

A Myst Demake for the Apple II

Vince “DEATER” Weaver

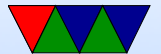
vince@deater.net



University of Maine

Kansasfest 2020 — 24 July 2020

Are there any Maine Apple II Connections?

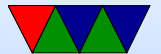


UMaine famous alum – Chuck Peddle

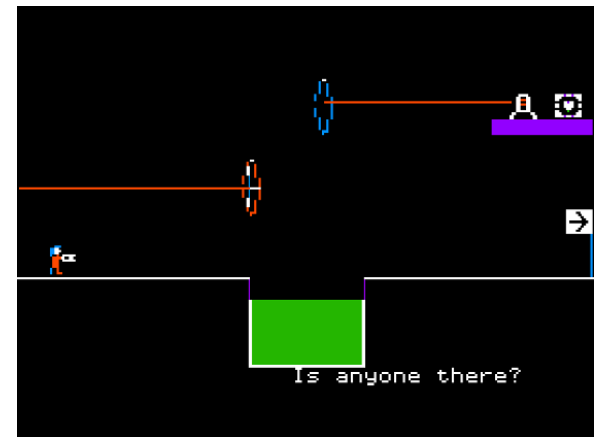
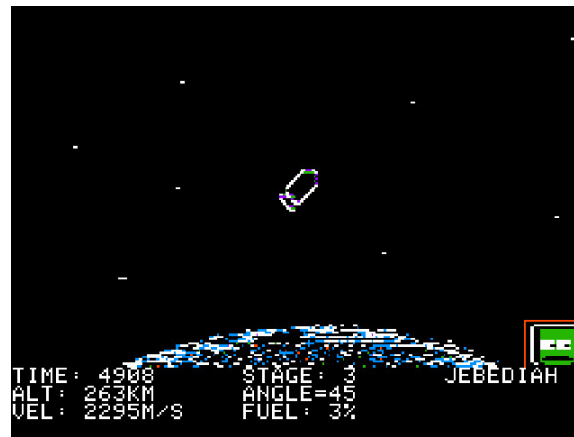
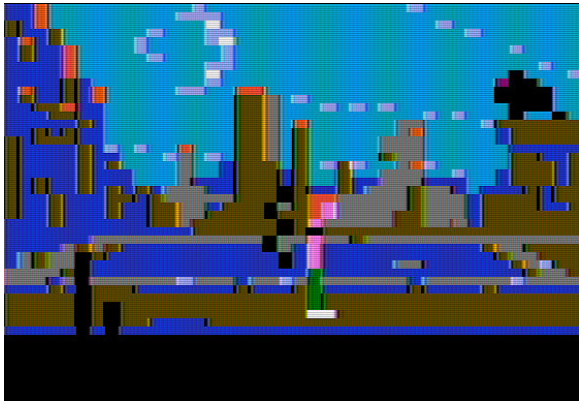
Designer of the 6502 Chip



6502 Exhibit



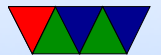
Apple II Demakes and Demos by Me



Background on Myst



- Early CD-ROM game, 550MB
- Point and click, pre-rendered graphics
- Cyan Inc (Miller Brothers) 1993
- Originally for Mac with 256 colors, HyperCard
- 6 million copies sold

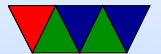


How Do You Fit a 1993 CD-ROM Game onto an Apple II from 1977?



System Requirements

- Apple II/II+/Ile (also IIfx/Iigs, not well tested)
- Disk II drive
- 48k RAM (64k required for digitized sound effect)
- Mockingboard for music in intro

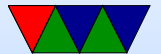


Status

- Game fully playable from intro to various endings
- The main island is more or less complete
- Some of the puzzles in the sub-ages are not quite done yet.
- Some, like the submarine puzzle, probably never will be



Broken up into 15 Segments (RAM)



Game Engine

- Overall fairly simple
- Wrote from scratch by playing game, did not investigate original game code
- Written entirely in 6502 assembly language



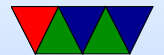
Main Game Loop

- Copy background from \$C00 to off-screen page
- Draw foreground items/animations (optional)
- Draw hand cursor: (special cases: edge of screen check for valid directions holding pages, matches, key)
- Graphics Page Flip
- Check keyboard, Move Cursor
- If clicked, change location/bg, or special action
- If click takes us to new level, exit back to loader



