D&D Paperless Adventure Experiment
Owen Henderson, May 14, 2007

**Objective**
To assess the benefits, disadvantages and methods involved in running a D&D adventure entirely on a PC.

**Equipment**
Personal computer (Toshiba Satellite P25 laptop w/ XP Professional ver. 5.1.2600 SP1 Build 2600 operating system) running the following software:
1. Microsoft Word 2000 (ver. 9.0.4402 SR-1)
2. Microsoft Excel 2000 (ver. 9.0.4402 SR-1)
3. WinAmp ver. 5.093
4. IrfanView ver. 3.91
Two documents, “adventure.doc” (info, NPC stats) and “adventure.xls” (map, combat spreadsheet)
Numerous music and sound effect files
Game dice
Props: A piece of paper with a rhyme written in symbols, and a scrap of paper to decipher the code with Labtec Pulse 424 computer speakers and subwoofer (LCS-2424)
**Note:** The only paper used was for player character sheets.

**Setup**

**Documents**
Adventure map was drawn using borders and highlights with Microsoft Excel. The same workbook was used to track combat in a special spreadsheet. Adventure details (NPC stats, key dialogue, etc.) were compiled with Microsoft Word in “open” format (see “Open Format” Adventuring). WinAmp was used to play background music in MP3 format; sound effects (WAV and MP3 files) were played using IrfanView.

**Equipment & Sound**
The DM took the seat at the far end of a rectangular table with speakers flanking computer (facing away from DM) and sub-woofer positioned underneath table. Sound test was conducted before adventure; speaker volume [dial] was set to 50%, sub-woofer 40%, computer sound level 100% and WinAmp at 31%. The Toshiba P25 is equipped with an external volume dial that was used to tweak sound level throughout the session. WinAmp volume was set to achieve desired balance with sound effects volume.

**Display**
Computer was set to highest resolution (1440x900) for maximum visibility of material. Word document was set to “Normal” view, with landscape orientation and margins stretched to make optimum use of screen size without being obstructed by WinAmp’s display. Excel and Explorer windows were maximized. IrfanView window was shrunk and placed in the upper left corner to avoid overlapping the file list portion of Explorer.

**Adventure Flow**
This section is included so reference materials can be more easily understood.

**Background**
The previous adventure was intended to be a “one-shot Great Escape”. The party was sent to an island penal colony to work as slaves with the implied objective of escape. After many failed jailbreak attempts,
they succeeded in killing the guard leader (a minotaur) with a single critical hit to the head with a quarterstaff. The minotaur’s horn was cut off for use in a ritual, and the head severed to be used to intimidate other guards (and later discarded). The party recovered their equipment and looted the treasury for good measure before escaping the island by boat. To conclude the single adventure on an entertaining note, the DM (not the author) included a Deck of Many Things in the treasury, although the results of the draws were nullified. The characters were meant to be discarded after the adventure was completed.

The document used (“adventure.doc”) was written as a follow-up to the above scenario, in addition to explaining many of its discrepancies. Two PC’s were converted to NPC’s, the DM’s former PC (Alesander Calenora) and an evil bard (Laughlin). The Deck of Many Things was replaced with the Deck of Moggordeon (a deliberate anagram of Demogorgon), a collection of 22 cards containing the essences of 22 demons. The artifact altered the party’s memories (causing them to confuse the two decks) and cursed them with double-edged powers, giving them incentive to go back to the island. After being rescued by a local temple, the Count tasked with overseeing the penal colony orders the characters to re-secure the island, on the grounds that a less-expendable prior expedition (sent while the party recovered) was almost wiped out.

The Adventure

Because the synopsis itself was an unanticipated follow-up to an ad hoc single adventure, the plot hook (cursing the party) was deliberately aggressive. Predictably, the players did not enjoy being forced into the situation.

NPC’s were used to nudge the characters in a productive direction; overall, the players required very little direct intervention. They successfully researched the Deck of Moggordeon before going to the island, recovered the artifact, and delivered it to the temple’s High Priestess (not realizing she was the true villain of the adventure until afterwards).

Use of Operating System (Managing Content)

The taskbar was heavily used to rapidly switch from one program to another. All programs involved remained stable throughout the session, so crashing was not a concern. However, creating links to all programs in the Quick Launch menu in case a program crashes or terminates is highly recommended. Explorer was used to manage sound files (background music and sound effects). The files themselves were displayed in list format to make optimum use of screen space.

