% of your CPU time until you stop it with ARTM or ↔
 suchlike)

The script that executes DooM2d II should open a window without one, but this ↔
 may not work

on old versions of AmigaOS (below 1.3). To run it, just type "CD <directory ↔
 of DooM2d II>"

[return] "DooM2d_II.executable"

I am trying to fix this, but in the meantime be careful!

The menus are quite self-explanatory:

1: Start a new game	<==	Starts a new game on the level set by option 5
2: Instructions	<==	Gives a concise version of this
3: About DooM2d	<==	Gives some info on the program & a note from our ↔
sponsors, TACsoft		
4: Choose map	<==	Select a map from those loaded in the Maps/ ↔
directory		
5: Quit	<==	Exit the game

(Levels are called maps in DooM2d, no particular reason ;)

Type the number or first letter, or quit or whatever.

Ah, this is really quite simple. Type plain english instructions like the ↔
 following (the case doesn't matter):

North, N or 8	=	go north
East, E or 6	=	go east
South, S or 2	=	go south
West, W or 4	=	go west
Get, G or 9	=	get object in room (if there is one)
Kill, Attack, A, 3 or Murderize	=	kill the enemy (if there is one)
Look, L, or 5	=	repeat description of surroundings
Inventory, I, Weapons or 0	=	list (and choose) weapons
Quit, Exit, End or Q	=	go to menu

Using the numerical keypad is quicker, for the pros (!)

There are a few other things which I'm not telling you ;)

You can't yet save games in DooM2d2 - not that you need to...

Tactics:

As the maps are mostly very different, there aren't many rules that apply to all ↵
, but here are the most important:

- 1: Manage your ammo! Some is rarer than others, and not all weapons need ammo. ↵
Use these against
weaker enemies.
- 2: Beware! Enemies may attack when you try and pick something up or even run ↵
away, others
virtually never attack you. They may spring upon you, too! Also ↵
, just because you got somewhere one
way, you may not be able to backtrack there, and so could be ↵
trapped!
- 3: Remember where you are! If necessary, draw a map.

1.6 DooM2d II Legal disclaimer - or - why not to sue me

I bear no responsibility for whatever the user does with this software, or any ↵
damage it may cause. This software, and
all supplied files are freely distributable under FREeware. That is, the program ↵
can be distributed via any means,
provided that that the price is no higher than the accepted price for that form of ↵
media or whatever, and that the program
and this file remain unchanged. The program, this guide, the DooM2d II logo and ↵
the supplied maps are (C)opyright
Alex Watson 1998. All the other stuff too.

This program is not at all a ripoff of id's Doom. Oh no... I don't know where ↵
you could get that idea...and anyway, the
name is different - it's spelt DooM not DOOM and has '2d' on the end. So there.

1.7 DooM2d custom maps

This is possibly the most interesting part of DooM2d, as you can ↵
have a good laff blasting all your worst enemys away!

It's really simple too, just remember the following:

- 1: A DooM2d map file is just a text file containing information on what the ↵
program should do in each room.
 - 2: Everything is ignored unless contained within square brackets <...>
 - 3: Every < must have a > to end it or DooM2d'll get a little stuck
 - 4: The text in the square brackets must start with a keyword ending in a colon ↵
(:) or it'll be ignored, for example
-

<ROOMS: 30>

5: The file must be terminated with <ENDFILE:> or DooM2d won't know to stop!

6: The commands are separated with the '|' character

7: Text should not start with spaces (not important, but it'll be double spaced otherwise) ←

8: You always start in room 1

9: There must be a space after the colon or strange things will happen ;(

10: Every room needs a <ROOMEXITS: > statement, or you will be stuck there! (← until you quit anyway)

If that sounds complex, believe me, it isn't (ish). Please don't be scared, it's ← much better with your own maps

It is usually helpful to design a map by drawing a grid on paper, numbering the ← rooms, drawing doors etc. then converting

it to a file. An example can be seen with the picture "Example_Map.iff" - click ← it now if you want (MultiView will

open to show it, so you can keep it on the same screen)

If you have read this far and are still baffled (it seems simpler for me, as I ← wrote the program!) then read the

rest, try the example and if all else fails,

send me

a drawing of the map as in 'Example_Map.iff, which ← should be with this software and text for all the rooms, and I'll be more than happy to compile the map for you providing:

- * There isn't too much demand (no, just you)
- * It isn't too huge
- * I can understand the drawing!