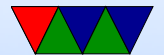
Some Common Helper Routines

- Decompress background in LZSA to \$C00
- Fast memory copy to \$400 or \$800 (lores page1/page2)
- Sprite drawing code
 - Cheats and Y must be even (code much simpler)
 - Handles transparency, Grey2 is transparent color (so we can have black in sprites)
- Text is often stored in graphics, as LORES/TEXT are equivalent in Apple II

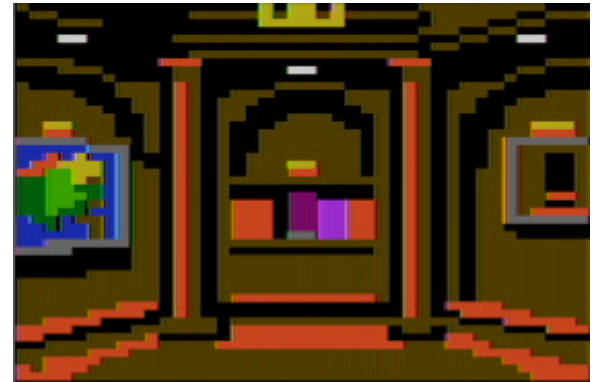
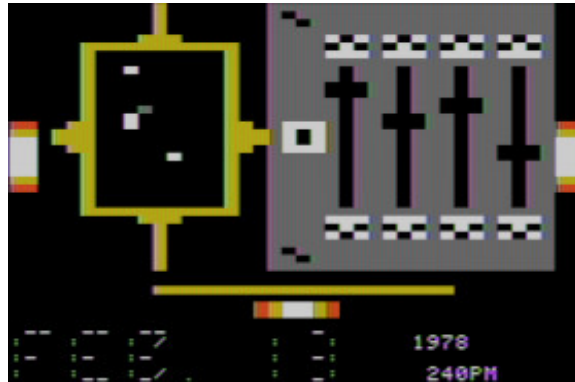


Series of Connected Rooms – Example

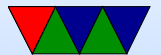
```
; MIST_CLOCK                -- clock
location11:
    .byte    MIST_TREE_CORRIDOR_5    ; north exit
    .byte    MIST_CLOCK_PUZZLE       ; south exit
    .byte    $ff                     ; east exit
    .byte    $ff                     ; west exit
    .byte    DIRECTION_N             ; north exit_dir
    .byte    DIRECTION_S             ; south exit_dir
    .byte    $ff                     ; east exit_dir
    .byte    $ff                     ; west exit_dir
    .word    clock_n_lzsa             ; north bg
    .word    clock_s_lzsa             ; south bg
    .word    clock_e_lzsa             ; east bg
    .word    $0000                   ; west bg
    .byte    BG_SOUTH|BG_EAST|BG_NORTH
    .byte    DIRECTION_E             ; special exit
    .byte    5,30                    ; special x
    .byte    2,46                    ; special y
    .word    enter_channel_clock-1    ; special function
```



Complex puzzles

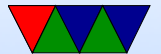


- Mixed Text/Graphics (Generator and Planetarium)
- Rooms with more than 4 exits (library)

