

W

C

O

Privateer

W

C

O

Privateer

Design Document Review

DDR

W

C

O

Privateer

2001



COMBAT

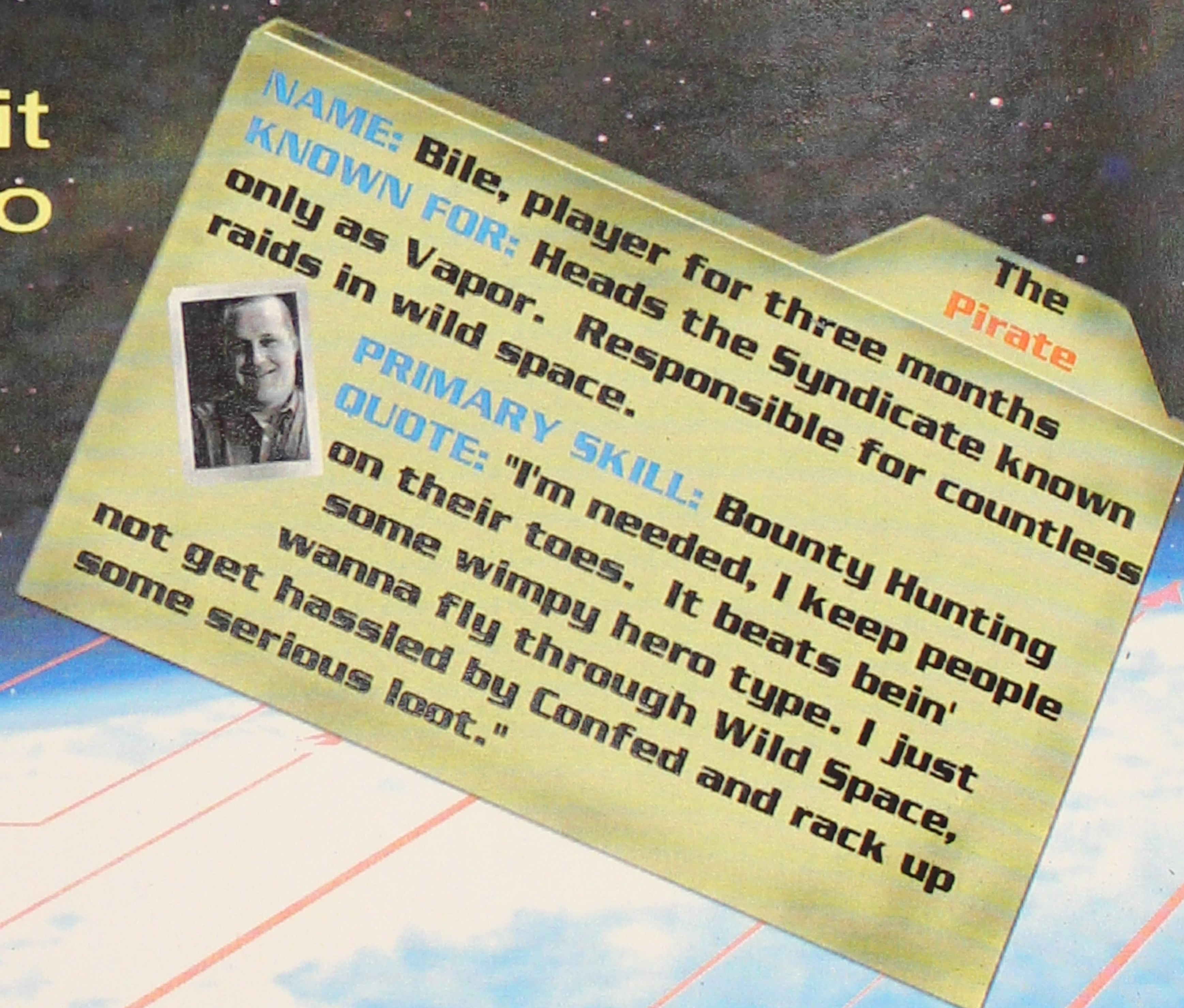
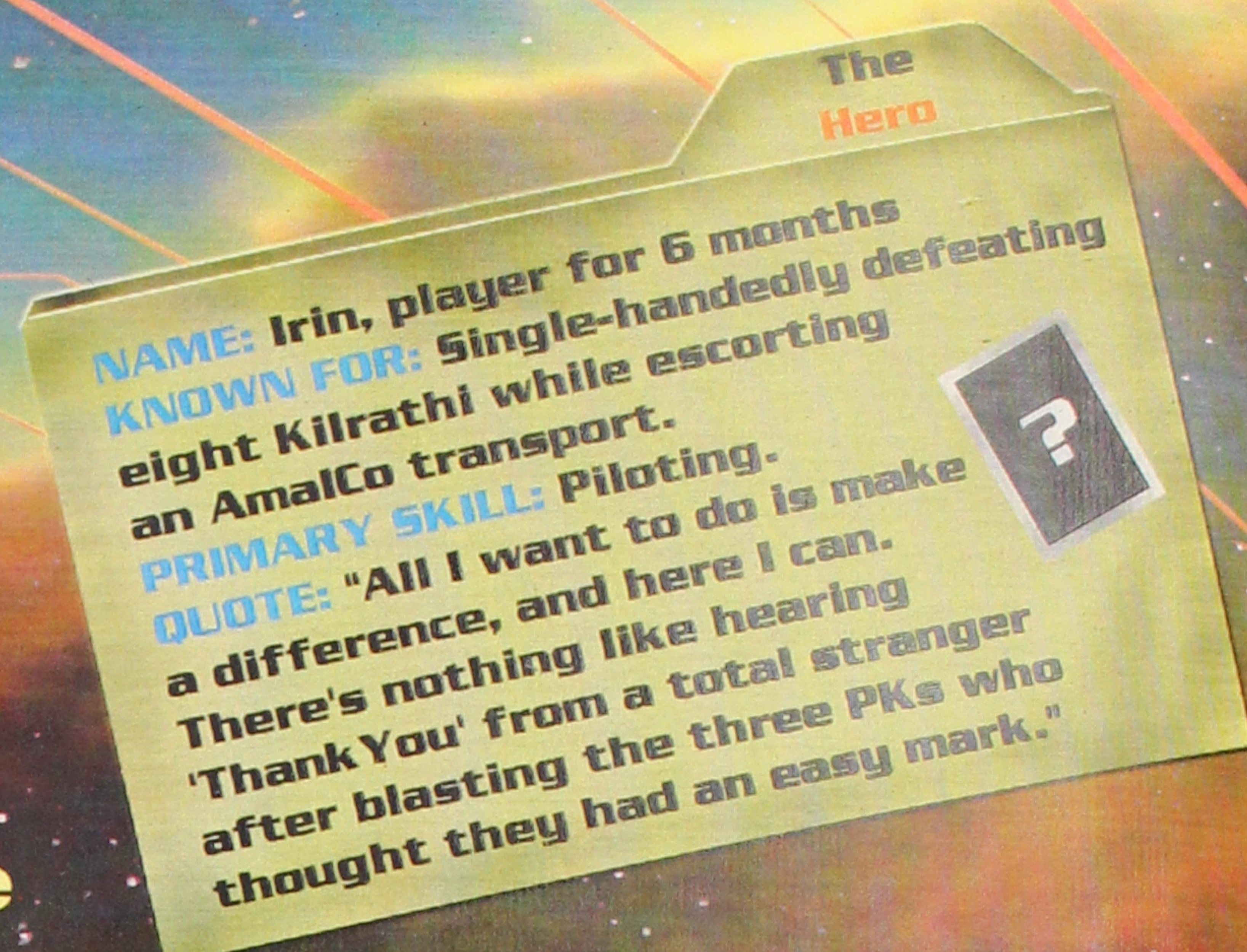
Thrilling space combat in the Wing Commander universe.

Ships you can customize and hotrod with a wide range of weapons and defenses.

Fight pirate syndicates and alien races-or other players in wild space.

Capital ships that multiple players can fly together.

A mission board that makes it easy to instantly find things to do out in space.



Rule the

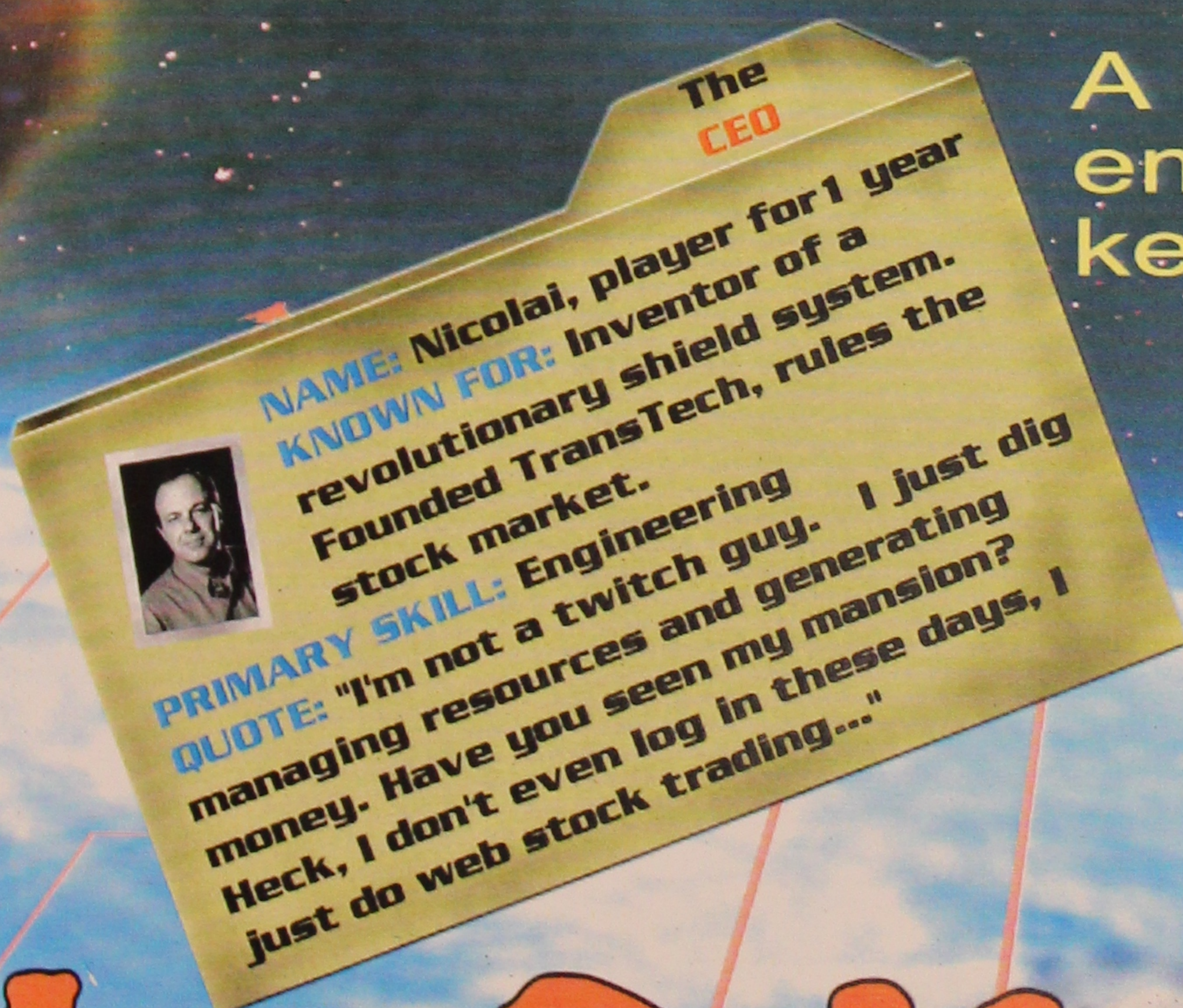
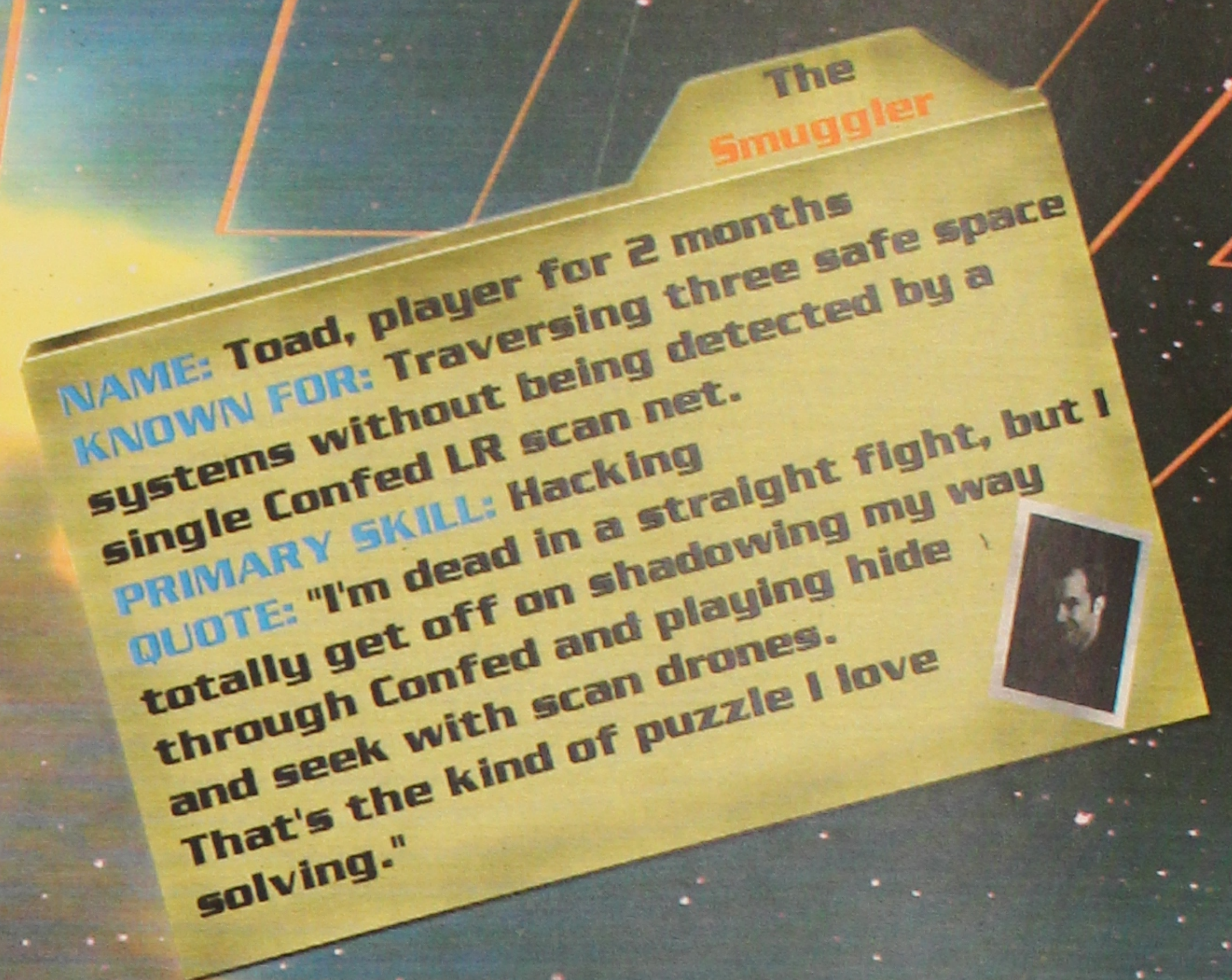
COMMERCE

Form companies and put them on the stock market. Invest in your fellow players!

Check your stock and make trades on the web—at work, at home, on a coffee break!

Mine resources from a limitless supply of planets and trade or smuggle them between space ports.

Manufacture a wide range of goods and sell them on the in-game commodities market.