Example

Everything is built up from a few commands, as follows:

<ENDFILE:>

Signals the end of your map file.

MUST BE PRESENT

<ROOMS: #rooms>

Tells DOOM2d the number of rooms in your map. Important!

Maximum = 128

<NAME: ...level_name...>

Define the name of your level for choosing it.

<SILLY:>

Enable silly comments during game

<STARTENERGY: energy#>

Sets player's energy at beginning of game. Defaults to 100, must be in range ← 1..200

<STARTWEAPONS: weapon#[|weapon#...]>

Sets the weapons you start with (apart from 0, which you always do). Put as ←
 many as you want (up to 10)
 separated by '|'. Unnecessary for most levels.

<ROOMTEXT: room# | line# | ...text... >

What you want the room to say. Lines 1-7 are always displayed, 8 is only if the ←
 object
 is still there (if there is one), and 9 only if the enemy is there (if there is ←
 one)
 Quotation marks are not necessary.

<ROOMOBJECT: room# | object# | ...object_text... >

Give a room an object. object_text is what it says, eg "you pick up [object_text ←
]".

room# is the room you want it in.

If, when writing ROOMTEXT statements, you include a line #8 (info on the object ←
 there)

and don't put in ROOMOBJECT, then it won't display that line, as it'll think the ←
 object

is no longer there - so it'll make more sense than saying "you see a sword...> ←
 get sword...what sword?"

The object# definitions are as follows...

1 - 10	=	user defined weapons (see WEAPON)
11- 20	=	corresponding ammo for the above weapons
21	=	5 health boost UNLIMITED (never runs out)
22	=	10 health boost UNLIMITED (never runs out)
23	=	5 health boost
24	=	10 health boost
26	=	20 health boost
27	=	40 health boost
28	=	80 health boost
31-40	=	as 21-30, but health can increase upto 200 not 100
41	=	steroids
42	=	suicide
>42	=	does nothing at all (but you can still pick it up)

Hmmm, not the best organised list but who cares ;)

If you're a little confused on this one, here's an example:

<ROOMOBJECT: 1 | 26 | big chunky medikit >

would put an object called a 'big chunky medikit' in room one, giving you a 20 ←
 health boost.

<ROOMENEMY: room# | weapon# | accuracy# | varability# | energy# | agressive# | ... ←
 enemy_name... | ...enemy_weapon...>

This puts an enemy into the room.

Again, if you don't assign one with this on a room where you have put a #9 line ←
 (telling you about the enemy)

it will ignore the line 9, as there will be no enemy.

room# = room enemy will be in

weapon# = how much damage the enemy does

accuracy# = percentage of times it hits, ie 0=never hits; 100=always hits

varability# = percentage of #weapon can vary, ie 0=doesn't vary, damage=weapon#; ←
 100=varies by 100%, damage=0 to 2*weapon#
 put 50 if you can't be bothered

energy# = how much energy the enemy has (same scale as your energy)

agressive# = when will it attack:

```

0...never
1...only if you attack it
2...if you attack it or try to pick something up
3...if you attack it, try to pick something up or leave the area
enemy_name = what it's called eg "you hit [enemy_name]"
enemy_weapon= what to say when the enemy hits you, like "You are singed by the
rabid toaster"

```

```
<ROOMEXITS: room# | north# | east# | south# | west# >
```

```

Determines which room you link to in the various directions, or 0 for no exit
that way.
room# = current room
north# = room that going north takes you to
east# = room that going east takes you to
south# = room that going south takes you to
west# = room that going west takes you to

```

```
<WEAPON: weapon# | ammo# | accuracy# | damage# | ...name... |...text1... | ...
text2... >
```

```

Defines a weapon for you to pick up. You automatically start with weapon 0.
weapon# = weapon, corresponding to the object numbers in ROOMOBJECT. 0-10
ammo# = object number corresponding with the ammo it eats. 0 if no ammo is
needed.
Note that the amount you get is ten less than this number: if 11 you
get 1, 12 gets you 2 etc.
accuracy# = percentage of times it hits
damage# = damage it causes to the enemy if it hits
name = what it's called!
text1 = text displayed before name of enemy if you hit
text2 = text displayed after name of enemy if you hit

```

For example of the use of the texts, with 'a grunt' as the enemy:

```

You shoot a grunt in the guts
A grenage blasts a grunt
A grunt is no more!

```

So, you don't have to write anything for both of these. If you dont write
anything, just leave the
separators together like "numbers...|blah...|blah too..."