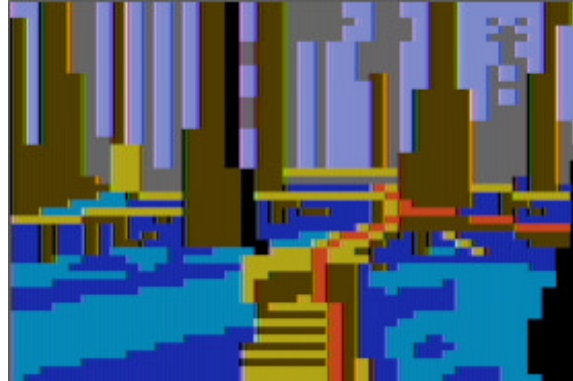
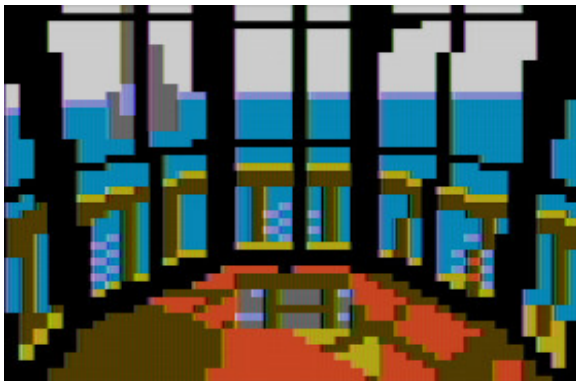
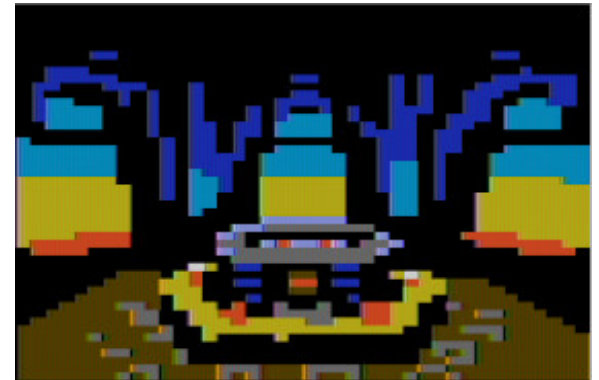
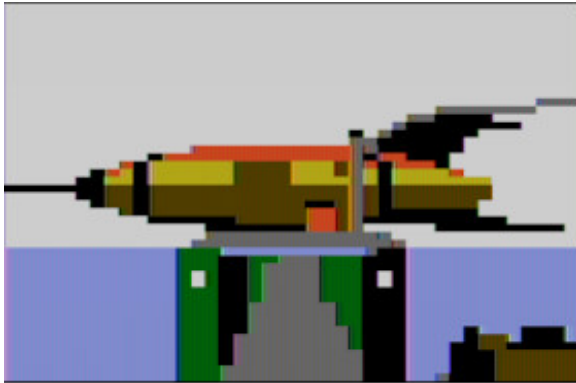


Input Issues

- Keyboard
 - Always a challenge on Apple II
 - No up/down on II+, so also allow WASD
 - Can't tell how long key being held (needed for example in the gear puzzle)
- Joystick – TODO. Latency?
- Mouse – don't have one?

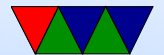
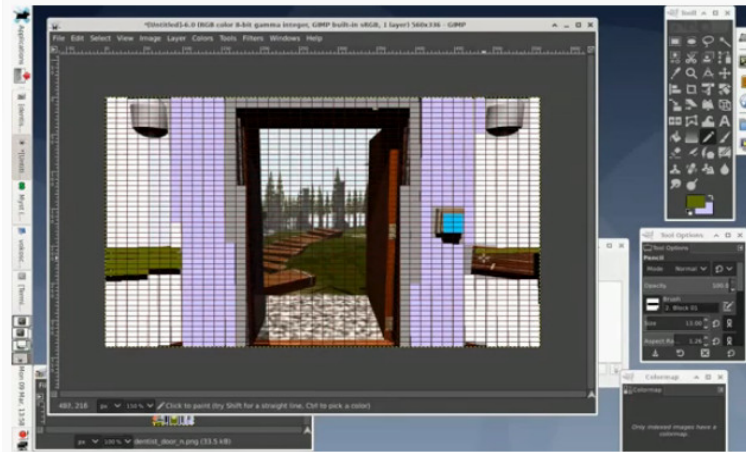


Currently 729 Background Images



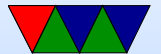
Graphics Design

- All are hand-traced over screenshots from game
- Slow. Can watch a video of the procedure here:
<https://www.youtube.com/watch?v=6XNu3v8rukQ>



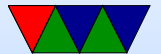
Most common question: Why not use better graphics?

- I like lo-res graphics and have existing code for loading, animating, sprites
- I want this to run on an Apple II+
- I want to fit as much into RAM as possible to avoid disk accesses