A robust news system that will email you or page you when key events occur!

in any way

COLONIZATION

The Explorer



NAME: Horshak, player for a year and a half
KNOWN FOR: Opening up 10 new jump points, discovering four systems, 10 species of life form, and encountering a mysterious stellar being never before seen by man.
PRIMARY SKILL: Surveying
QUOTE: "All I wanted was the freedom to... you know... 'Go Boldly where No Man' and all that stuff..."

Millions of planets to explore, each unique. Land on them and walk around!

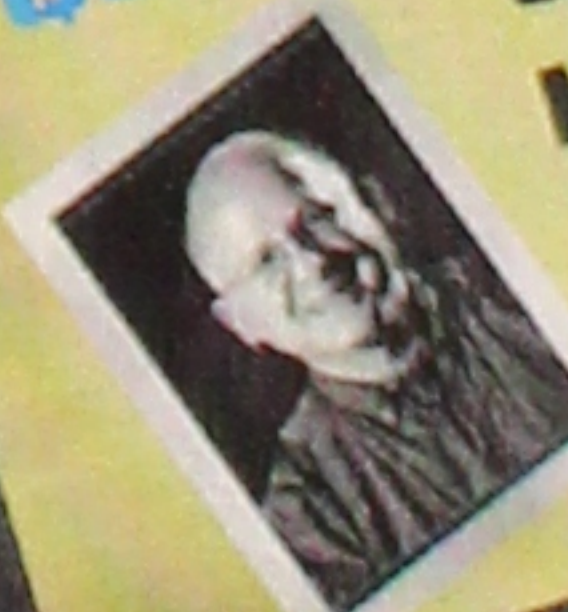
Bizarre alien creatures lurk on the ground, to be fought or harvested for resources.

Colonize a planet: build a home there, vote in a governor, and run the world!

Run a bar, tailor clothing, or write a newspaper: live a virtual life in a science-fiction universe.

Civilize the frontier! Turn wild areas into part of ConFed through cooperation and diplomacy.

The Governor



NAME: Jenn Blaine, player for one year
KNOWN FOR: Converting the former Wild System of Xp31C to Safe in the face of constant raids, then growing it into a center of trading.
PRIMARY SKILL: Architecture
QUOTE: "I was the only one that could draw the colonists, balance the books, setup the Funplex, and make this planet a safe place to live."

you can

We Create Worlds





We create **worlds**®

October 6, 1999

Re: *Wing Commander Online: Privateer*

Dear colleagues,

ORIGIN is the leader in persistent world online games. With *Ultima Online* as our first beach-head, followed by the launch of *Ultima Online 2*, we find ourselves in the position of cementing and extending our leadership. The project outlined in this document is the opportunity to do more: to open new markets, reach entirely new customer bases, and to push the possibilities of the genre further than anything has attempted previously.

Wing Commander Online: Privateer benefits not only from one of ORIGIN's key properties, but also from its innovation. By combining the addictiveness and community-building aspects of the online roleplaying game with the action and excitement of the space sim, it brings together players from differing backgrounds to make an online galaxy that benefits from their synergy. Access to important game mechanics via the World Wide Web makes it a product more accessible to the casual gamer. We are also focusing on ancillary revenue by addressing it early in the design process. This design serves as a pre-emptive strike against potential big-name licenses such as *Star Wars* and *Star Trek*.

The document you hold in your hands serves as both executive summary of the proposed design based upon the previously published Minimum Feature Set, and as a resource for more detailed design information. It begins with a three panel brochure (which you will find in the folder pocket). We include this to give you a better idea of the ways we feel this design can appeal to multiple audiences. Then there is the DDR document proper, divided into sections on overall gameplay, the game mechanics and design, the development plan, and the staffing and budget information. In coordination with Alex, Gordon, and Dominique, we have augmented the budget with an additional workup of the project's projected ROI. After the DDR proper, you will find the "Pile of Thought," two hundred pages worth of design documents that have been developed by the design team during brainstorm sessions over the past few months. At this time, many of these documents are not yet final, pending further refinement, but they should be sufficient to give those reviewing the design a clear picture of our approach to the game design.

The team we have assembled so far is one of the best I have had the privilege of working with. There are no egos here and everyone resolves differences of opinion with the goal of creating a successful product in mind. We hope that this document serves to better clarify our vision for *WCO:P*. We look forward to hearing your feedback on the design, and hope that you are as excited about its potential as we are!

Sincerely,

Alan Pavlish
Executive Producer

and the *WCO:P* Team

5918 West Courtyard Drive • Austin, Texas 78730-5036 • phone 512.434.4263 • fax 512.794.8959

AN ELECTRONIC ARTS® COMPANY

Table of Contents

1

OVERVIEW

2

GAME DESIGN

3

**DEVELOPMENT
PLAN**

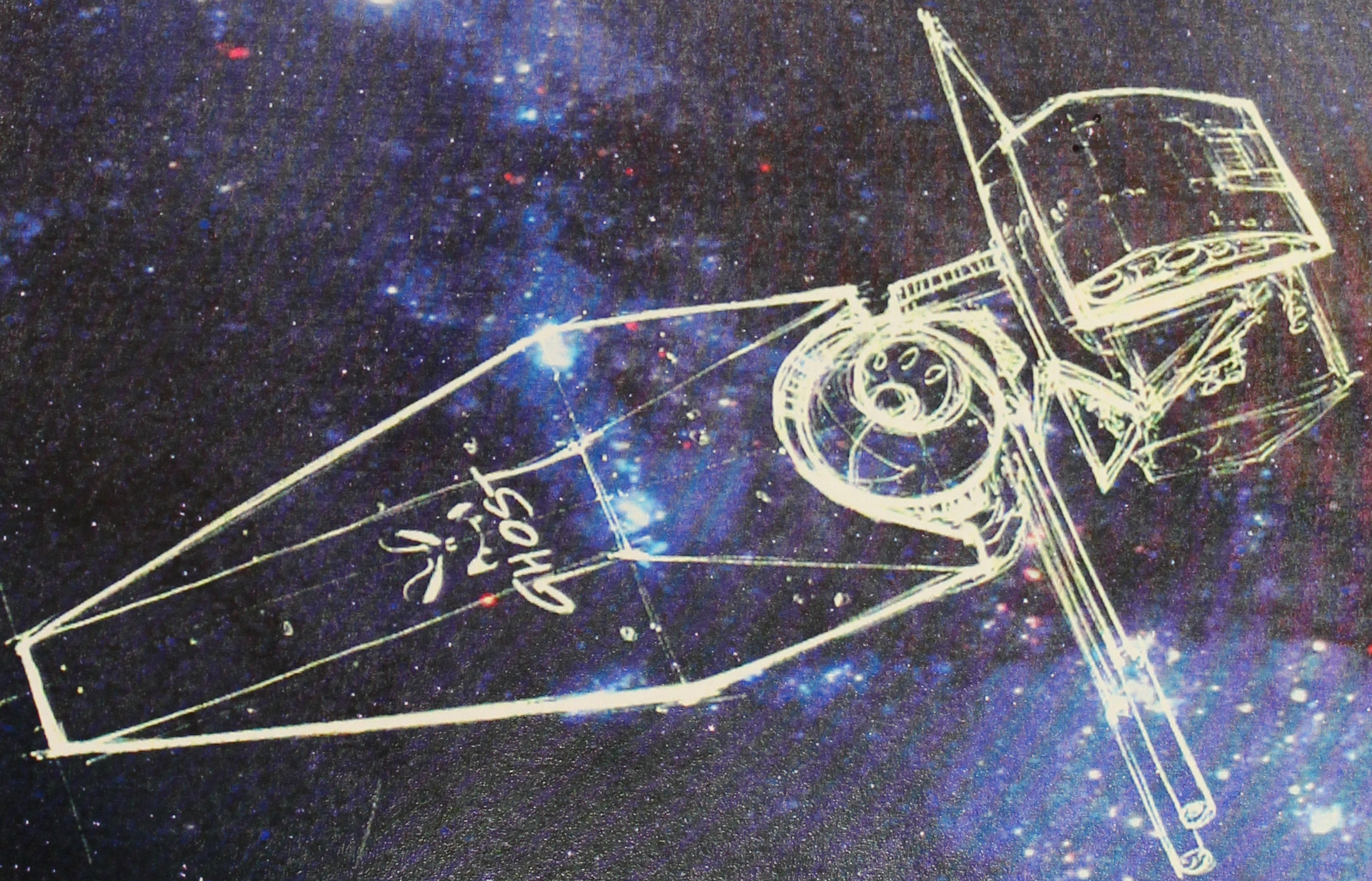
4

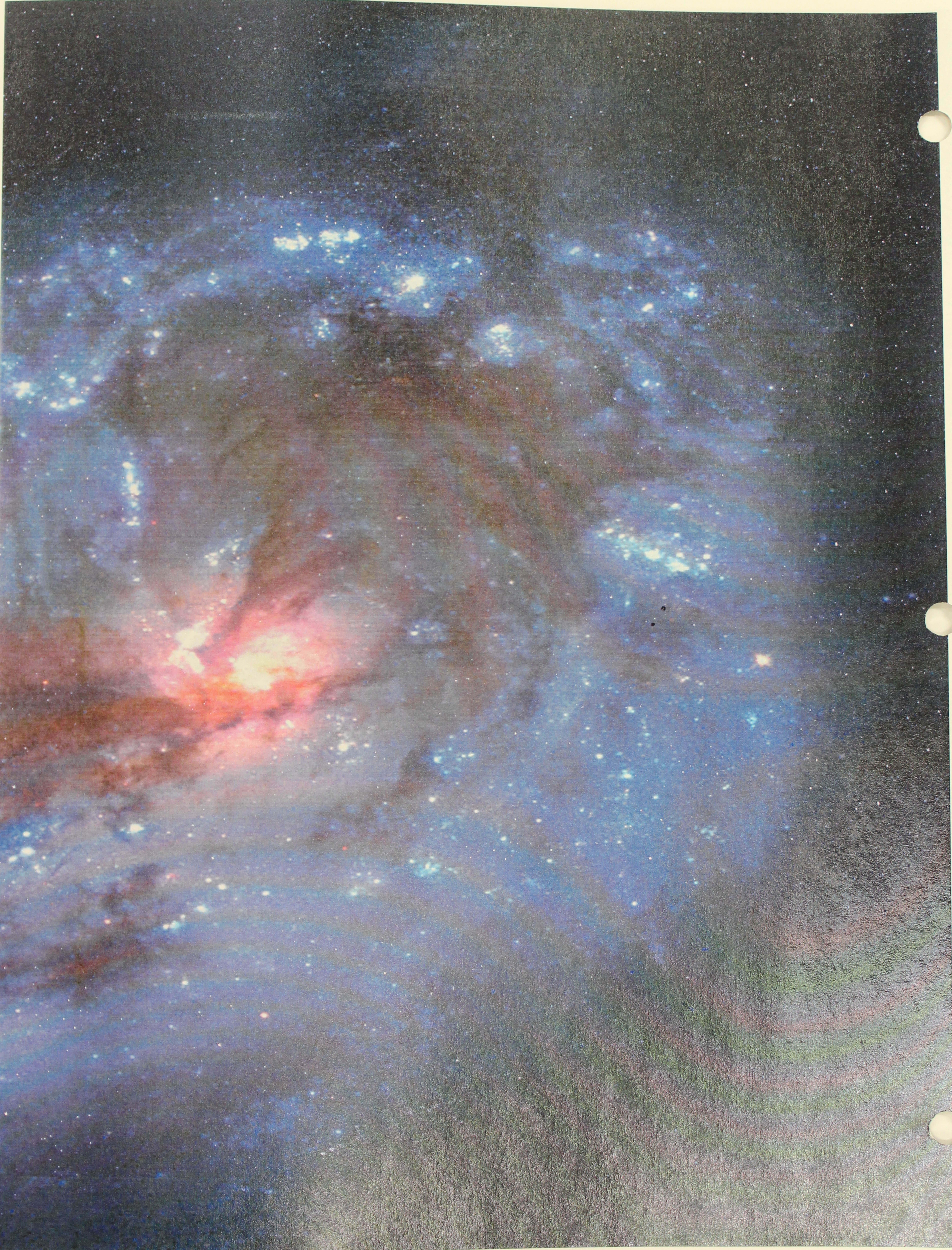
**STAFFING/
BUDGET**

5

**PILE OF
THOUGHTS**

PLAY SESSION





The Experience

Hi, my name's Jimmy, but here most folks know me as Irin. I haven't been playing for very long, but long enough to know I'm going to be here for a while before I run out of stuff to do. Let me give you a run down of everything I've done, from install to some game play...

I found out about the game from a review in a magazine. So I thought I'd try it out. I mean, the thought of owning my own ship and shaping my own destiny against the backdrop of celestial intrigue, war, commerce, and exploration...pretty attractive. When I found out I might be able to discover my own planet, name it, AND colonize it I had to see if this was for real.

When I installed the game I freaked. The install was the coolest! It showed me a complete timeline of the universe against a shifting galactic map and cool sound bytes from key points in history. By the time the install was finished I knew where I stood in the universe. Better than some games that force you to read or sit through interminable cut scenes to get to the game play.

Then the game began. I was expecting some canned character creation process that is totally separate from the game...you know, more like configuring my windows desktop than creating an alternate persona. Well WCO surprised me. During the install and initial screens I learned that I was applying to be a part of the rebuilding effort being run by Confed and the corporations. They called character creation initial CCSP (Citizen/Corporate Subsidy Program) Registration. I chose my look using a super cool interface that let me make my guy fat or skinny and grow the hair...everything!

So, I made my character then started into an interactive tutorial system that was framed as my training to be a new pioneer. It first made recommendations as to what initial certifications tutorials I should pursue based upon a personality profiler. WOW! For the first time I didn't feel like I was being forced into a role I would regret. I could play the character I wanted without being clueless when the game began as to what to do!

Another bonus was that I didn't have to sit through *all* the tutorials/simulations available. Whew! I wanted to be a merc, so that implied combat and piloting training, it then tailored the tutorial regimen to reflect my tastes! And boy those were impressive tastes of the game I was about to explore.

In the first part of the tutorial I got to experience an amazing space combat sequence while running a turret on a capital ship. Next I got to pilot a small ship through a vast asteroid field. All within the first thirty minutes! Man, I can't wait until that is my ship tearing through *real* Kilrathi and not some simulator. I also learned the basics of contacting people, walking around, and basic game functions. Luckily the game's interface was so clear that this was a breeze to get accustomed to.

Okay, so I'm pretty psyched now right? I've only been in the game for a few minutes and I already know (and have seen!) what I have to do to achieve my dreams of being a neutral mercenary, like Han Solo. It gave me three missions I had to fulfill then directed me to the mission board to review/accept them at my leisure. So, I exited training onto my starting planet. My ship is located nearby, with handy directions on my PDA. It's not the nicest ship, but I'll get better. I picked an all-purpose light ship with a good cargo hold and some light guns.

A greeter met me at this point. He was a really nice guy who went by the name of Vapor. He asked me how I was doing, if I had any questions, or needed any help in general. I asked him if he was a GM, he said no. Turns out there are players that tested the game in Beta who now help newbies get the hang of the game beyond the tutorial. Really friendly idea! It's surprising how many real world folks actually give up their time to do this online. Personally I'd rather be alone, or invisible, until I have the hang of things), but I'm glad for the help.

Vapor guided me to the mission board (which is where I was heading anyway). From there I was able to accept the newbie missions that helped me get some early achievements with my piloting skill. These were pretty standard, but man space looks beautiful in this game. Real planets, real systems, nebulae, this game feels huge! I'm really glad the game focused me on specific tasks to get me rolling down the advancement path.

Whenever I got an itch to do some killing against folks I just hopped into the simulator to compete within the virtual arenas. This was great for me because I hadn't played the game long enough to have the strongest ship on the block. Turns out that there is a leader board and ranking system that takes players up through tournament ladders! The higher you are on the ladder the more customizable your simulator ship becomes! I met a couple of guys who only play the simulator tournaments and trade stock on the market offline. Action and money, I guess that's all some people need.