Overall, switching back and forth between programs was easy. Windows XP’s “Show Desktop” icon in the Quick Launch menu was used as a convenient “panic button” on the rare occasion when overlapping windows caused confusion.

Use of “Open Format” Adventure and Microsoft Word

“Open format”, “ad hoc” and “module” are terms used to distinguish the adventure format used in this experiment from that of conventional adventure modules and off-the-cuff DM-ing.

The open format works well with paperless adventures, but by itself it has nothing to do with the paperless method. It was originally conceived as a solution to the disadvantages of both off-the-cuff (ad hoc) and modular adventuring. Ad hoc adventuring, by definition, lacks preparation. Thus, even in spite of the availability of supplemental materials, much in-game time is wasted on designing locations and encounters on the fly. Ironically, the lack of pre-accumulated content results in less flexibility as the DM sacrifices creativity to keep the game moving, especially under pressure and/or when supplemental materials fail to help. Tracking details as they are invented quickly become cumbersome. Also, off-the-cuff statistics create tension between the DM and the players when [non-standard] details are determined to be
conveniently just beyond the party’s capabilities (e.g., a door’s Break DC is set at 27 when the highest Strength bonus in the party is +5). On the other hand, adventure modules require the DM to contain the players within them. Players have more freedom to move around and act, but only within the adventure’s microsphere. In short, off-the-cuff adventures offer more macro-level freedom at the expense of fleshed-out detail, while modular adventures offer micro-level freedom within a set of strict boundaries. (Scripted modules are the worst-case scenario and will not be discussed here.)

In open format adventures, key maps are used like those of adventure modules; supplemental maps are used when necessary. Unlike modules, the adventure’s content is nothing more than a collection of key facts (e.g., Skill Check DC’s or references to the Player’s Handbook), NPC statistics and information, and initial conditions. Events are not scripted; the adventure begins with the set of initial conditions & NPC objectives and is allowed to flow causally from there. The only forces driving the party in any direction are the thoughts and actions of fleshed-out NPC’s. This results in less on-the-fly design than ad hoc adventures while offering a greater degree of flexibility than standard modules. Because events only happen within the scope of PC’s and NPC’s interacting with each other, there is little perception of deliberate scripting of events.

Because NPC’s are so critical to open format gaming, a specific format is used to create and summarize them:

Name, Title/Occupation (any current status effects)
Visual description (include race, gender, physical size, appearance, clothing and demeanor)
Nth-level AL Race Class, S 10+2, D 10, C 10, I 10, W 10, C 10, BAB +0, Saves F+0/R+0/W+0, Init+0, HP 10
Protective Equipment (AC+10/T+10/FF+10, ACP – 0, Mv 30ft.)
Attack Mode (Hit +0, Dmg Effect, Crit 20/x2)
Gear (0/0gp):
Skills (0/0):
Languages (0/1):
Feats (0/1):
Powers:
Spells (0/0/0/0/0/0/0/0/0/0/0):
NPC notes, initial conditions and objectives
DIALOGUE TRIGGER
“Key dialogue”

Attributes, race, class and level are determined (in that order), then used to populate all other fields. Points (skills, feats, etc.) are calculated, then spent. Statistics are updated as necessary.

Open format adventures are compact by nature, generally requiring only 2-3 pages of space to compile. This gives it an advantage in paperless adventuring, where visibility of detail is limited by screen size.

Microsoft Word, Font Format and Hyperlinks

In addition to using “open format”, paperless adventuring heavily utilizes Microsoft Word’s font color function. Typing new content on-the-fly was unproductive; organized detail is critical in maintaining adventure content easy to scan by eye. Instead, content was tracked by varying font color. For example, expended NPC spells were changed from black (default) to grey. Key and/or temporary details were marked in red; hyperlinks were blue by default. Excessive use of color resulted in diminishing returns; the number of font colors was limited to four (black, blue, red, gray) for optimum results. When further breakdown was necessary (e.g., NPC information), font formatting such as boldface and italics were used but not changed mid-adventure.

Because the Word document was not in printed form, its dynamic nature was further exploited through the use of hyperlinks. Links to appropriate sound files were created within the document so the DM could
quickly create a specific mood. For example, when a recovering party member asked an NPC what he was doing recovering in an infirmary, the dialogue trigger “IF PARTY ASKS WHAT HAPPENED” was quickly clicked and “OK-ed” to initiate suspenseful music prior to the NPC explaining the character’s mysterious circumstances.