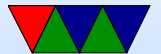
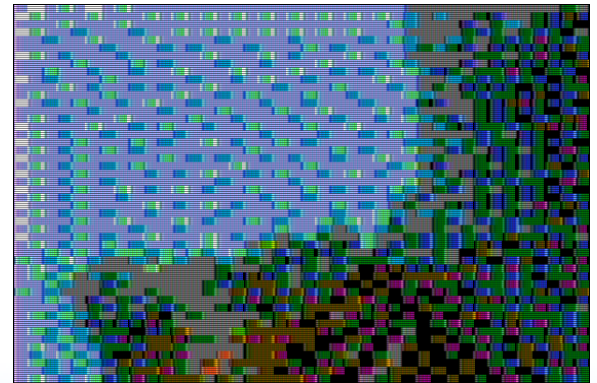
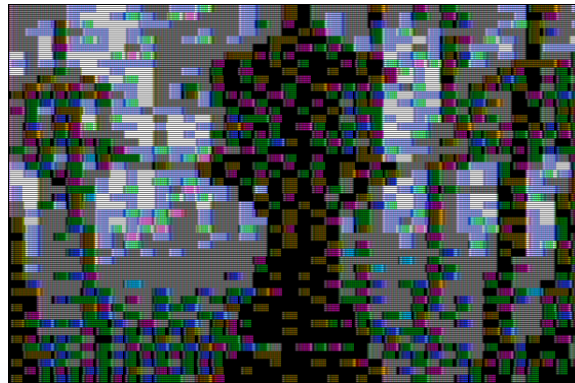
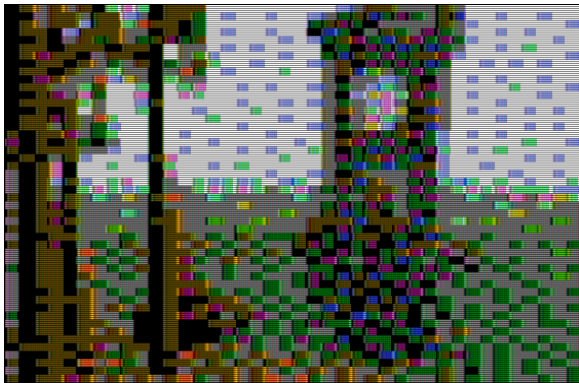
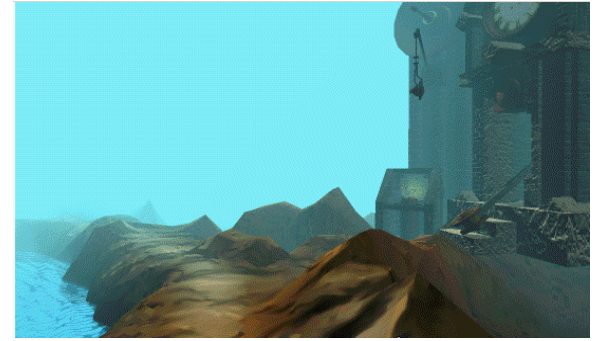
Graphics Examples

- AppleWin 1.29.12.0 with 50% Color TV
- Generated using Bill Buckels' Bmp2DHR v1.0
Using -d1 (Floyd-Steinberg dithering)
- SHR with GSplus 0.14 and SuperConverter 4.0



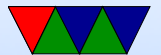
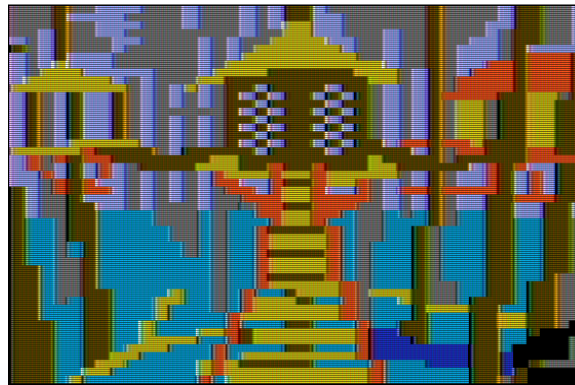
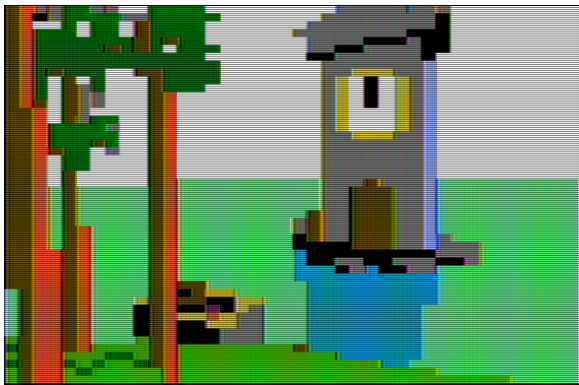
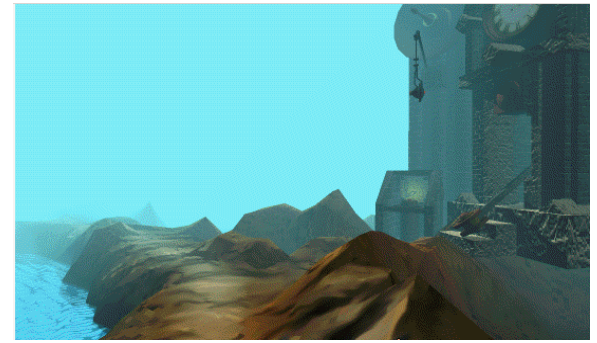
Lo-res (automatic)