Well, a few weeks have passed since then. I've got a pretty good ship now and can take out my fair share of pirates and Kilrathi raiders. I'm still not brave enough to venture into wild space to fight the player killer clans, but I now have friends to help me with that. These are all people I met in the game, so I've never even met them face to face. I met them all in different ways, though, trying out different things you can do in WCO. The coolest thing is that all my friends play the game in different ways than I do. Let me run down the list and make some introductions.

My friend Philo is the head of a small company (guild) called InterTech that made a pretty good splash on the commodity exchange after they constructed and sold the first shield unit with a 20% bonus against kinetic damage. Now all Philo does is hang out in his corporate mansion on the company's planet. He's traded his pilot certifications for bartending and now runs one of the more popular public bars in the quadrant (it's the only player run one for 10 hops with a simulator pod). I met him because I was hanging out in his bar, and we got to talking.

Scooby, on the other hand, is dangerous. He's been acting as muscle for Philo's delivery transports (which is how I met him). On one escort run alone he took out 5 real players and 10 Kilrathi raiders. His ship is a dangerous hotrod with a custom flame-job provided by one of Philo's buddies. It's got two plasma streamers and an MK5 Vulcan cannon. Coupled with his tech nine shields and dual missile launchers, you have someone you don't want to meet in a dark asteroid field.

Horshack is our resident explorer. He actually discovered the system where Philo's company's planet is located (he even got to name the system!). He hangs out at the bar a lot too. Horshack used to be a hero type, a bounty hunter who sought out and destroyed over 100 player killers before hanging up his badge. He made enough money from that, and got enough backing from Philo, to begin the search for new systems. His current ship is modified heavy transport that is more of a floating gunship these days. He can take a compliment of 8 other people with him and they can run the turrets on board or assist with tactical scans or navigation. He can set the turrets to auto-fire as well...but nothing beats a human eye. His old ship is in storage, a medium fighter

Wing Commander Online: Privateer

DDR

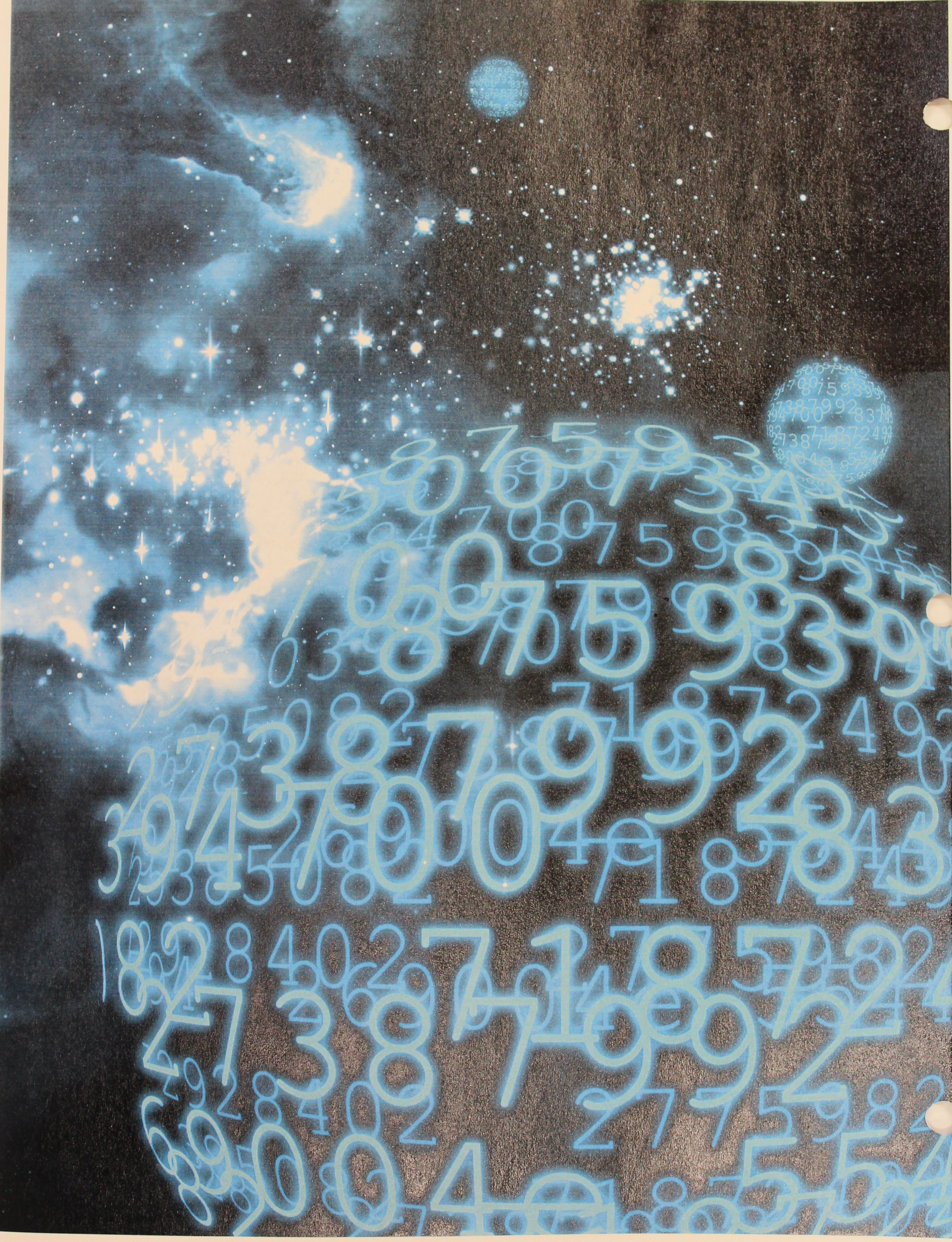
named "Lulu". He actually got the "Lulu" texture added to his ship after winning one of the simulator death match tournament championships. Good guy, pretty inspiring. I'm running jobs for Philo now, and making pretty good progress. He doesn't know I've "lost" a couple of shipments to a rival...but I'm a merc...and the price is right...



KEY

FEATURES





Key Features

Combat

- Space combat in highly customizable modular ships with a wide array of weapon types and defenses.
- Ground-based combat that is intuitive and easy to use, for the purpose of adventuring in a ground setting.
- Capital ships that allow multiple players to operate as a specialized Crew.
- Accessible teamwork interfaces support a wide range of strategies between players.
- Detailed relationships between different types of weapons, armor, and shields promote unique tactics and exciting combat.

Commerce

- A player-driven economy based around player-run vendors and commodities markets local to specific planets.
- A stock market where players can invest money in each others' guilds, usable both within the game and via the website.
- Players craft every item in the game: ship add-ons, engines, etc; clothing and ground-based equipment; miscellaneous decorative items; etc.
- Support for "community-building" professions such as storytelling, news gathering, musicians, barkeeping, etc.
- A mission board that makes it easy for players to quickly find something to do, and that allows players to post missions for one another to complete. Allows people to create their own business, such as a Taxi service.

Colonization & Community

- A colonization system that allows players to build homesteads on planets.
- The ability to make a planet into a safe zone.
- Govern your own planet, set laws, and build facilities that only true colonies can have.
- A robust set of tools for guilds, allowing players to form companies of various types, communicate easily with members, measure standing against other companies, and pool resources.
- A news channel system that supplies constant, customizable information about the game, both within the game itself, and also externally via the website and possibly paging, email, and push-to-desktop mechanisms.

Adventure & Exploration

- An easy-to-understand advancement model that allows players to gain status in professions, change professions on the fly, and easily compare their status to that of other players.

- An ever-expanding galaxy that is shaped by player actions during the Live service.
- Leader boards for professions and activities within the game, ranging from adventuring and combat to economic and socially-oriented activities. These leader boards are visible on the website.
- Player vs player combat controlled by physically breaking up the map into safe zones and dangerous zones.
- A planet generation system that is capable of creating a wide range of wilderness planets for both colonization and adventure.
- Generated alien creatures specific to each alien planet.
- Randomly created missions provide adventure on both the ground and in space.
- An engaging, long-term plot making use of the storytelling tools and techniques developed for *UO2*.

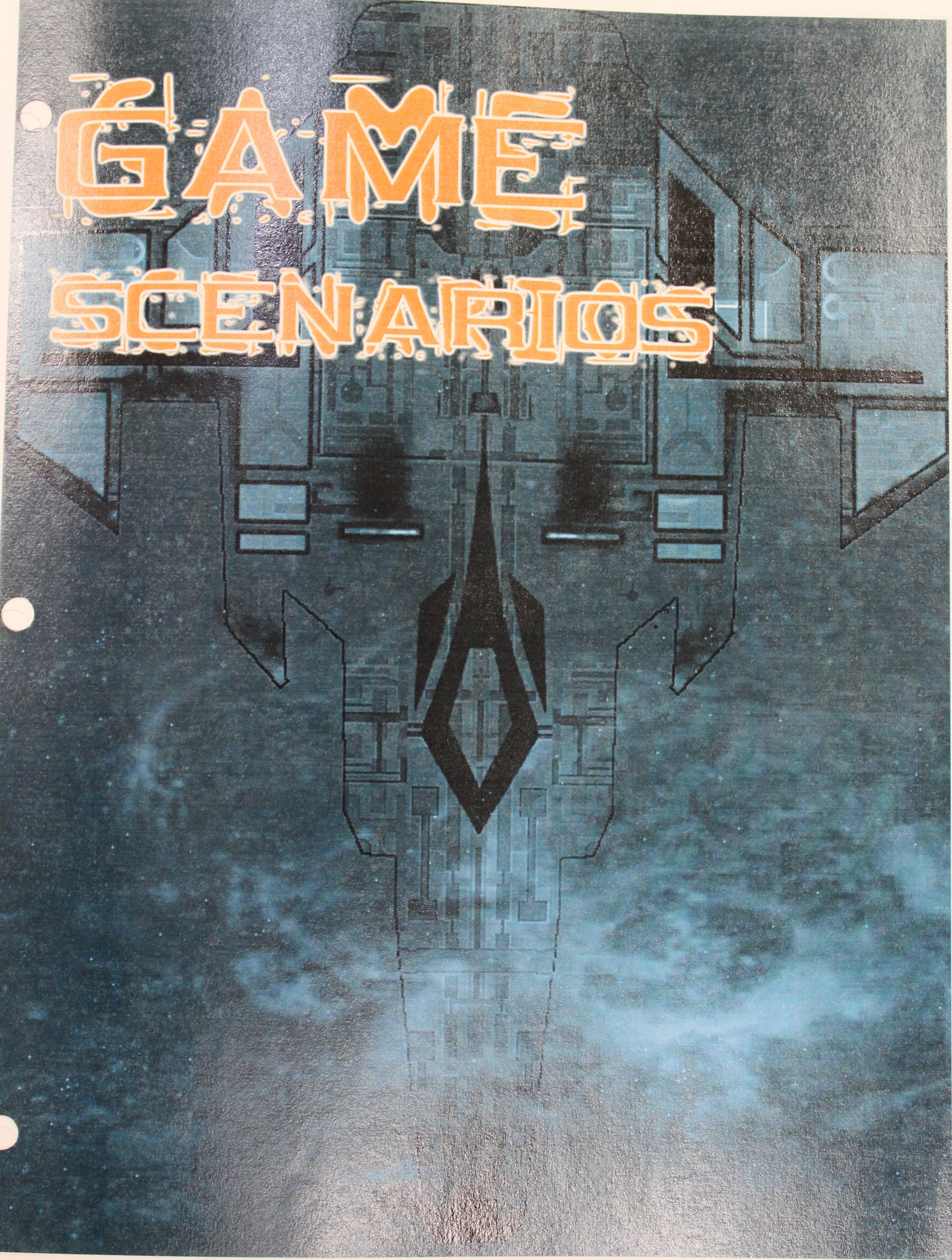
Play on the Web

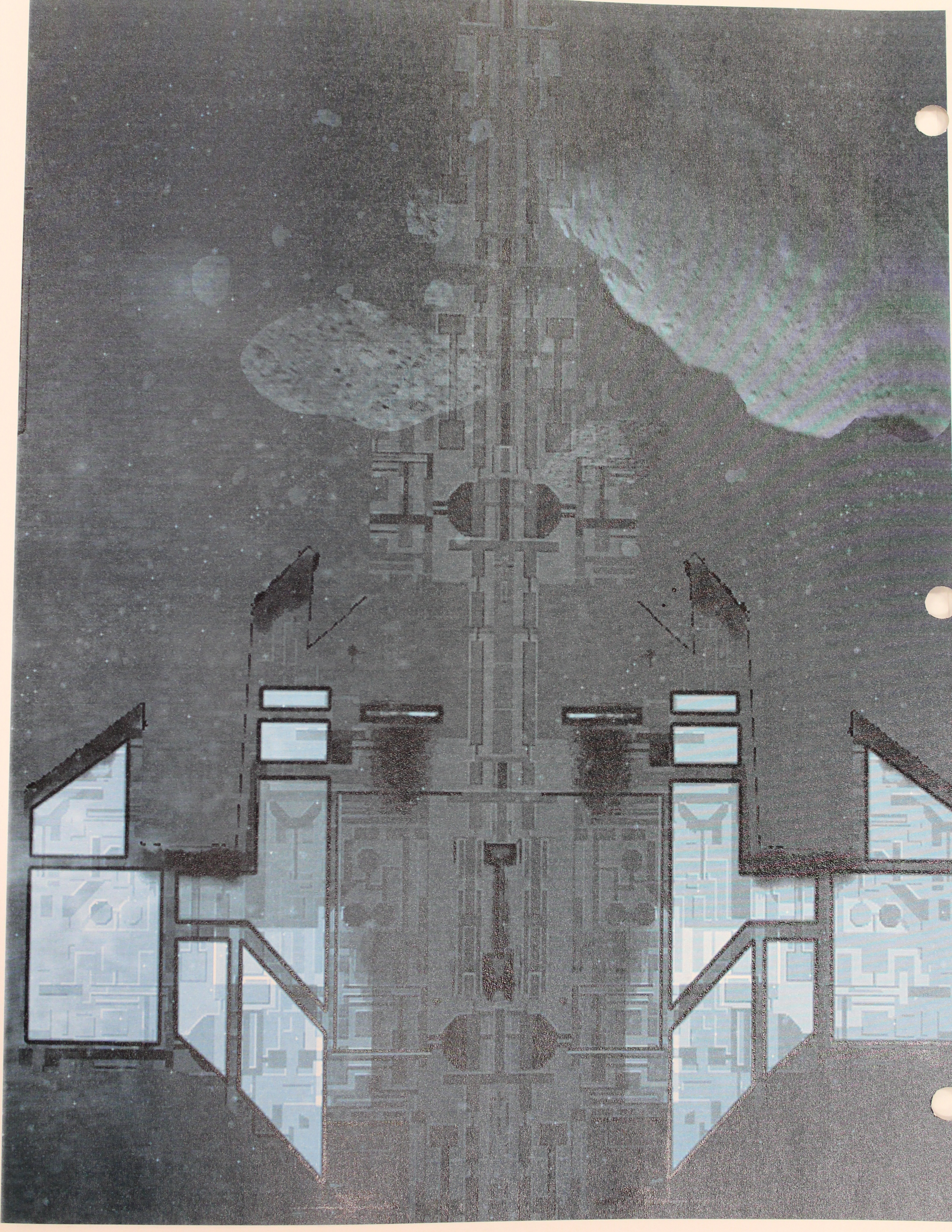
- Offline play for specific subsystems such as the stock market, resource mining (which occurs automatically regardless of the presence of the player), etc.

High quality of presentation and service

- Newbie training mechanisms and activities to enable a smooth learning curve.
- A focus on a high quality, immersive experience supplied by superior technology and art. A 3rd person overhead perspective on the ground and a 1st person perspective in space.
- Player avatars that are highly customizable, including faces, clothing, and body shape.
- An install that is fully within the fictional context.
- A robust chat system including player capability to create channels for their own use, private messaging, buddy lists, etc.
- A full array of GM support tools.
- A full array of player tools for policing their environment, including ignore facilities, harassment logging, and text filtering.
- A server that is localizable to multiple languages.

