Downing, Dave

To: Morone, Anthony; 'Roan, Frank'; Cain, Billy
Cc: Hilleman, Richard
Subject: RE: Status of MED

I want to get the relevent parties together once Frank and Paul return to begin the transition period. We need to understand when the mission editor will be ready to be used by the designers and what we need to do game wise to integrate the data.

-=dave=-

From: Cain, Billy
Sent: Tuesday, September 17, 1996 9:56 AM
To: Morone, Anthony; 'Roan, Frank'
Cc: Hilleman, Richard; Downing, Dave
Subject: Status of MED

I'm sending this to you guys first. If you wish to forward, go for it.

EXECUTIVE SUMMARY:
- We need a programmer to start working with Ed and Paul soon.
- We need to ensure our designers have a working knowledge of the WC IV mission system. Perhaps Jeff could put this together as one of his tasks.

Hope this clears things up for you; it did for me.
bjc

PS Perhaps Frank is ready to deal with this, so let's wait until he returns for him to act upon it.

MED - WHAT'S UP?

Ed came in this morning to discuss the state of MED. Here's what he told me: MED will create and manipulate data designers input through it. Someone on the engineering side of WC5 will have to create the system to eat and interpret the data.

WHAT IS MED, ANYWAY?
MED's ***purpose*** is to create and edit Room, Sector, Mission, and Series data in a visual environment.

MED is really just a tool that will get the data in order, it is not a full mission editor. It will utilize a GUI system, however the functionality of the data has not been within ACE's area of responsibility.

On the data creation side, MED supports syntax checking, but not logic checking. That is, it will know the difference between SET_VAR and SET_OBJ, etc. Designers will not have a lot lof 'stupid' programming errors, but there will still be a lot of programming knowledge required to utillize it fully.

WHAT WILL WE HAVE TO DO FOR WING 5?
On the data interpretation side, there is a LOT of work to be done. The designers (along with some programmers) have provided Ed and Paul with a list of commands, such as ATTACK_SHIP, but we haven't defined what those commands are. The functionality will have to be done on the WC5 side. Have I emphasized this enough?

Phil says "Defining AI functionality is on my Design milestone list. As a task for the designers to supply the needed commands, and an assumption that we will have all these commands defined (by April)."

We will also have to add two BIG buttons to the editor: TEST MISSION, and DOWNLOAD TO TARGET (PSX).

Phil says "The ability to debug a mission needs to be thought out in more detail. Watch mission flags, global flags, etc. Be able to tell when different objects become active, etc."

HOW DID BILLY GET SO UNINFORMED?
Perhaps I've been out of the loop, but I was under the impression that we were getting something else. Ed

took the time today to tell me what we were getting. I appreciate that. I'd like to submit we get someone to work with Ed and Paul as soon as possible to learn what they're doing, and get ready to 'take the ball and run with it.'

There is a strong chance that we will **not** have a completely graphical interface, and I would like to ensure that we will have a functional tool that the designers can use easily. If we need to teach them programming, let's get on with it. Please provide feedback, and help get this cleared up as soon as possible.

Phil says "The more the designer knows about programming, the better results they'll have with scripting missions. It's feasible, however, that work could be split up among the 'visual designer' laying out the specifics of the mission, and the 'technical designer' making the script work."

Another roadblock: it doesn't appear that MED will support the 'living briefing' idea. That is, MED won't play back a mission from data, so we can edit the briefings as we go. Ed says it may be possible - but it will take time. Time that would take away from other, more important tasks.

Phil says "This is a task Frank signed up for (mission briefings from mission data) saying he could make it happen. The design now depends on that assumption."

MED functions on four levels: ROOM, SECTOR, MISSION, and SERIES.

I tried to learn how MED is to function, so here's a small version. Ed and Paul can provide more detail as it becomes available. Suffice to say, MED is VERY generic.

ROOM indicates a 2D, ***persistent*** environment, i.e. the actual rooms themselves. Users can but are not limited to defining hotspots, associating uses for them, attaching postage stamps(sprites), etc. We can place temp art, calling for files that can be replaced later. Links from room to other places (another room, space, etc) are specified here. Rooms are connected by hotspots to create "bases."

SECTOR indicates a 3D, ***persistent*** environment, i.e. space itself. It can indicate anything permanent (like a planet or a jumppoint). These can be turned off, but the idea is to eliminate redundant work, and to maintain a permanent universe. Once the sectors have been defined, the missions can be designed. I believe that objects within sectors can be re-defined without affecting the missions. A major point to make is that there is no continuous path from sector to sector. (i.e. you have to jump out of a sector.)

MISSIONs specify object behaviors. A MISSION is EVERYTHING from where you land, what movies play, the postage stamps, the mission itself, and the debriefing. Mission designers will need to know simple programming from previous WCs to utilize the system to the fullest. Again, temp art can be used for hotspots, and having one copy of each mission will (hopefully) eliminate redundant work, and help with version control. MISSIONs are temporary modifications of fixed objects, in a binary format.

SERIES is the binary mission tree system. It hasn't been designed or implemented yet. This is where you turn ON one mission (or set of missions) or turn OFF one mission (or set of missions).

SUMMARY: MED "the GOOD, the BAD and the UGLY"

Good:
- Consistent universe due to rooms (bases) and sectors.
- Minimalizes repetition of data
- Can add and delete missions at will now, with minimal impact on the game.
- Will ensure a framework of a game before we have missions. Bases (attached rooms) and sectors should be defined first.
- Lots of placeholders, so we can streamline where necessary.
- Phil says "I think the designers will find creative ways to amuse themselves with inventive placeholder art. My prediction is this will add fun to a normally mundane task."
- When real art comes in, replacing placeholders should be minimal effort.
- Can re-use system on many different types of games, including a Privateer...

Bad:
- Can't divide up tasks of gameflow and spaceflight mission design, they're one and the same.
- Could make us sick of our gameflow system, by having all placeholder art in there at the time of mission creation.

- Could be difficult to ramp a programmer up so they can take over.

Ugly:
- Won't be as pretty as we'd like for it to be, and it'll take someone that really enjoys this type of work to come along and clean it up for us. And we don't have a lot of time, if we want to tweak these missions so they're fun to play!!
- Communication on this tool was/is not as good as it should be.
- The Object Editor is on the back burner until MED gets to where it should be. (i.e. Paul's helping out for a while.)
- Phil says "Much more thought and work needs to go into how existing missions will be tested and debugged."