40x48, 15 colors. ~1kB



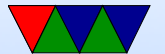
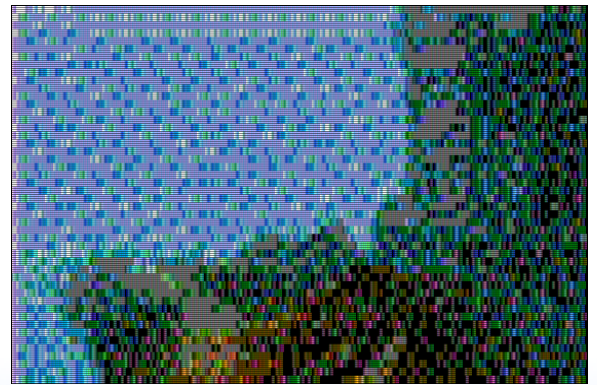
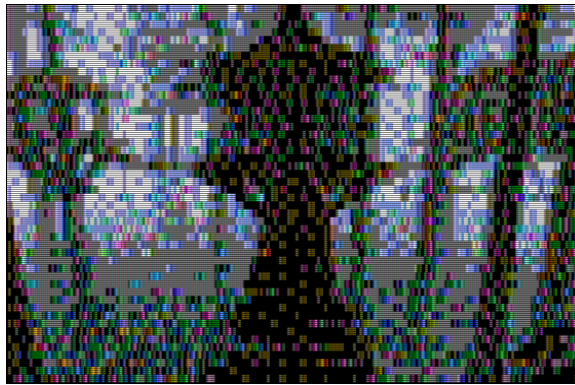
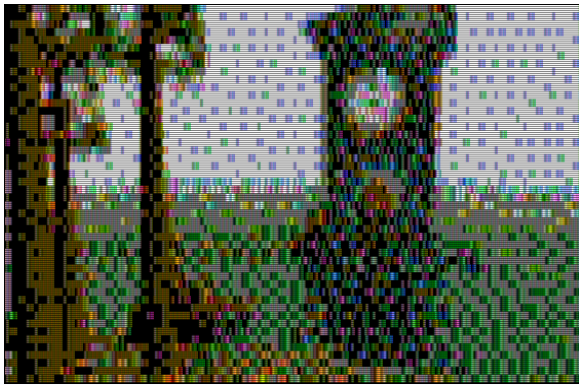
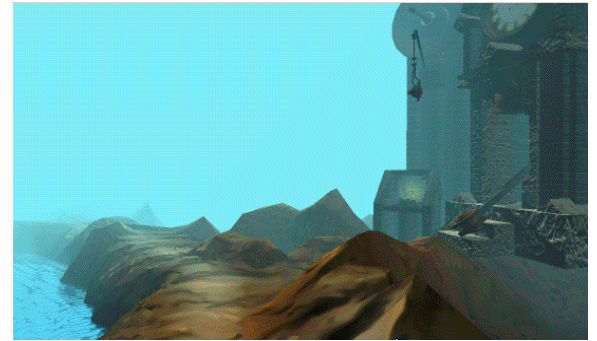
Lo-res (hand-drawn)