GAME SCENARIOS





Typical gameplay scenarios

From the beginning as a first time user getting started

I create my character, choosing an appearance and proceeding through the immigration process to head out into the frontier. I am told the corporations are looking for colonists, traders, and other rough-and-tumble types who want to tame the unknown. I do not pick skills or professions yet—I just jump into the game and start in my ship. I'm thrust into a jumpgate that is spawned just for me. There are about 5 other computer AI ships, all with very little health and weapons that do very little damage. All I have to do is fly and shoot. I cannot possibly die because it is so easy. I do not need to mess around with my ship. Just fly and shoot and kill. I get plenty of rewards for killing. Money, goods, and most importantly, a land grant for a new homestead on the frontier... that will help ease me into the game when I land.

Once everyone is killed I am prompted by my computer on how to dock and land on the newbie planet. Better yet, I could step out of a simulator and realize that the whole thing was just a simulation.

The stock market

I've been investing in some companies that are building a large city out in the Gamma sector. When I get into the office in the morning I check my WCO:P investments. Maybe make a few trades. Check to see how my friends are doing in their investments. Also, check WCO:P commodities markets to see if anything of interest is for sale. I also check the items that I've placed bids on, as well as items I have for sale. Check my email, top news stories and any missions that might interest me. Sometimes I check the forum if I have some questions out there that need answering.

When I get home and log in, I check everything I checked in the morning. Usually not a lot has happened during the day, so it doesn't take much time.

When I get into the game I first check to see if any of my friends are playing. If so, I hook up with them and see if they want to mission together. If not, I find a mission of interest. I like missions where the combat is not too long. I'm saving up to buy a house in the game so I look for missions that are of the highest reward for least time to put in as I don't have hours each night to spend time playing. I make most of my money on my investments. I also try to find missions that will help out the companies that I have invested in.

Trumpsuit the pirate - a playerkiller scenario

Trumpsuit was busy getting repairs done to his fighter in a pirate friendly city. This was important, because Trumpsuit had personally killed over 350

hapless merchants and pilots through the course of his lucrative career. He was not only an ace space pilot of the highest level, but also a better than average marksman with his ion hand cannon. While choosing which of his armor types to repair, and keeping an eye out for bounty hunters, he received an instant message on his COMM from his buddies in the Black Hand. They told him about a high paying mission to the Starlap sector to retrieve some Alien Shield Technology. The reward was both high credits and personal use of this tech...an ideal ambush location for brave pilots electing to go after the reward. He establishes a comm link in the Black Hand guild channel and informs them that he is on the way. The Black Hand goes on to ruthlessly murder 18 pilots seeking the alien technology before securing it for themselves.

Hammer the capital ship

"Our cruiser better show up fast or we're dead meat!", screamed Snake, the second element lead pilot. Plasma bolts burst to all sides of his cockpit, shaking his ship nearly to pieces.

Hammer burst from the jump gate like some sort of battering ram crashing through the front door. The red and black honor insignias along her cyclindrical length turned purple in the intense blue light of the gate's glow. Her crew was making good use of the dying glare from the jump event to begin targeting enemies whose eyes were only beginning to recover. Streams of laser light leaped out from the corona, producing several orange blooms of fire as they found targets.

Hammer's Captain, Nick Newhard, quickly assessed the situation: on the scanners the amount of red dots (enemies fighters) about equaled the number of blue friendlies, but it was the larger red icon on tactical display that worried him. The Tac officer was on the ball and a small block of data began forming near the red marker. The enemy cruiser's shields were mostly intact, hull armor didn't appear damaged at all. More data regarding reactor levels and weapon power would soon follow, but Captain Newhard already knew enough for now. He begin sending orders to his departments through the command net and then repeated them over the comm.

"Engineering, divert power from the jump systems to the forward batteries, minimal mid-range fighter defense, strong close-in defense. We're staying until this is finished. Tac officer, authorization granted for full-auto point defense, don't bother with long range enemy fighters, the squadrons will have to hold their own. Prepare for front battery engagement on that cruiser." He looked back to the tactical screen and centered it upon the enemy cruiser that was now only beginning to turn to. He'd met this captain before.

This time, Branston, you aren't going to get away."

Harry the miner

Harry mines everything he can from carbon to gold. He personally owns over 50 mining installations across the universe. This is very expensive to operate. Consequently, they are all not running all the time. He speculates which systems will need which minerals as he watches the patters of pilots buying weapons and armor that require his minerals. About once an hour from work, he checks commodities markets in various systems against the supplies he has in his. Most of the time he is always offering some of his raw material for sale, but stockpiles a certain amount of each waiting for good deals. Today he's in luck. The armor manufacturer is requesting 100 tons of gold for doping fullerene armor. He is located only one system away. He can't play the game

right away, so he puts up a sell offer for 100 tons of gold at an inflated price, but still 10% lower than what the buyer is requesting seeing if some prospector will transport the goods for him. He'll check again in an hour when he can play the game to see if anything has changed. If nothing has changed, he will run the cargo himself to the armor dealer for a handsome profit.

Space combat

"Spectre" banked her ship around desperately searching for her opponent. She had been conservative in using her thrusters just in case "Mustang" launched a missile and she had to evade. But the warning system wailed only about a radar lock. She looked down once more to check- still no inbound missiles. Had he run out? Mustang was out here somewhere... Lasers sliced down from above, but Spectre quickly juked sideways. She played a hunch and held herself back to a steady turn conserving her energy. Mustang's ship crept just barely into view- he was sliding backwards because he had used up his energy in a quick pivot maneuver. She could see the star-pointed flares along the length of his ship as his flight computers tried to reorient his ship forward. Now was the time to strike. Spectre hauled back on the stick bringing her target within her sights. In her peripheral vision, battery gauges fell, systems went dim with energy starvation but she barely noticed - the crosshairs were almost aligned. Stark shadows flashed in her cockpit as her plasma bolts streaked out and met the target. Shield panels sparkled around Mustang's ship then collapsed inward. A plasma ball punched through the hull, which then split apart in a large cloud of fire and debris. "Your path is clear, Venture One. Proceed into orbit." "Thank you, Spectre. You've just earned a 30% bonus."

Teresa the funplex owner

Teresa had a problem. Her little bar on a frontier world just wasn't cutting it, not against the powerful chain of entertainment complexes operated by the Burning Heart guild. She'd started running it just for fun, but now it was time to get serious.

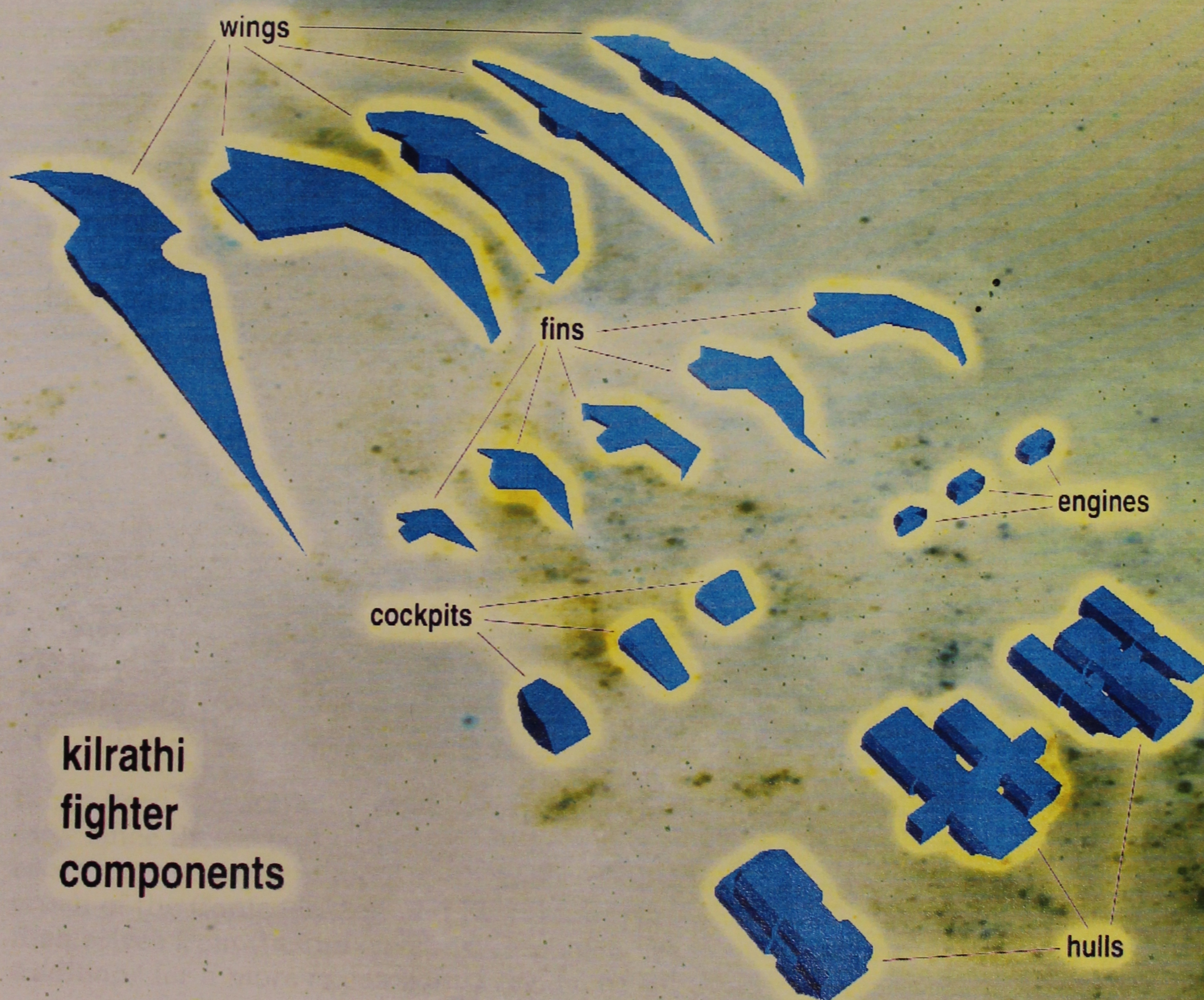
She contacted the best-known news writer in the quadrant to come by her little pub, and he liked it so much he decided to make it his base of operations. Attendance immediately began to rise. As she took in more money at the door, she was able to purchase better facilities. But then the attention turned nasty—her little frontier world wasn't protected by Confed, so as it grew in popularity, pirates and other meanies starting hanging out there and killing the patrons.

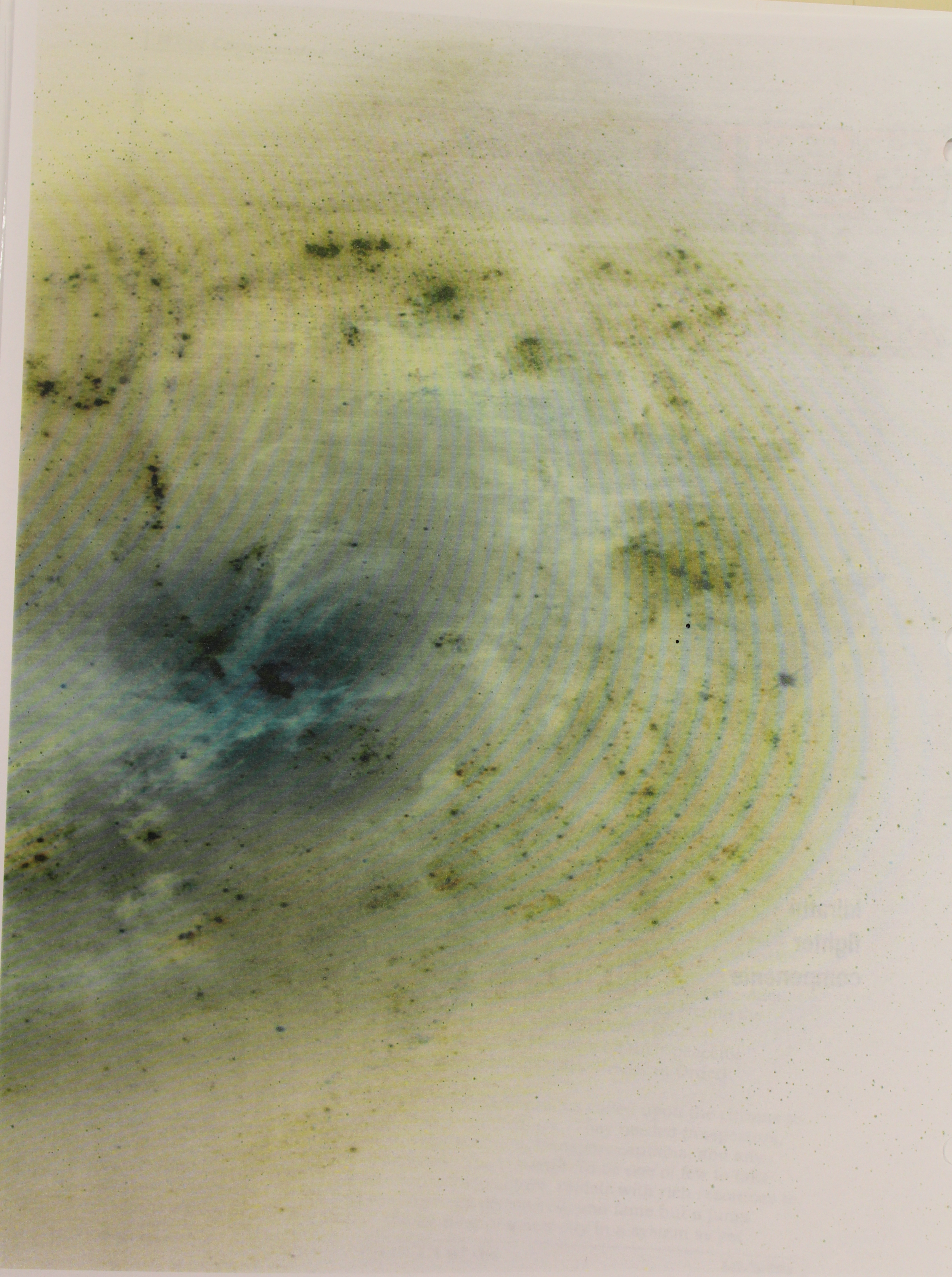
Her solution was somewhat unorthodox. She invited them to make the planet their base of operations. As long as they obeyed the truce at her bar, she'd invest some money in cloning facilities, run a gray market, etc.

Soon there were enough citizens on her little frontier planet that it incorporated as a formal colony. She was, naturally, elected governor. The pirates have grown more civilized, and a few of them have even gone over to the legal side of the law. But most sweet of all—every time the Burning Heart guild tries to set up a Funplex on her planet, she uses her gubernatorial powers to deny them building permits. Nobody grudges her this little abuse of power and now not only is her bar well on the way to being the top watering hole in the quadrant, but her planet is becoming a true crossroads.

SETTING

AND BACKSTORY





Setting and Backstory

"Time passes without regard for the whimsy of your spirit. " --Sivar

In a galaxy torn apart by war, rebuilding requires an understanding of what has come before, what is now, and where you fit within its fragile structure. Your place in the universe resides with the Confederation.

From your home world of Earth, the Confederation spread throughout the galaxy after the discovery of Jump travel; a faster than light pipeline that lead to hundreds of systems full of opportunity and secreted dangers.

Many of these systems were empty. Some were not. Confed made contact with the benign Firekkans, a peaceful race of avian beings, and the Double Helix, an enigmatic race that preferred to be left alone, and had the means to shroud their home systems from discovery.

The Confederation found its first interstellar enemy in the form of the Kilrathi. A ruthless race of expansionists ruled by a draconian government in service to a God of war, the Kilrathi did not waste time with diplomatic endeavors. A war started in 2634 that threatened to eradicate the Human race. Fortunately a hero rose from the ranks of the Confederation military to end the conflict with a decisive strike against the Kilrathi home world of Kilrah.