**Note:** Microsoft Word hyperlinks pointing to a file within a computer reference the file’s path on that machine. They will not work if the file is moved, changed or deleted. Similarly, they will not work if the document is copied to and opened on a different computer.

Figure 1. A hyperlink pointing to a music file in the computer. Hovering over a link with the mouse reveals its file path.

### Advantages & Disadvantages of Dynamic Content

When relying exclusively on an open Word document, changes were relatively easy to track without the use of scrap paper, as combat-related changes were tracked using Excel (see Tracking Combat) and non-combat changes to detail were rare. Like most adventures, the party’s general progress was monitored mentally. Although only a small portion of the document was visible at any one time, searching for detail through pages of material was not a major concern. However, this latter benefit was partially due to the open format used.

Unlike a printed module, an active document is vulnerable to the “fat-finger” effect. That is, it is possible for entire sections of the document to be altered or deleted through a short sequence of inadvertent mouse and/or keyboard motions. When this occurs, the DM must either quickly undo (Ctrl-Z) the change or close and re-open the document.
Figure 2. When a hyperlink is clicked, a dialogue box opens and is “OK-ed” for fast, easy opening.

Figure 3. To create a hyperlink, highlight the text and select the “Insert → Hyperlink” function.
Use of Microsoft Excel

Adventure Map

Creating maps using borders, text and highlights in Excel was tedious, but not any more or less cumbersome than any other method. Drawing by hand on paper is faster for quick, simple maps, but Excel was much better at accommodating numerous changes as maps were fleshed out. The final map was relatively clean despite dozens of last-minute (pre-game) changes, and providing a “players’ version” sans DM’s details was relatively easy to make. Balloon autoshapes were created for the purpose of tracking individual character (PC and NPC) positions, but proved unnecessary. In the end, NPC locations remained relatively static while the party’s location was adequately tracked using a single text box.
Figure 5. Adventure map. Separate spreadsheets were used for different maps, to good effect. The green and red balloons shown were not used; the yellow “party” box tracked player location instead.

Tracking Combat

Tracking combat with Excel involved a sharp learning curve, but eventually turned out to be far more efficient and organized than hand-scribbled notes – especially for large, complex encounters. After many on-the-fly adjustments, the format effectively used was a spreadsheet containing four vertical sections. Only the first three need to be always visible:

1. **Initiative Phase Tracker.** This section monitors initiative order and NPC current hit points. Initiative order was quickly determined using the “Data → Sort” function. Current initiative phase was tracked with an “x” in the far left column, used exclusively for that purpose and thus left otherwise blank. **Note:** This was a key in-game adjustment. Until the “x” column was used, tracking initiative was chaotic and many phases were skipped or done out of order. Because players tracked their own hit points, they were fixed at “999” to avoid confusion. Defeated characters had their hit points blanked (deleted). When a character was disabled or defeated, their initiative cell was blanked. The Conditional Formatting function was used to automatically darken disabled/defeated characters for further ease of tracking. As actions were
resolved, the DM switched back and forth from the Combat spreadsheet to Explorer and IrfanView to play appropriate sound effects (see Execution of Sound Effects).

2. **Combatant Stat Section.** The second section contained combat-related stats of those participating in the current encounter: HP (Hit Points), Init (Initiative Modifier), AC/T/FF (Armor Classes), F/R/W (Saves), Hit (Attack Bonus), Dmg (Damage), Crit (Critical Rate/Effect), Move (Movement Rates), and Misc (Miscellaneous Notes). If they were NPC’s, their stats were quickly copied and pasted from the fourth section on the far right part of the spreadsheet. For random encounters et. al., a blank stat section was copied from the fourth section and quickly filled using the Monster Manual. This second section became necessary after relying exclusive on the far right section proved too cumbersome and lacked space for random encounters.

3. **Combat Round Tracker.** This simple section was used to track temporary effects lasting longer than one round. Use of highlight to track rounds proved cumbersome and was quickly abandoned for an “x” column similar to that of the Initiative Phase Tracker (see above).

4. **NPC Quick Stat Section.** See “Combatant Stat Section”, above. This section contained combat-relevant statistics for all NPC’s in a compressed format. It proved useful after relying exclusively on it was abandoned in favor of creating the Combatant Stat Section.