40x48, 15 colors. ~1kB



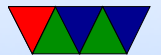
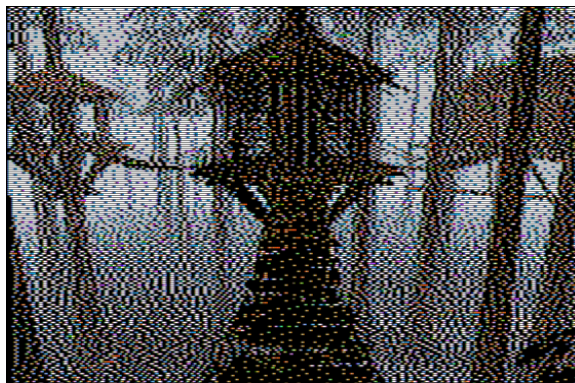
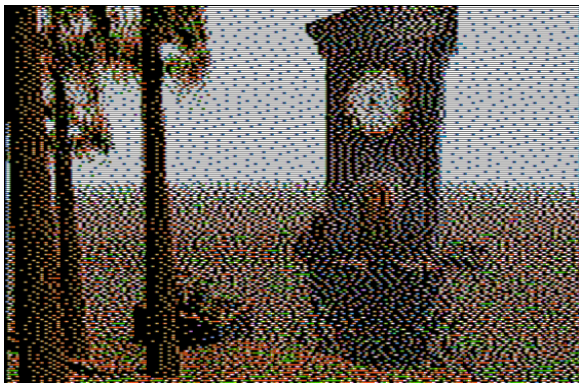
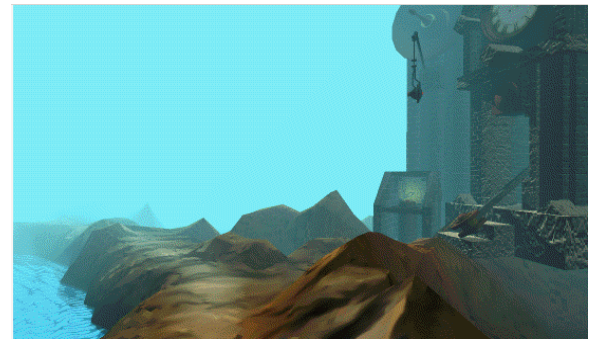
Double Lo-res

80x48, 15 colors. ~2kB, Ile/80col card, complicated code



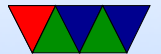
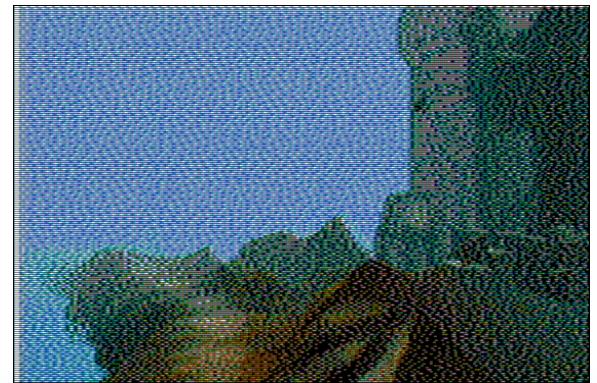
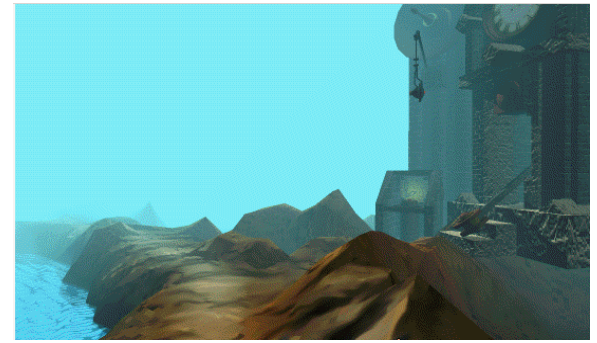
Hi-res

280x192, 6 colors. 8kB



Double Hi-res

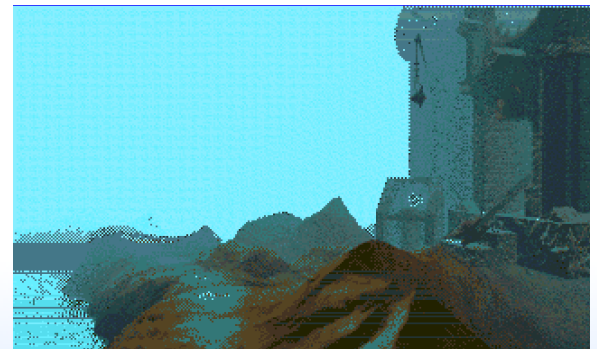
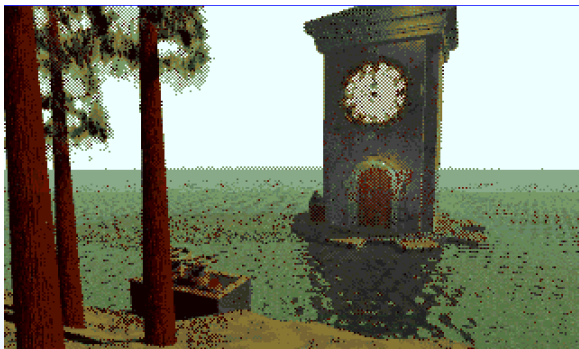
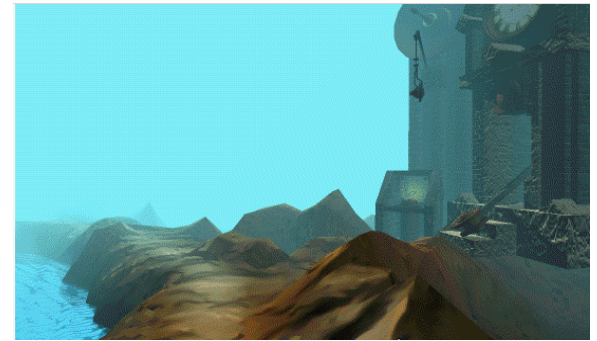
280x192, 15 colors. ~16kB, 128k Ile, complicated code