What followed was a time of intermittent peace interrupted by internal power struggles, and invasion by a new race hinted at by Kilrathi legend and known only as Nephilim. At the third and final encounter with the Nephilim the Confederation lost 40% of their standing fleet in an as of yet unexplained conflagration. The Nephilim were seemingly banished from known space, and half a century of peace shrouded the motives of Kilrathi bent on igniting a flame extinguished with the destruction of their home world.

Neither the Kilrathi nor Confed could claim victory at the end of the struggle that ensued. Both sides beat each other to submission; retreating back through the razed remains of a front line a sector wide. This section is now referred to as Dead Space, and only the hardest merc and pirate raiders thinly populate its charred systems.

It has been a hundred and fifty years since the first Human/Kilrathi war. It's only been forty since the second. The face of known space has been irrevocably changed. Alien races once benevolent have been rendered isolationist xenophobes after feeling the crush of the Confederation and Kilrathi. In particular, the Firekkans isolated themselves from the universe, developing in secret and casting off their peaceful traditions for a more expansionist regime called the NFO (New Firekkan Order).

As the rebuilding began, Confed and its internal corporations called upon the citizens to pioneer the frontier sectors deep in Confederation territory. They needed prospectors, community builders, terraformers, mercenary couriers, transport captains, and any other skilled citizen with the spirit to succeed. The reward? To be one of few to take part in the building of a new Confederation infrastructure, replete with rich resources to exploit, new trade routes, treasures waiting to be discovered, and fame but a jump away. To be a pilot on the frontier, or the founder of a new city in a system as yet

unknown, these are lures drawing the best of Confed's citizenry, and the worst of opportunists.

The risks? Scavs, Pirates, Kilrathi and Firekkan Raiders, unknown life forms planet side or space side, Mercs of all flavor, harsh planetary conditions, and countless celestial unknowns make the opportunity a challenge worthy of you the Privateer.

Special considerations in storytelling

The intent is for the Live team to make use of the *Synthetic Cinema* plan developed by UO2 for ongoing story development. We also plan to enlist the assistance of a name writer (or several) to help shaping the back story and the future fictional developments.

The storyline is also intended to provide numerous hooks for future fictional development, in order to non-intrusively frame the addition of new game mechanics such as additional player races.

We will of course provide the usual array of tools for the Live service and Game Masters to develop storylines and plot. See the section on support and tools for details.



ple

4

18

18

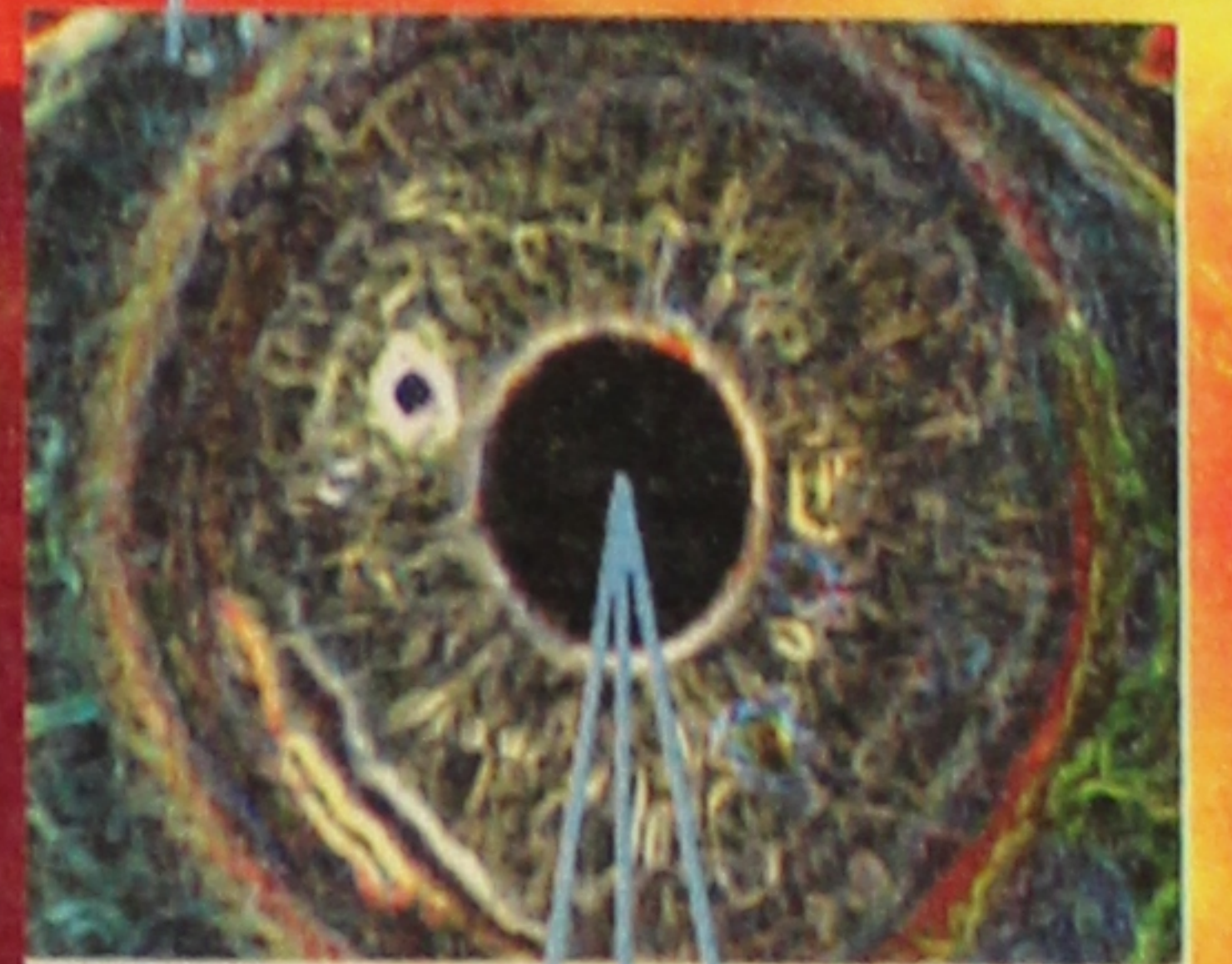
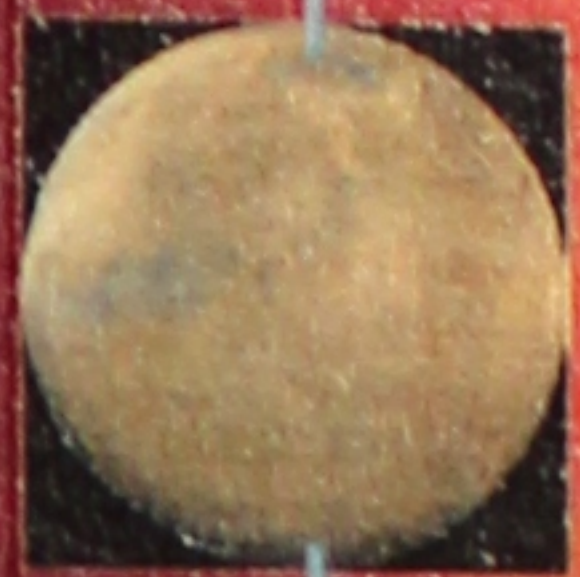
18

18

18

18

18



Gameplay

Gameplay is provided through the use of a variety of tools and techniques. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics.

Design: A simple combat system will make use of the same basic materials, damage, and status effects. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics.

Gameplay is provided through the use of a variety of tools and techniques. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics. The game is designed to be a tactical simulation of a war zone, with a focus on strategy and tactics.

how big is the universe? you tell me! it's a million times bigger than you think it is!



010



Ground control

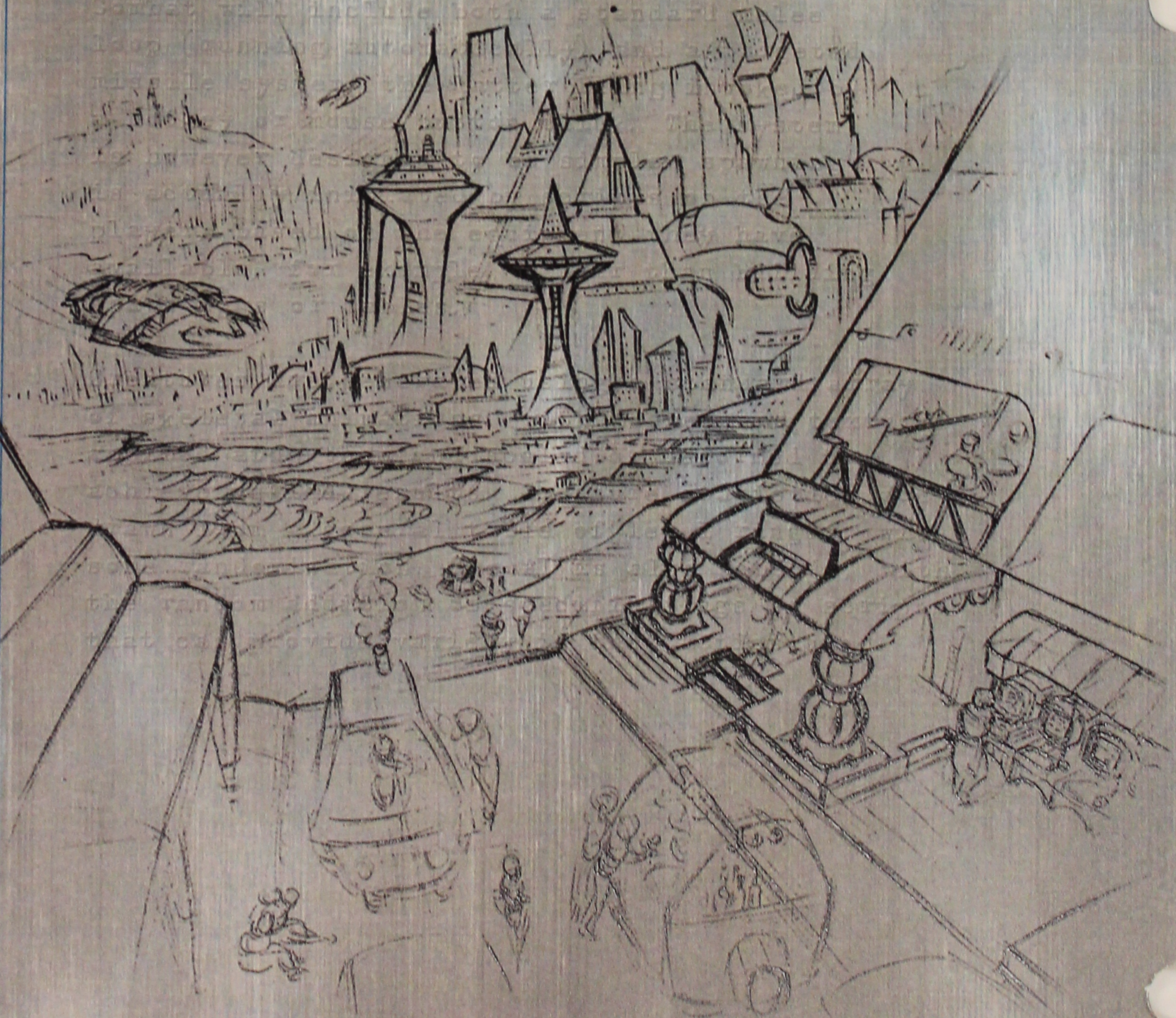
years. To provide more for these players who are not experts, and provide a setting for competition, also to increase the range of missions that are available in the game.

Design: A simple combat system will make use of the same basic materials. Damage types, and damage scaling factors that are used for space combat (eg, the same array of laser types, missile types, and armor types will be used for both systems). Combat will include both a standard melee loop (running automatically) and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple, avoiding as actually initiated by scripts on the player, based on the equipment they have available. For example, rather than waiting around for a script to be triggered.

Creatures are generated using a standard array of skeletal animations, with motion and texture applied after they are loaded or referred to. Creature statistics are drawn from a general table of level ranges, with some randomization. The AI is also generic, with the addition of specific flags or scripts that can provide variety in the tactics used.

Goals: To provide more for these players who are not experts, and provide a setting for competition, also to increase the range of missions that are available in the game.

Design: A simple combat system will make use of the same basic materials. Damage types, and damage scaling factors that are used for space combat (eg, the same array of laser types, missile types, and armor types will be used for both systems). Combat will include both a standard melee loop (running automatically) and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple, avoiding as actually initiated by scripts on the player, based on the equipment they have available. For example, rather than waiting around for a script to be triggered.



Game mechanics

Space combat

Goal: Space combat must be an exciting experience that is easy to get into so that it can serve as a "get in the door" feature for those not familiar with persistent world games.

Design: The space portion of the game takes place in algorithmically generated volumes of three-dimensional space. Each of these sections will be active and interesting, containing a wide range of stellar objects and server-controlled entities for players to interact with. Our intent is to give each section a "personality" that challenges players in many different ways. The graphics of the space environment will be vibrant, dynamic, and colorful. The ships and space-based facilities will have very different appearances, reflecting a wide range of different design priorities and maintenance conditions.

Space flight for will be presented in the traditional first-person viewpoint. The level of immersion will be enhanced with a functional virtual cockpit, as well as a series of visual and audio cues designed to minimize the need to enter keyboard commands. Large ships will be controlled through a simple iconic interface capable of supporting several players performing different roles. Several automation features also allow a single player to easily control a large ship if the need arises. The highest level of accessibility will be our goal, and a joystick will not be required at any time to enjoy space flight.

Ship-ship combat will differ slightly from the established Wing Commander tradition via the inclusion of a *maneuver energy* play mechanic. This provides lower bandwidth internet-friendly play, while adding more strategy to the dogfighting experience. The offensive and defensive ship components are designed with Rock/Scissors/Paper elements. This allows players to balance the strengths and weaknesses of their ships to suit specific strategies while encouraging cooperation. We also foresee the large Capital Ships often being the focus of intense combat in a manner consistent with the *Star Wars* paradigm.

Planetary colonization

Goals: To provide a sense of ownership in the game. To formalize the process of building a player town in the wilderness and provide players a degree of governmental control over the cities they build. Hand in hand with housing, the goal of this feature is to increase the sense of ownership players have in the game, but beyond that, to provide an end point for the colonization process. It is also intended to add a political sub-game to *PO*.

Design: "Wild space" planets can be turned into members of Confed via colonization. Players have the ability to buy homes in various floor plans. These buildings are highly customizable, and cost a maintenance fee on an ongoing basis. A given character may only own one house. Once enough homesteaders are on a given planet, the village they heretofore formed is now eligible to become a city. Homesteaders gain (probably via their basic house interface) the ability to vote for

one of the homesteaders as the governor. Players can choose not to have a governor, in which case the village remains a village and lawless. See the section on "Cities" for more detail.

When enough buildings are present on a planet and they organize sufficiently, they can form a town. Towns have a simple method of determining who the governor of the town is, and governors have a suite of special powers that enable them to control aspects of the town. The primary goal of town formation is to develop towns into safe areas that are policed by ConFed.

The governor of a city has special capabilities. He can dictate tax levels, with the money going into the city treasury. He can set the "laws" of the city, including whether or not the city is a safe zone or not, and to what extent its policing occurs—which costs money, which must come out of the treasury. He also has the capability to "force buy" any installation or building on the planet, for the purposes of controlling commerce, exiling players, and in general strong-arming his political dominance.

Only established cities can have certain facilities built: cloning facilities, banks, insurance systems, and stock and commodities markets. In addition, formally recognized cities with the basic features required can automatically become a newbie starting point.

Ground combat

Goals: To provide scope for those players who are not expert pilots, and provide a setting for colonization. Also to increase the range of missions that are available in the game.