It doesn't matter to DooM2d where the commands are in the map file, but it is
clearer if it is arranged:

```

Any comments on the general map
Name
Rooms

```

```
Define Weapons
```

```

Roomtext for room 1
Roomobject, Roomweapon and Roomexits for room 1
Next room...
Endfile

```

The commands don't have to be on one line, and you can have more than 1 per line:

```
<ROOMTEXT: 1|1|Hello you
```

are in a room>
and
<ROOMTEXT: 1|1|Line one> <ROOMTEXT: 1|2|Line two>
are both perfectly legal. In fact, it is good to have very long lines in the file ←
as they will be 'wrapped' to fit the
DooM2d window, so with small windows it won't appear as broken up. Aaanyway...

Once your map is ready for testing put it in the DooM2d 'maps/' drawer with a name ←
ending '.map'
Run DooM2d. If it doesn't load properly (getting stuck on your map at startup) ←
then ensure it complies with the rules above,
the only things that should cause this are:
* Missing out <ENDFILE:>
* Forgetting a terminating '>'

On the menu, press 5 for choose map and choose yours. Then press 1 for start new ←
game and test it out!

If you find yourself stuck in a room, there is probably a problem with the ←
ROOMEXITS for that room, so the program thinks
there are no exits.

Here are some hints to make your map seem more sophisticated...

- * You can have the ROOMEXITS bit allow the player into a room...but not out!
- * You can have several weapons that use the same ammo
- * Secret areas - very simple, just don't mention the route that way, but put it ←
in the ROOMEXITS
- * It doesn't have to be rooms - see "Shopping Spree!" (mall.map) for an example ←
of this
- * You can easily have multiple storeys - just have a room that is stairs or ←
whatever.
When planning this, draw two (or however many storeys) maps and have one room ←
linking them.
- * You can have the effect of something changing when you go in a room by having ←
the exit take
you to a different room than you came in.
- * Teleporters - easy, using ROOMEXITS again.
- * Don't always have medikits, use the unlimited ones for ROOMBOJECTS ond call it ←
a 'spring of life' or something
equally tacky
- * Have an enemy give you energy by putting a negative value for weapon#
- * Have bilingual maps, with room #1 as an introduction, and you go east for ←
german, west for english or whatever.
- * Have various difficulty levels using the same principle
- * Write maps more easily, by instead of using 1,2,42 etc for map and weapon ←
numbers, write in something unique like
#big_north_room and then do a find&replace to exchange this for a number. (←
Careful!)
- * Always draw a map of the level on paper with room numbers - it's much easier!
- * Experiment & look at other maps!

Please spread your map far and wide!

1.8 Example .map file

These here lines are ignored since they have no brackets or keywords!

Use them for writing comments and stuff!

Keywords don't have to be in caps, but it makes it more readable.

```
<NAME: Test map!>
```

Hmmm...a very small map methinks!

```
<ROOMS: 2>
```

Our default weapon...no ammo needed, 70% accurate, called a hand and says 'you ↵
give (enemy) a bit of a slap'

```
<WEAPON: 0|0|70|5|Hand|You give|a bit of a slap>
```

The second weapon...uses ammo #13, 50% accurate, called a cow prod and says 'you ↵
fry (enemy)'

```
<WEAPON: 1|13|50|Cow prod|You fry|>
```

Starting room -

```
<ROOMTEXT: 1|1|You are in the starting room!>
```

```
<ROOMTEXT: 1|2|You can go north from here>
```

This is line 8 which is only shown if there is an object here...

```
<ROOMTEXT: 1|8|There are some cowprod batteries on the floor>
```

In the WEAPON statement, we said that the cow prod used object #13 as it's ammo, ↵
so put some in this room named
batteries.

```
<ROOMOBJECT: 1|13|Batteries>
```

North goes to room 2, can't go east, south or west

```
<ROOMEXITS: 1|2|0|0|0>
```

Room 2 -

```
<ROOMTEXT: 2|1|You are in the second room!>
```

```
<ROOMTEXT: 2|2|You can go south>
```

The object text, displayed only if it is still there...

```
<ROOMTEXT: 2|8|There is a cowprod on the floor!>
```

The enemy, only if you haven't slayed it yet...

```
<ROOMTEXT: 2|9|You see a big square come at you!>
```