Super Hi-res

320x200 or 640x300 4096 colors, 32kB, IIGs

IIGs was contemporary with systems that saw real ports



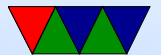
Minimizing the number of 140k Floppies

- Compression
- Lo-res images
- Fewer areas of the game
(generally skip every other step)
- Simplify some corner cases
(e.g. pentagon platforms in channelwood)
- This also helps fit in RAM better, swapping floppies and waiting 45s to load takes you out of game



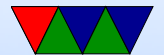
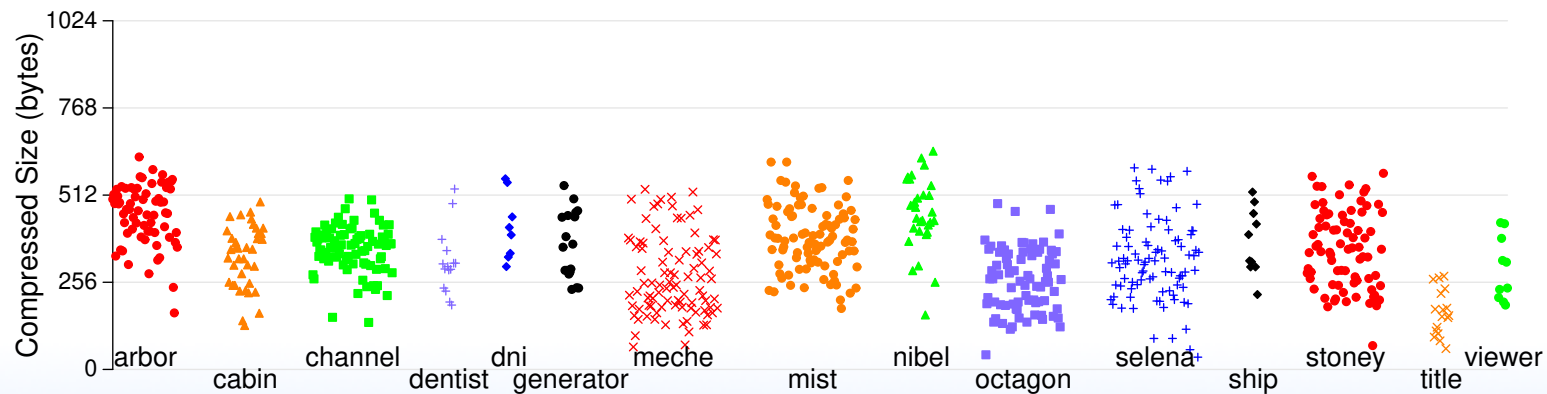
Potential Ways to Get Even Smaller

- Simpler graphics (that compress better)
- Generate backgrounds algorithmically (pillars area with distinct backgrounds)
- Some paths in game are mirror-images, have some sort of flag to specify flipped background?
- Crazy disk hacks
 - Use custom booter/RWTS to save 12k from dos33
 - RWTS18 disk format (Gustafsson) 157k on disk



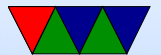
Compression

- Uses LZSA2 compression library by Emmanuel Marty
<https://github.com/emmanuel-marty/lzsa>
- Qkumba had some part in the optimizations
- Graphics compress down to only 33% of original size