Design: A simple combat system will make use of the same basic materials, damage types, and damage scaling factors that are used for space combat (eg, the same array of laser types, missile types, and armor types will be used for both systems). Combat will include both a standard melee loop (running automatically) and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple: spawning is actually initiated by scripts on the player, based on the equipment they have available, for example, rather than building spawn eggs or tables.

Creatures are generated using a standard array of skeletal animations, with meshes and textures applied atop them and scaled or deformed to achieve varied results. Creature statistics are drawn from a general table of level ranges, with some randomization. The AI is also generic, with the random addition of specific flags or scripts that can provide variety in the tactics used.

Modular ships

Goal: A "ship as avatar" philosophy that encourages personal attachment to ships and provides a deep play mechanic.

Design: The modular nature of the player-owned ships allows many different structural combinations, in addition to the ability to customize paint schemes, lighting configurations, and other vehicle-specific appearance features. Reflecting the "ship as avatar" philosophy, every player in the game can possess a ship as unique as their character.

Capital ships

Goal: To allow a group dynamic for piloting large ships, adding a new cooperative play mechanic to space combat.

Design: Capital ships will have multiple interfaces for differing subsystems, including power management, sensors, individual gun turrets, and other such systems. They will be designed to be streamlined enough to be run by one individual, but tuned such that a group that is cooperating will be able to get more out of the capital ship's performance.

Safe space and wild space

Goal: To provide a geographical metaphor for dividing up those players who are interested in player-vs-player activity and those who are not.

Design: We will provide initial safe zones using the fictional device of calling it Confed-controlled space. New sectors of space that are discovered and explored will begin life as wild space, which is not protected by Confed, and as players colonize the planets and join Confed, these sectors will change into safe zones as well if the players so choose. Seams and boundary issues will be handled by the fact that the entire volume on the other side of a jump gate will always be entirely of one type of space or the other.

Planet generation

Goal: We want to provide a dynamic, ever-changing map that players can affect in a significant way, and that provides a never-ending amount of varied territory to explore and visit.

Design: We will use the concept of a "fixed random seed" to generate the terrain. Identical algorithms on the server and client will use the same seed numbers (transmitted from the server to the client) to generate matching maps. This allows us to have as many maps as we have seeds, without hard drive footprint or significant bandwidth costs.

The algorithms will generate fractal terrain, but the algorithm will be tuned to provide dramatic landscapes that are suited for display using our engine. Individual planets will be only a few square kilometers at most, just large enough for a city to develop on. Because of the overhead perspective, we can have the psychological effect of distance scaling that we could not if we were using a first person perspective where you can see to the horizon.

The terrains of the planets will vary significantly, offering ice worlds, water worlds, swamps, mountains, plains, deserts, etc, as well as inhospitable atmospheres that require special equipment. Plant life is generated using morphing and deforming meshes: we create a suite of typical portions of plants and trees, and then mix and match, scale, and deform. Since we are planning on being set a significant amount of time after *Wing Commander: Prophecy*, the planets discovered can have abandoned Confed colonies on them, permitting the use of familiar plants and creatures. The life on the planets is also generated, using standard skeletal

animations, scaling, and morphing. A wide array of textures and colors appropriate to the setting are used to give the creatures a large range of visual variation. The behaviors of the creatures are fairly simple and generic—no sentient beings are found on the planets. Different creatures can however have somewhat different behaviors, based on the attachment of extremely simple basic AI scripts. Some of these creatures would be domesticable, and could therefore be pets.

Companies (guilds)

Goal: To provide a way for players to organize in groups, and to ensure that we can measure relative standing of groups. To ensure that players can invest in each other's success.

Design: Guilds in *Privateer Online* are formalized as companies. Companies have all the expected tools that guilds do, but in addition must also define a type of business for themselves (large guilds can run as "corporations" which subsume several different companies). Companies are measured against each other within their classification.

In addition, companies can list themselves on the in-game stock exchange, wherein players can buy and sell stock in each other's guilds. We intend to make it possible to check the stock market and engage in transactions via the Web.

Resource mining

Goals: We want to provide a slow-paced, fairly peaceful means of playing the game for those who are not interested in combat. This should also serve as the basic elements on which the rest of the game economy is built. At all times, a goal is to make this less expensive in bandwidth than it was with *UO*. Concomitant to that goal, a way to do this while offline is desired.

Design: Resource mining is done primarily via "installations" which are robotic devices such as automatic harvesters, miners, etc. As far as players can tell, these run regardless of whether or not the player is present. In fact, they merely timestamp the last visit, and compare timestamps at the next visit, and then introduce some random factors and calculations to arrive at the amount of stuff they have mined or harvested.

Missions and the mission board

Goal: A primary goal is to make it easy for players to find things to do in the game—to give them a clear goal. To allow players to easily advertise their wants and needs, providing missions for others to complete. Missions should also cover a wide range of possible activities, to appeal to the broadest possible range of players.

Design: Computer generated missions are tagged with difficulty ratings, mission types, and other such factors, so that we can sort the available missions for a player, showing them only ones which are within their capabilities. Mission types will include item or resource retrieval, mapping, dogfights, defense, etc.

Players will also be able to fill out a simple form in order to add missions to the mission board. These missions will be indistinguishable from the computer-generated missions.

Crafting

Goal: To provide an economic means of play for those players who are not interested in combat. To provide the engine on which the game's economy runs, by making supply and demand be dynamic.

Design: Every item in the game is intended to be a craftable item, made from the resources that players can mine or otherwise obtain. The technology trees for advancing in crafting will offer interesting branches so that players can specialize in crafting particular types of items. "Maker's marks" and other such methods for making prominent crafters famous will be part of the system as well, leading up to the ultimate reward of having newly crafted items named after you forever after in the game. The intent is to automate the tediousness of crafting items by using factories. Factory installations will be similar to resource mining installations, except with parameters players can affect in order to determine what will be produced using a simple menu system. Players will be able to use a variety of means to sell the resultant items.

Commodities markets and vendors

Goal: To give as many economic transaction tools as possible to players, in order to better drive the player-run economy.

Design: Commodities markets function on an auction system, whereby players place items or resources up in lots at specific prices, allowing them to sell items even when they are not present.

Specific items can also be obtained "boutique" style from specific player-run vendors. The intent is for these to be extremely similar to the *UO* feature.

Advancement system

Goal: We want to make it possible for players to back down an advancement tree and choose to take a different path—no character development choices are irrevocable. As much as possible, we would like to measure advancement in a very wide range of professions on a single ladder so that they can be fruitfully compared.

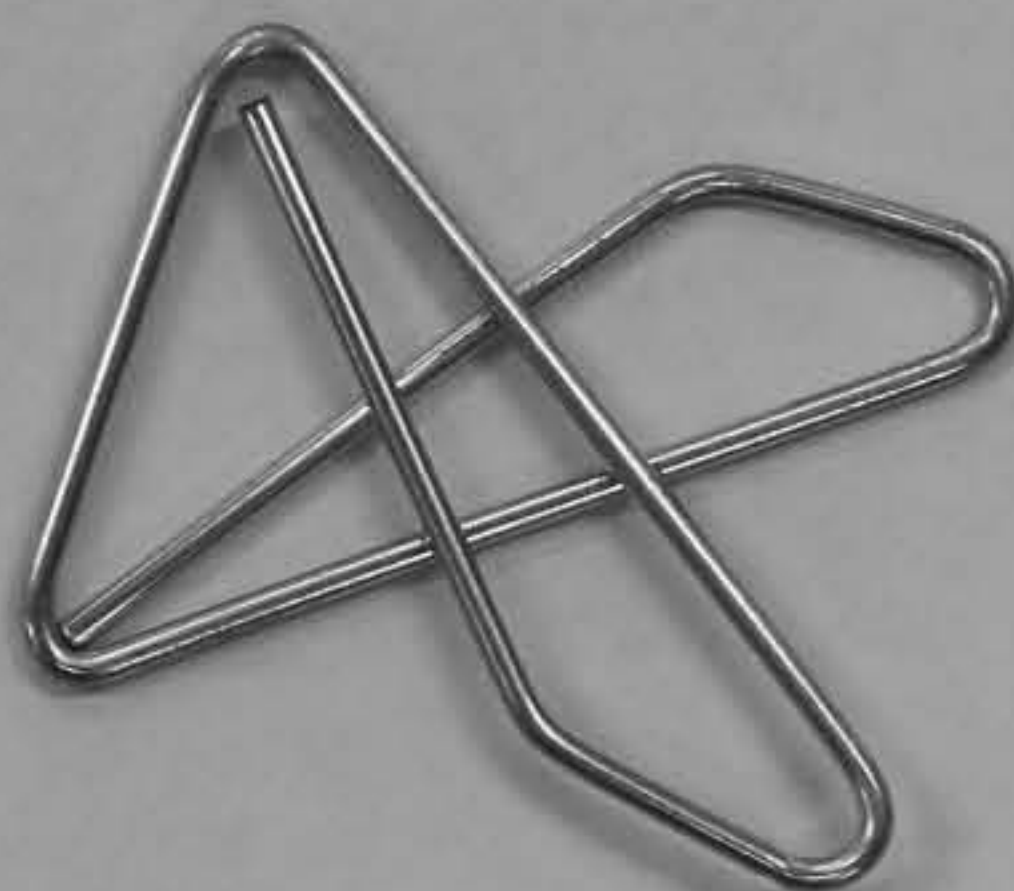
Design: Our preliminary design involves obtaining "certification" at levels of professions in all the professions. At lower levels certification is free or a one-time expense—as you go higher, you may be required to pay dues. Certification essentially opens up new capabilities for the profession (more items to craft, more tools to use, etc). Levels of certification will also require pre-qualification via other means, such as a required stat for number of kills for the mercenary profession, or number of well-regarded articles for a news ticker reporter.

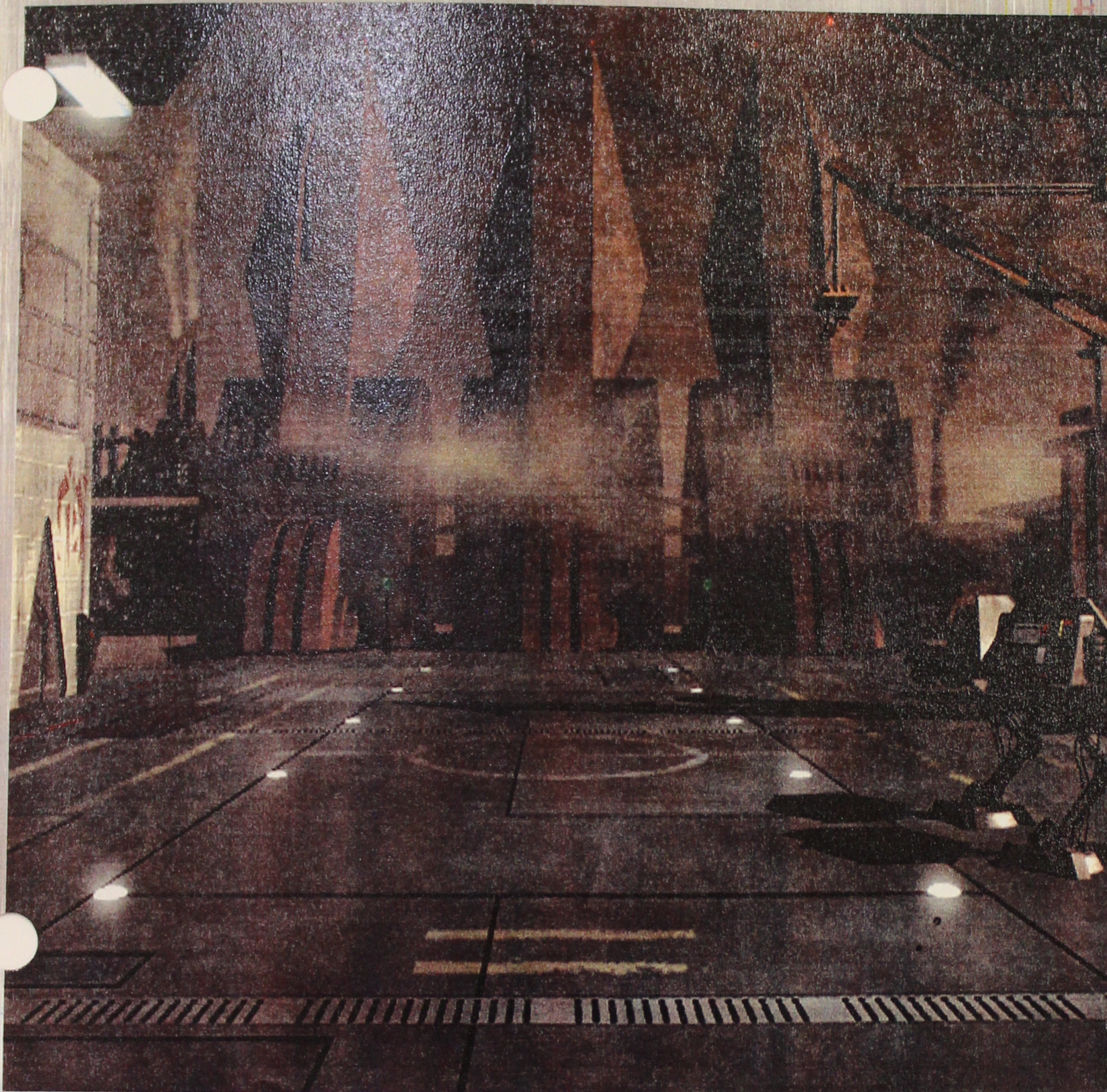
Putting all the professions on a monetary basis allows us to easily compare them, but using the other statistics for providing access to specific profession levels prevents people from being able to buy their way too high in a profession.

Storytelling

Goal: To provide world-class storytelling that is compelling enough to keep people logging in on a regular basis to find out the next development in the ongoing saga.