Our scary square...in room 2; does 6 damage on you; hits 60% of the time; varies a ↵
bit; has 20 energy; not aggressive, so
you can get the cowprod first; called 'square thing'. Don't need to say what it ↵
does, as it'll never attack.

```
<ROOMENEMY: 2|6|60|50|20|0|Square thing|>
```

Frymaster 2000 in room 2, object #1 (set with the WEAPON at the beginning)

```
<ROOMOBJECT: 2|1|Frymaster 2000 (tm)>
```

You can only go south, into room#1...

```
<ROOMEXITS: 2|0|0|1|0>
```

n e s w

DONT FORGET THIS!!!
<ENDFILE:>

1.9 All about me and my favourite things

Hi. My name is Alex Watson, I live in Selsley which is a crap village near ↵
Stroud in south west england. Satisfied?
I am 15 years old (1998) and am at school at Marling (see the DooM2d map of the ↵
school - including teachers. Nrrrrr)
I am not yet sure if I am year 2000 compatible.
I like saying "Spam!", "Drinck!", "dodgy" and "hey, fool!" for no apparent reason
I have:

A1200 (not tower yet) stuffed full and posing a fire risk with
APOLLO A1240 board @40MHz - 040 Rulez!
16Mb SIMM
EZ-IDE interface + wire sticking out side
EZ-VGA flickerfixer (good but only 16bit I think)
540Mb Hard drive from Eyetech
crammed into the standard case and...
4xCD nice and cheap from Eyetech too!
15 inch VGA monitor
Epson Stylus Color 600 printer
Second floppy drive
Mouse * 3 (all sticky)
Joystick * 2

A500 (now retired ;(
20Mb HD
3Mb RAM

As part of TACsoft (i.e. me + two friends (Tom Williamson & Chris Wiseman) we are ↵
currently working on:
Command & Destroy - a C&C clone.
Escape from the Tri Islands - a Monkey Island clone with sampled speech.
Wangle - Space trading type game (strange), out before ↵
previous two
More DooM2d maps
DooM2d III? This may go ahead if there is enough (any) demand! Please ↵
tell me what you'd like in it!

Fave games:

Command & Conquer	(no amiga ;(
Doom	(loadsa extra stuff, and PC friends play it too)
Quake	(PPC?)
The Settlers	(Brilliant.)
Myst	(when's Riven coming out?)
F\A-18 Interceptor	(dodgy on '040)
Napalm	(could be faster...)

DooM2d II web site:

This is now operational, and can be found with the other TACsoft pages on Wangle ←
! and CrappyArc, and info and (soon) pics
of us lot. The DooM2d II support site features a comprehensive list of levels (←
please send us yours) and
always the latest DooM2d2 upgrades and buglists (hopefully not too many!)
Point your browser at

<http://freespace.virgin.net/jim.watson1/doom2d2/index.htm>

Please contact, for help / bugs found (other than the one mentioned in the playing ←
section)
/ ideas for DooM2d III at:

E-Mail: tac.soft@virgin.net

Alex Watson ←

|
The Stirrups ←

|
Stanley Park ←

|
Selsley ←

|
Stroud ←

|
GLOS. ←

|
UK. ←

|

1.10 About the included DooM2d II levels

In this archive the following levels are included:

+Dingisten1.map

Name: Dingisten (Quake clone)

Description: Inspired by Quake, featuring the obligatory dingy castle and ←
enemies like grunts and ogres. Pretty
standard blasting level, further Dingisten levels will include more traps/ ←
puzzles. Plenty of secret places.

Hints: The lift will only go down...

Difficulty: Medium

Boss: Yes

+Mall1.map

Name: Shopping Spree!

Description: You are outside a Mall, and for some reason you must get ↵
through it. Facing on the way many
ordinary objects turned evil murder machines, like the psycho washing ↵
machine, really annoying little car and out of
control lawnmower!

Hints: Wander around outside picking everything up before going in, or you ↵
won't stand a chance. Use your
weapons wisely and conserve ammo.

Difficulty: Medium

Boss: Yes

+MKru.map

Name: TACsoft kru special level!

Description: Just a small level created for a laugh one day. Features some ↵
people from our school, probably
pretty cack for anyone else. Oh well...

Hints: None necessary.

Difficulty: Easy

Boss: No

+.map

Name:

Description:

Hints:

Difficulty:

Boss:

Remember that more levels can be obtained from the DooM2d II support site at:

<http://freespace.virgin.net/jim.watson1/doom2d2/index.htm>