![Figure 6. Combat spreadsheet sections. Use “Data → Sort” to organize Initiative order.](image)

Combat flow was roughly managed as follows:

1. Combatant Stat Section was quickly filled out using Copy & Paste functions and Monster Manual.
2. DM and players rolled for initiative, Initiative Phase Tracker was quickly filled and sorted.
3. Initiative order was resolved using “x” column to track progression. Sections 1, 2 and 3 were updated as necessary.
4. End of round – “x” was moved in Combat Round Tracker. If combat conditions changed dramatically, initiative was re-rolled and the Initiative Phase Tracker was re-sorted.

**Advantages & Disadvantages of Using a Spreadsheet**

The spreadsheet offered no distinct advantage over other map-making tools. However, tracking the party’s location on and changes to an existing map were much easier using Excel than conventional...
methods. Balloon autoshapes turned out to be unnecessary during the adventure but have potential for tracking unusual situations (e.g., a single PC breaking away from the party). Highlights were used to indicate areas of poor illumination, but this was not an advantage of Excel so much as a variation of mapmaking. Descriptions of encounter areas were placed directly on the map; in hindsight, the conventional numbered description system used in modules is superior. Fortunately, that format can be easily integrated into a spreadsheet map.

The Combat spreadsheet took some time to adjust to, but the end result was a success. Sorting and re-sorting initiative was a distinct advantage, but in its initial form, it was worse than using paper to track combat timing. When the “x” columns and Combatant Stat Section were added, the timing problems quickly disappeared.

Background Music and Sound Effects

In addition to key pieces triggered using Microsoft Word hyperlinks, the DM used a combination of Explorer, WinAmp and IrfanView to simultaneously run background music collections and specific sound effects.

Execution of Background Music

When not using hyperlinks, background music required “fire and forget” functionality. It needed to be easy to maintain, easy to use, and not awkwardly change styles or stop. To achieve this combination, WinAmp was used with some pre-game preparation. First, the "Show Winamp in folder context menus in Windows Explorer" option in "Preferences → General Preferences → File Types" was checked. This allowed WinAmp to open folders directly without the use of playlists (which are cumbersome to maintain). The “Shuffle” and “Repeat” options were also enabled to avoid redundancy or awkward stops, respectively. Music files were categorized in folders by style (the breakdown required extensive preparation but only needs to be done once and displayed in Explorer).
During the adventure, when a certain style of music was needed, the folder was right-clicked (to open the context menu) and the “Play in Winamp” action selected. When necessary, volume was manually controlled for fade-in and fade-out effects.

**Execution of Sound Effects**

Sound effect files were kept in a special folder that was kept open in Explorer’s main window at all times. To play a sound effect, the DM clicked on Explorer, then IrfanView in the task bar, then dragged and dropped the appropriate file onto IrfanView’s window. Because IrfanView opened any file dragged onto any part of its window(s), the program window could be kept small and using it was a simple process. Also, IrfanView allowed stable yet simultaneous playback of music in WinAmp at a different volume.

![Figure 8](image.png)

*Figure 8. To play a sound effect while WinAmp is running, drag & drop the file into any part of IrfanView’s window(s).*

**Effect of Music and Sound on the Gaming Experience**

Player reception of music was lukewarm – not the music itself per se, but the use of it. Certain pieces of music were effective in conveying a certain mood, such as when the DM wanted the PC’s to regard a certain character or situation seriously. Also, although players were verbally, the pace of combat noticeably increased when music was utilized.

On the other hand, sound effects were an immediate hit. Sounds such as a secret door opening did not need to be described, although overall it involved extra work to integrate sound effects into an adventure. That said, and despite no visible effect on player reaction, the feedback was overwhelmingly positive. Use of sound effects in combat was thoroughly enjoyed, especially critical hits and explosions.

On a subjective yet relevant note, the group in this experiment placed heavy emphasis on “kill & loot” gaming. This at least partially explains the group’s reaction to mood-setting music vs. “splat/boom” sound effects.
Conclusions

Overall, the paperless adventure was a success. Combat was awkward at first but the adventure otherwise ran smoothly. Music and sound effects were integrated seamlessly into the experience, with generally positive results. Combat positioning was eventually tracked using the cheap, conventional “dice on a grid” method. Room for rolling dice was limited but not a big issue. The biggest drawback to running a game on a computer was stress: the players ate and drank at the gaming table, so a “no food or drink near the computer” rule was quickly instituted and strictly enforced.