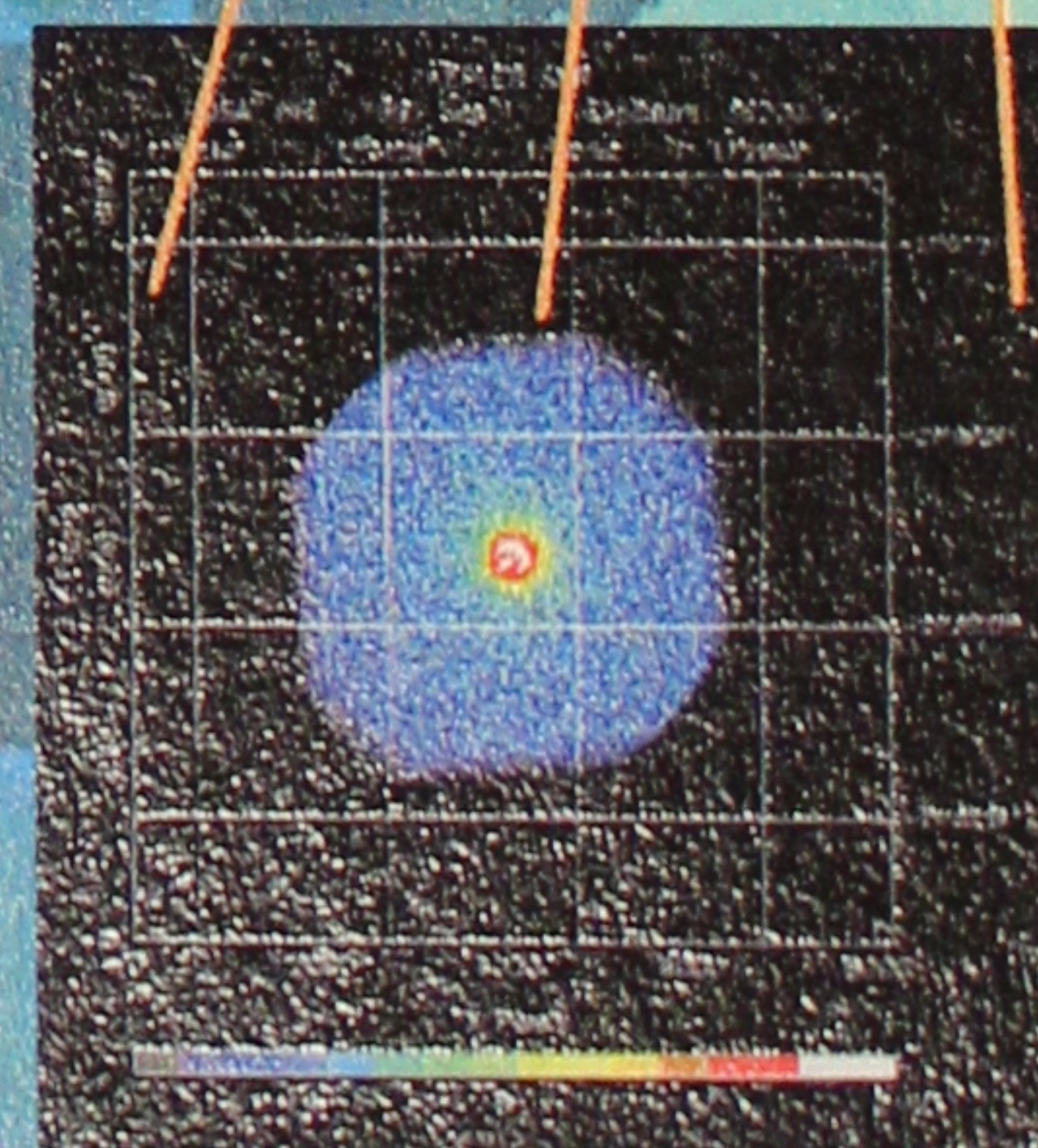
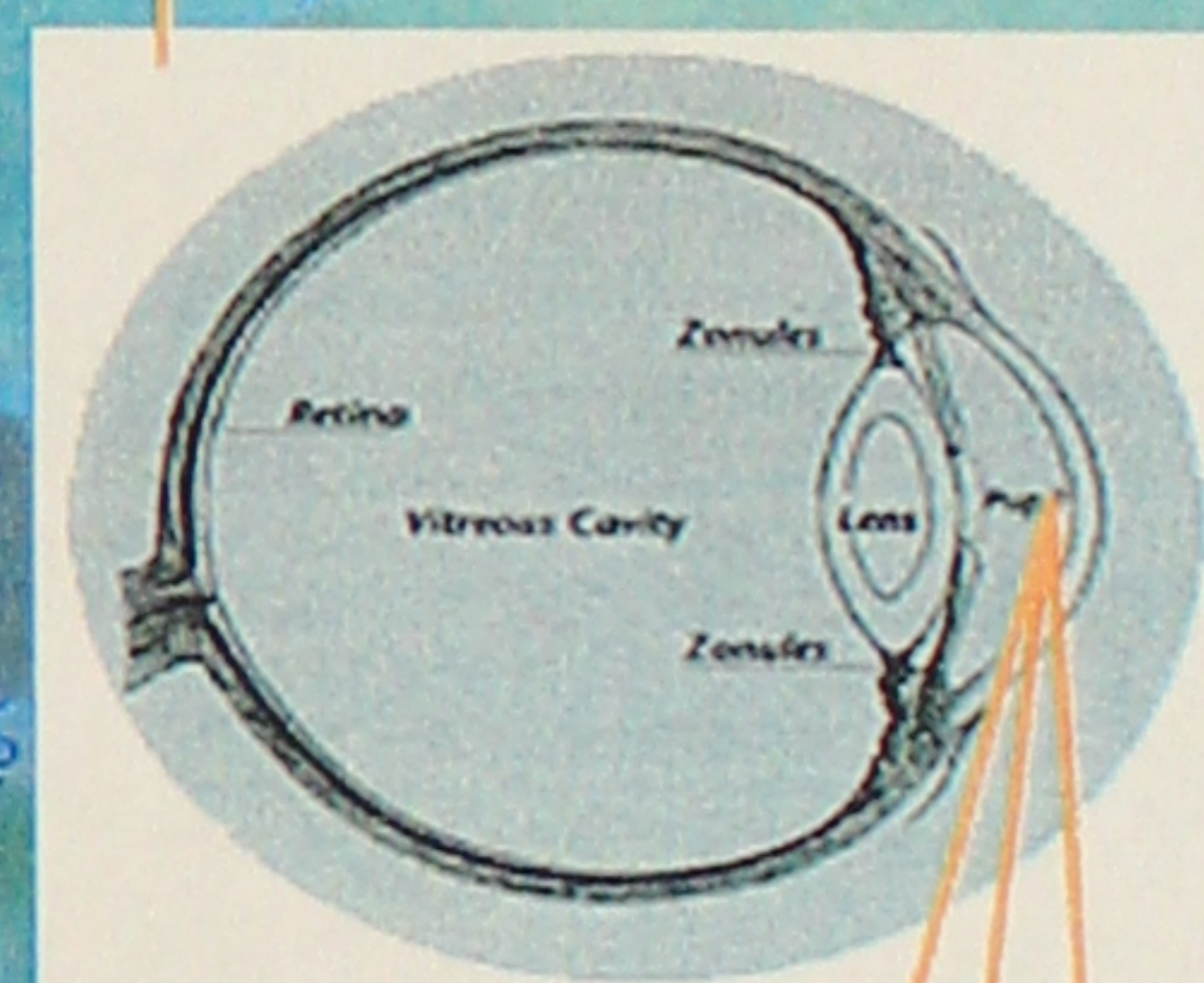
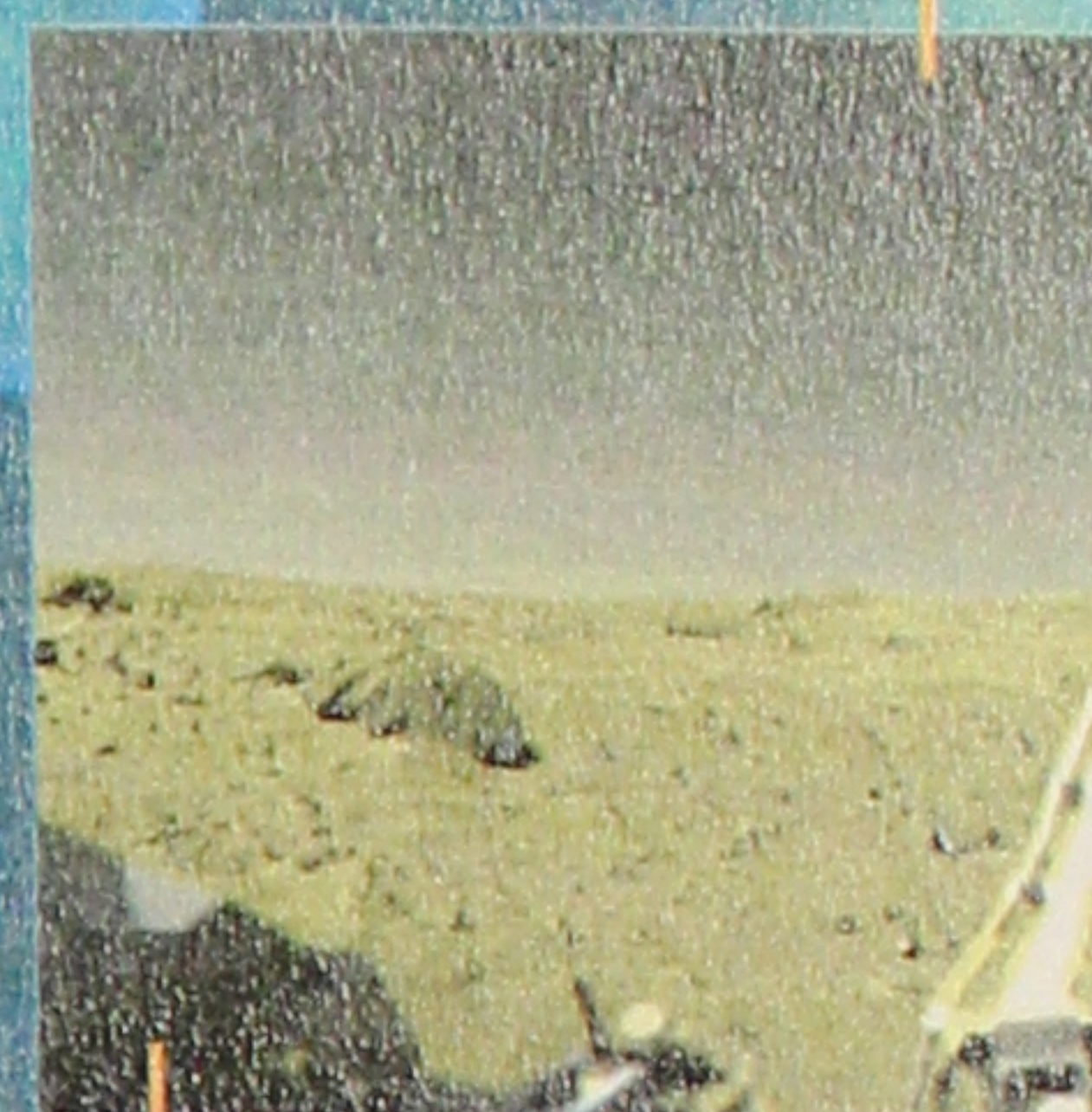
Design: We intend to make use of any tools or processes developed for *UO2*, such as the *Synthetic Cinema Plan*. We are also investigating the possibility of having a well-known science-fiction writer acting as a creative consultant to the project. We have engaged in preliminary talks with the science-fiction writer Bruce Sterling (award-winning author of *The Hacker Crackdown*, *Distraction*, and other bestsellers).

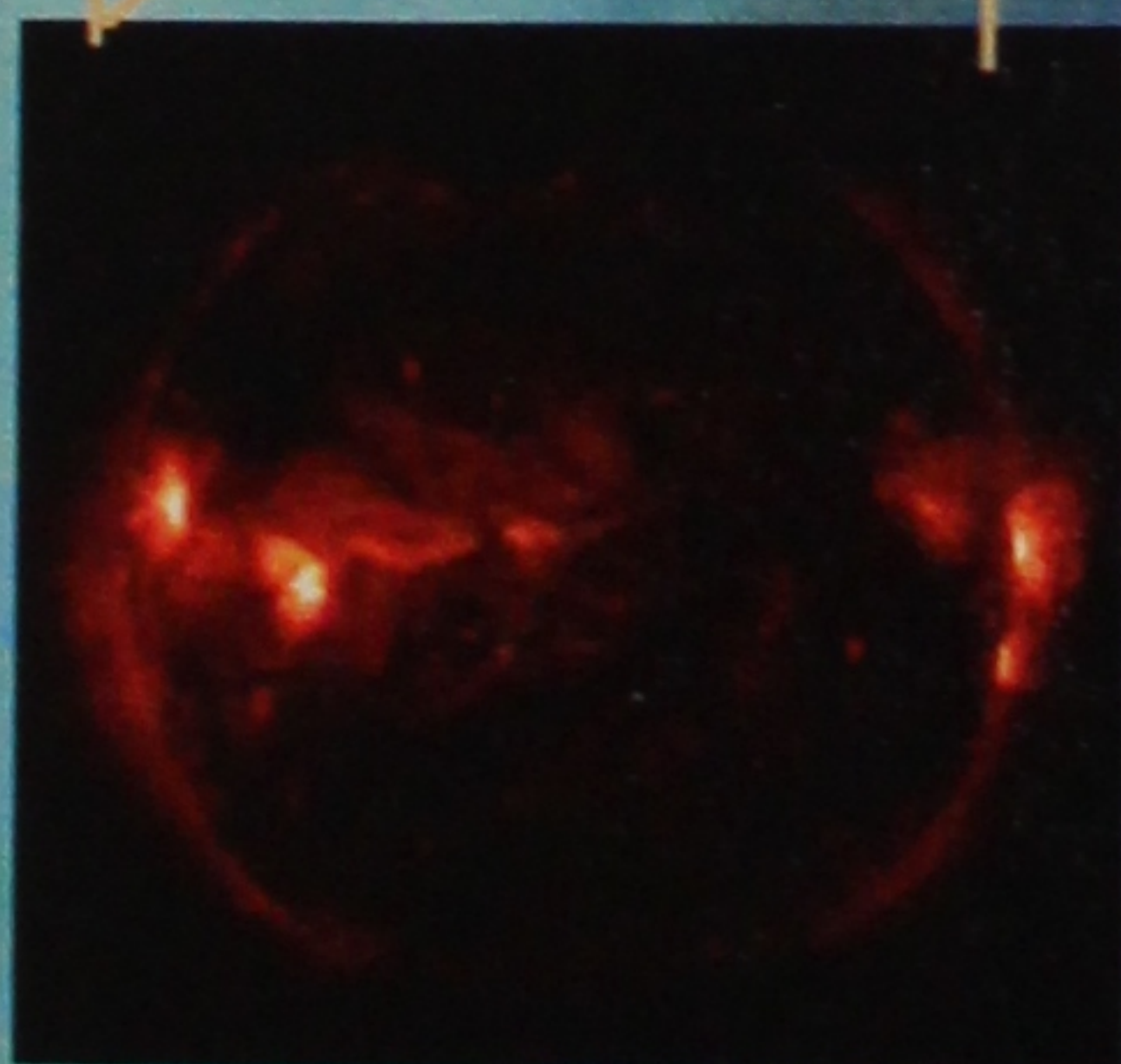
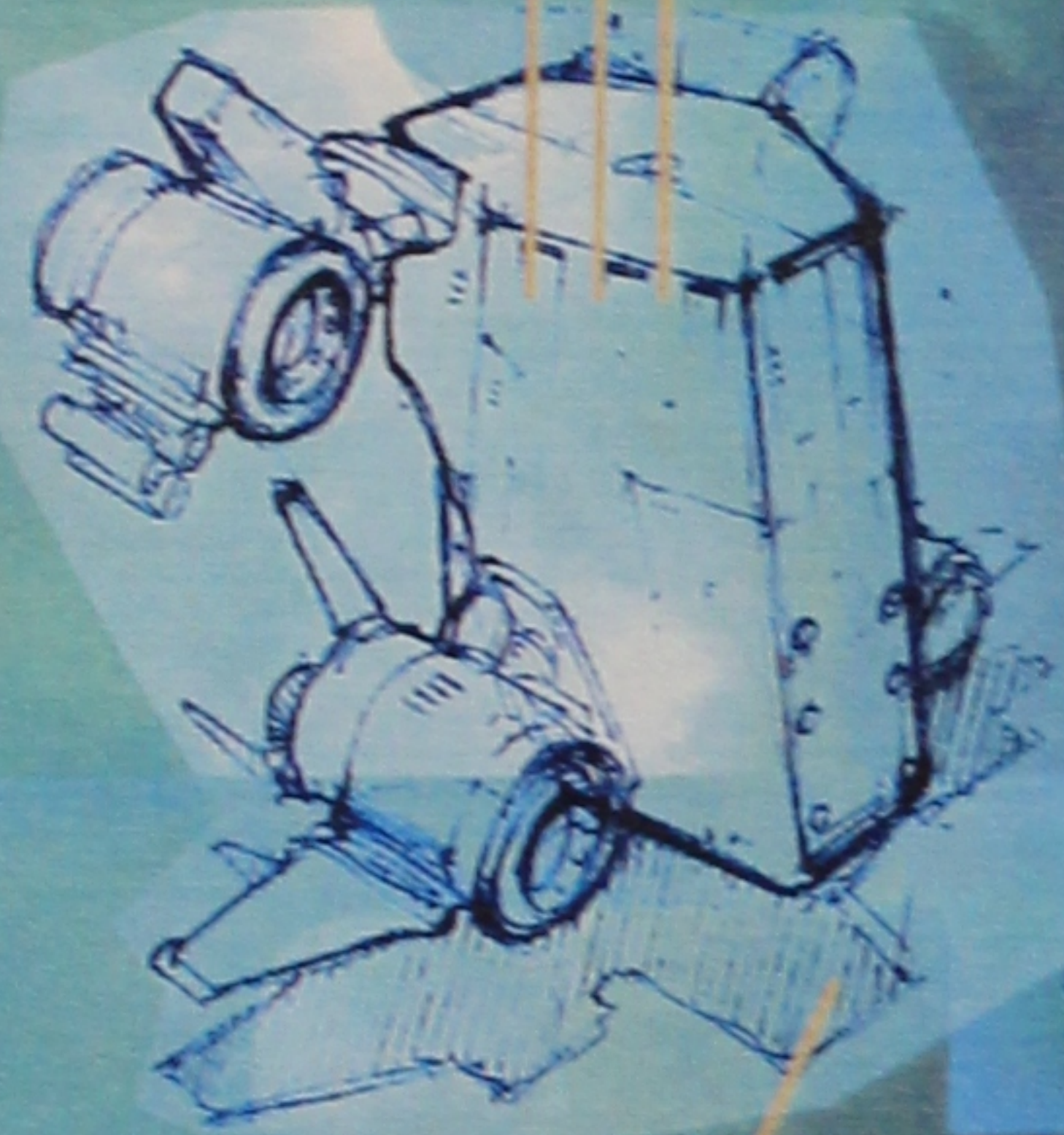