Disk I/O and Load/Save Games

- Levels are split up as max size is 40k
- DOS33 filesystem but not DOS33 (too big in RAM)
- Use Qkumba's RTS
- He made write code; haven't had time to use it yet
- Load and Save games
 - Loading is straightforward, state is in zero page and is only 74 bytes (\$80-\$CA)
 - Save is TODO but definitely possible



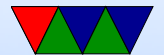
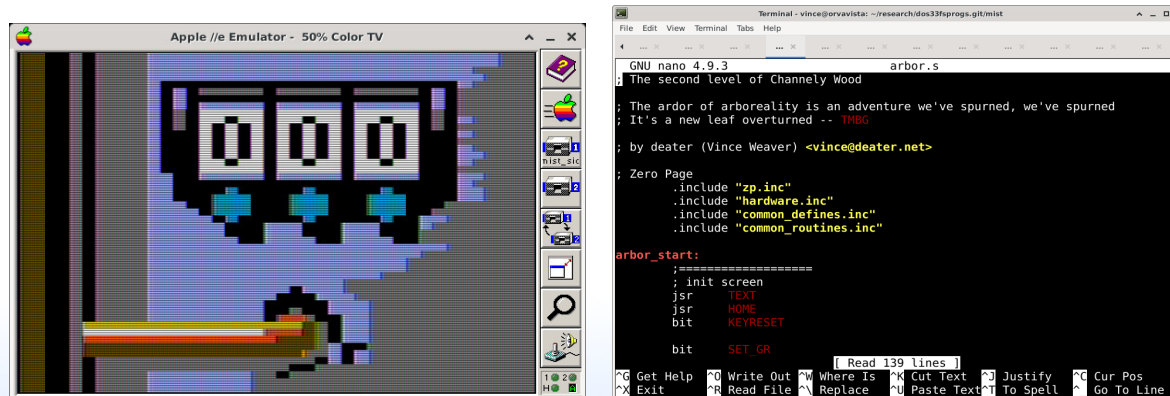
Myst has a lot of Sound

- Many sound puzzles, hard to implement on Apple II
- For simple clicks/tones just use the speaker
- Digital sounds are large, use language card.
- Currently linking noise (12k) with the 4k possibly for “bring red page”
- Use “BTC” player by Oliver Schmidt.
- If you have a Mockingboard a PT3 rendition of the Myst theme plays during the intro using my pt3-player



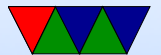
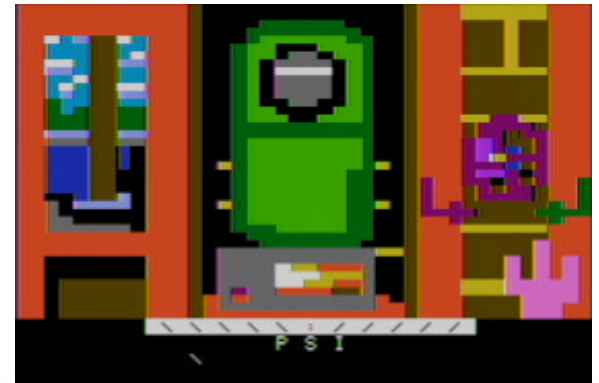
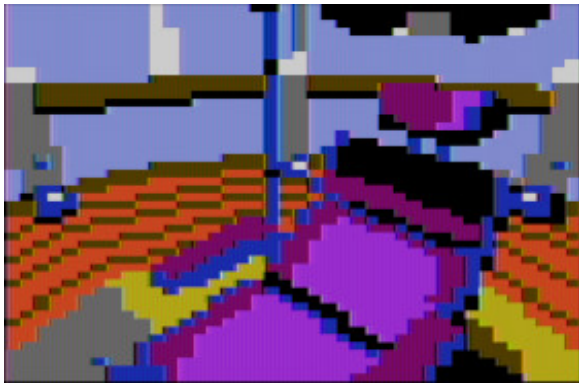
Development

- Develop on Linux
- Graphics in Gimp plus custom tools
- Assembler is ca65 from the cc65 project
- nano text editor
- Testing in AppleWin under Wine on Linux



Future Work

- Hooking up rest of puzzles, especially Stoneship
- Some of the missing locations, especially Channelwood
- Various missing animations
- Sound and music



Questions?

`vince@deater.net`

More Info and Sourcecode

Mist/Myst Apple II Demake

`http://www.deater.net/weave/vmwprod/mist/`

Sourcecode

`https://github.com/deater/dos33fsprogs/mist/`