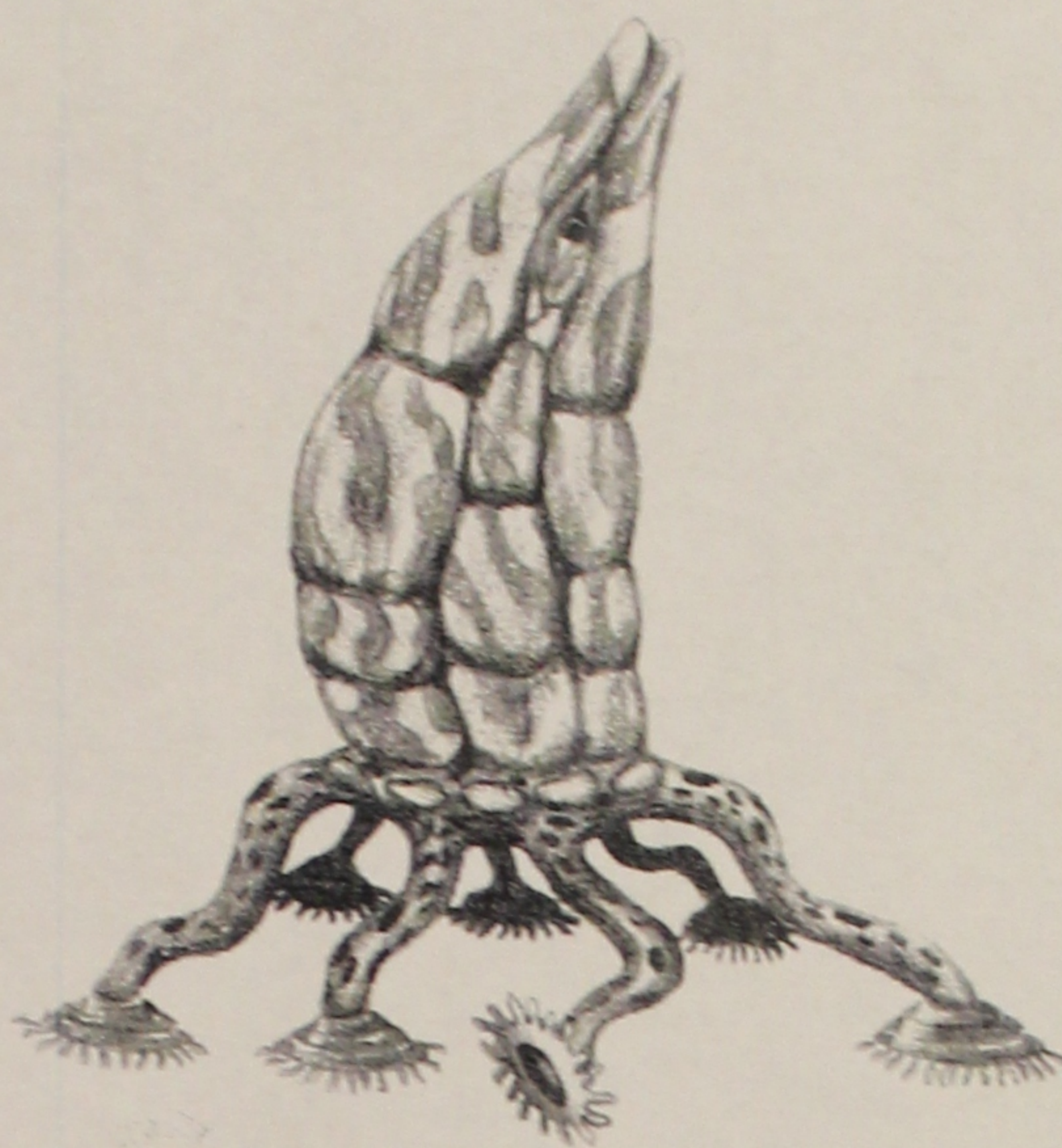
Design: A simple combat system will make use of the same basic materials, damage types, and damage scaling factors that are used for space combat (eg. the same array of laser types, missile types, and armor types will be used for both systems). Combat will include both a standard melee loop (running automatically) and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple: spawning is actually initiated by scripts on the player, based on the equipment they have available, for example, rather than building spawn eggs or tables.

Creatures are generated using a standard array of skeletal animations, with meshes and textures applied atop them and scaled or deformed to achieve varied results. Creature statistics are drawn from a general table of level ranges, with some randomization. The AI is also generic, with the random addition of specific flags or scripts that can provide variety in the tactics used.





how big is the universe? you tell me! **WORLDS TO BE FORGOTTEN**



scope for these players
pilots, and provide a
ization, also to increase
ions that are available

combat system will make
asic materials, damage
scaling factors that
e combat (eg, the same
array of laser types, missile types, and
armor types will be used for both systems).



Combat will include
loop (running auto
missile system, the
a hotkey or mouse b
is however designed
is actually initiate
player, based on the
available, for exam
spawn eggs or tables

Creatures are genera
of skeletal animatio
applied atop them an
achieve varied resul
drawn from a general
some randomization.
the random addition
that can provide var



Design risks

Systems in our game design that are new and will require prototyping are:

- **Server architecture**
The server architecture for *PO* is different from the server architectures used for both incarnations of *Ultima Online* in that it relies on launching and shutting down many small servers constantly, rather than running a few large servers that are persistent. This is perhaps the largest technological risk in developing this game. As part of our goals for the Technical Design Review, we intend to arrive at hard numbers for how many concurrent processes we can run, what the CPU utilization of those processes might be, and how large a server farm would be required to run the game.
- **Planet and space volume generation**
We expect to be able to prototype these well before we have a functioning server architecture, within the Simis engine. There are, of course, many instances of prior art and known algorithms for fractal terrain generation. The Simis engine also supports skybox generation already, though not to the level that we will require.
- **Flora and creature generation**
We plan to prototype these within the 3d modeling tool, and within the Simis object viewer. Many aspects of the technology are quite similar to what *Ultima Online 2* is attempting with its snap-together deforming meshes (see below), and we plan to leverage that technology or learning.
- **Economic advancement system and the stock market**
We intend to play paper-gaming sessions with this system in order to ascertain its strengths and weaknesses prior to implementation. The nature of the system is such that it should not be difficult to build a simple simulation of the system, either for paper-gaming purposes or for implementation as a simple standalone "black box" game.
- **Space combat "maneuver energy" system**
We intend to implement this system early on in the development process so that we can determine how well it functions. However, the nature of the system is such that it requires a functioning client-server to really determine its viability.
- **Web integration**
At this time, we do not have specific plans for prototyping this system. However, there are already basic forms of what we intend to do in place for *Ultima Online Live*, and *UO2* has further plans along these lines. We do not anticipate this technology being a substantial risk given the work already being done on this front.

There are other elements of the design which have not yet been tried, but which we anticipate will be tested as part of the development of *Ultima Online 2*. These include:

- **Snap-together deforming meshes**

This system will be used for handling equipment and clothing on avatars, and we plan to use it as well for the creature creation process. As part of our prototyping phase, we intend to engage in an extensive technology sharing effort with UO2. They are currently scheduled to have their solution for this system in mid-September.

- **Storytelling (*Synthetic Cinema* plan)**

At this time, we are relying on the development of the *Synthetic Cinema* plan as our storytelling solution, under the assumption that a common solution for storytelling needs across all ORIGIN products is the way to go. At this time, this system is still in early design phase. However, the nature of the system is such that it should not be tremendously difficult to integrate post-launch if it slips beyond all expectation.





Ground combat

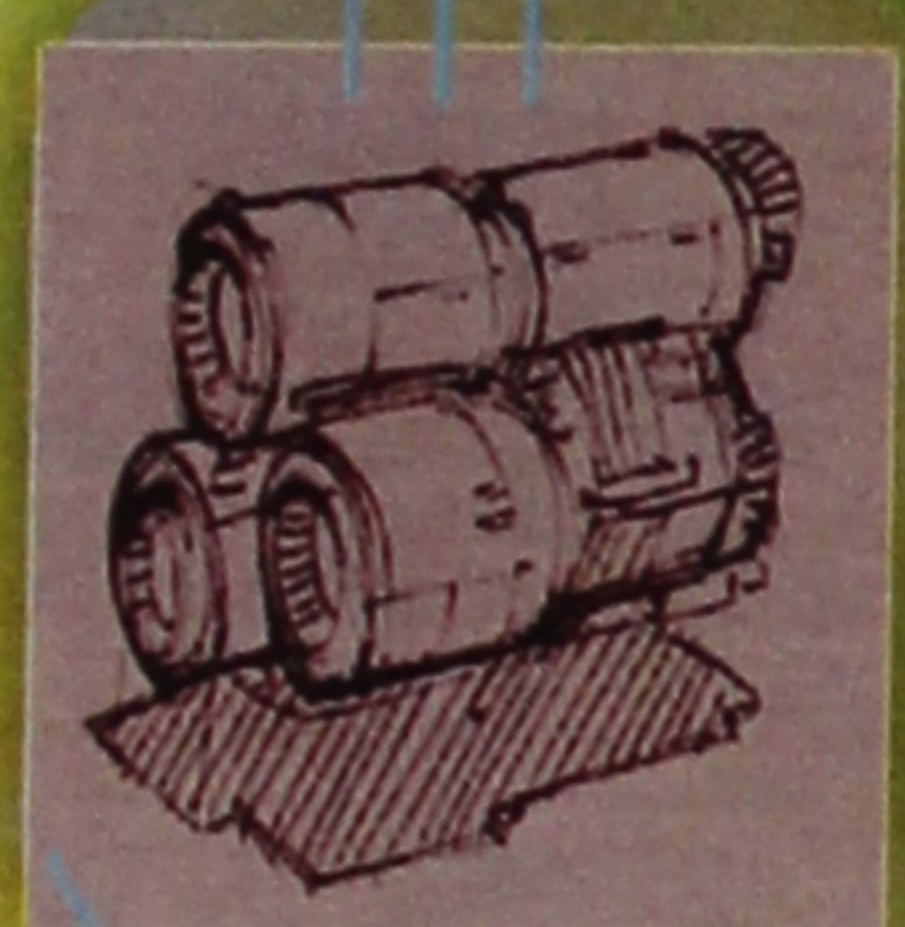
Goals: To provide scope for those players who are not expert pilots, and provide a setting for colonization. Also to increase the range of missions that are available in the game.

Design: A simple combat system will make use of the same basic materials, damage types, and damage scaling factors that are used for space combat (eg. the same array of laser types, missile types, and armor types will be used for both systems). Combat will include both a standard melee loop (running automatically) and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple: spawning is actually initiated by scripts on the player, based on the equipment they have available, for example, rather than building spawn eggs or tables.

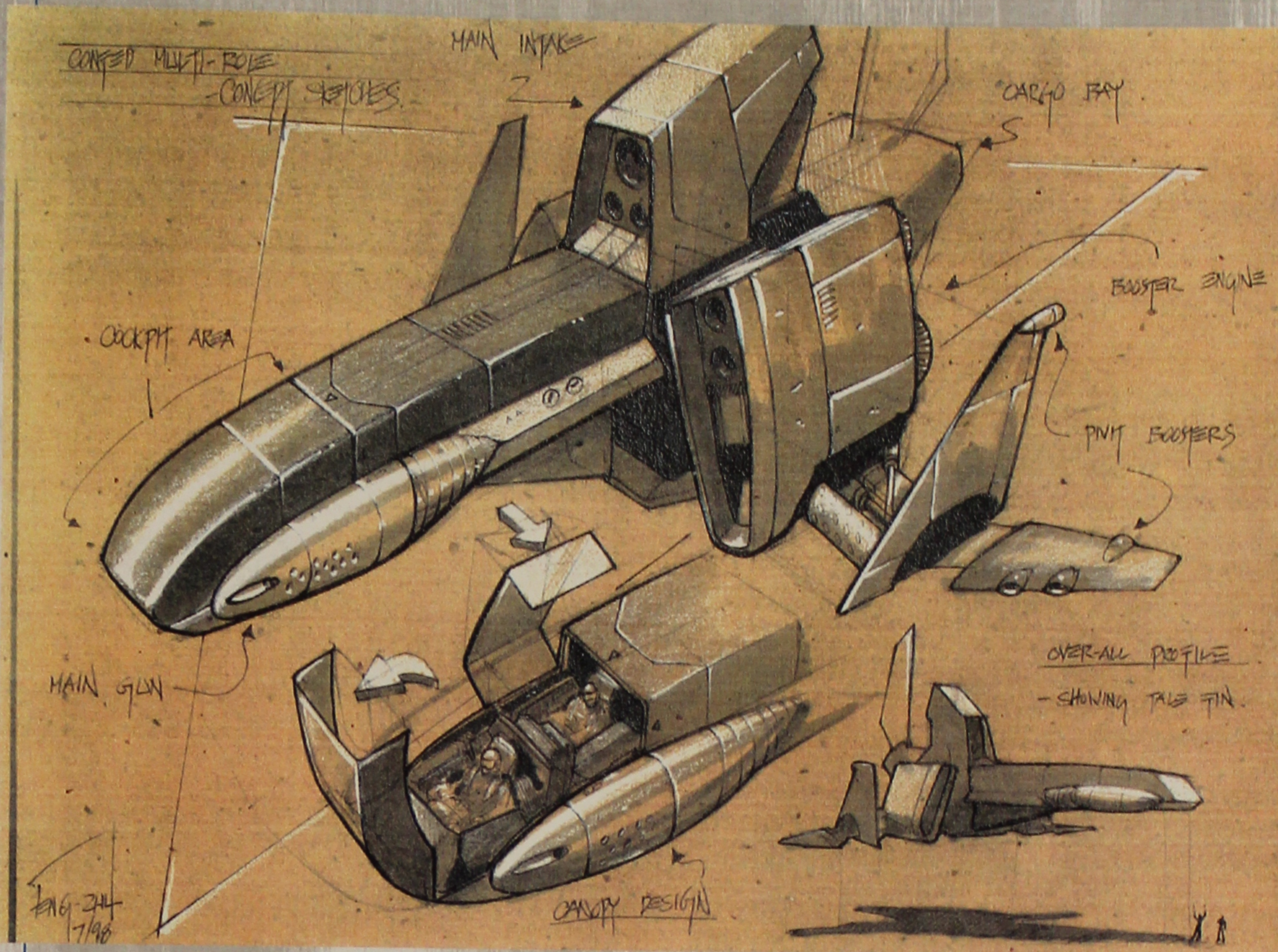
Creatures are generated using a standard array of skeletal animations, with meshes and texture applied atop them and scaled or deformed to achieve varied results. Creature statistics are drawn from a general table of level ranges, with some randomization. The AI is also generic, with the random addition of specific flags or scripts that can provide variety in the tactics used.



WARRIORS TO BE CHALLENGED

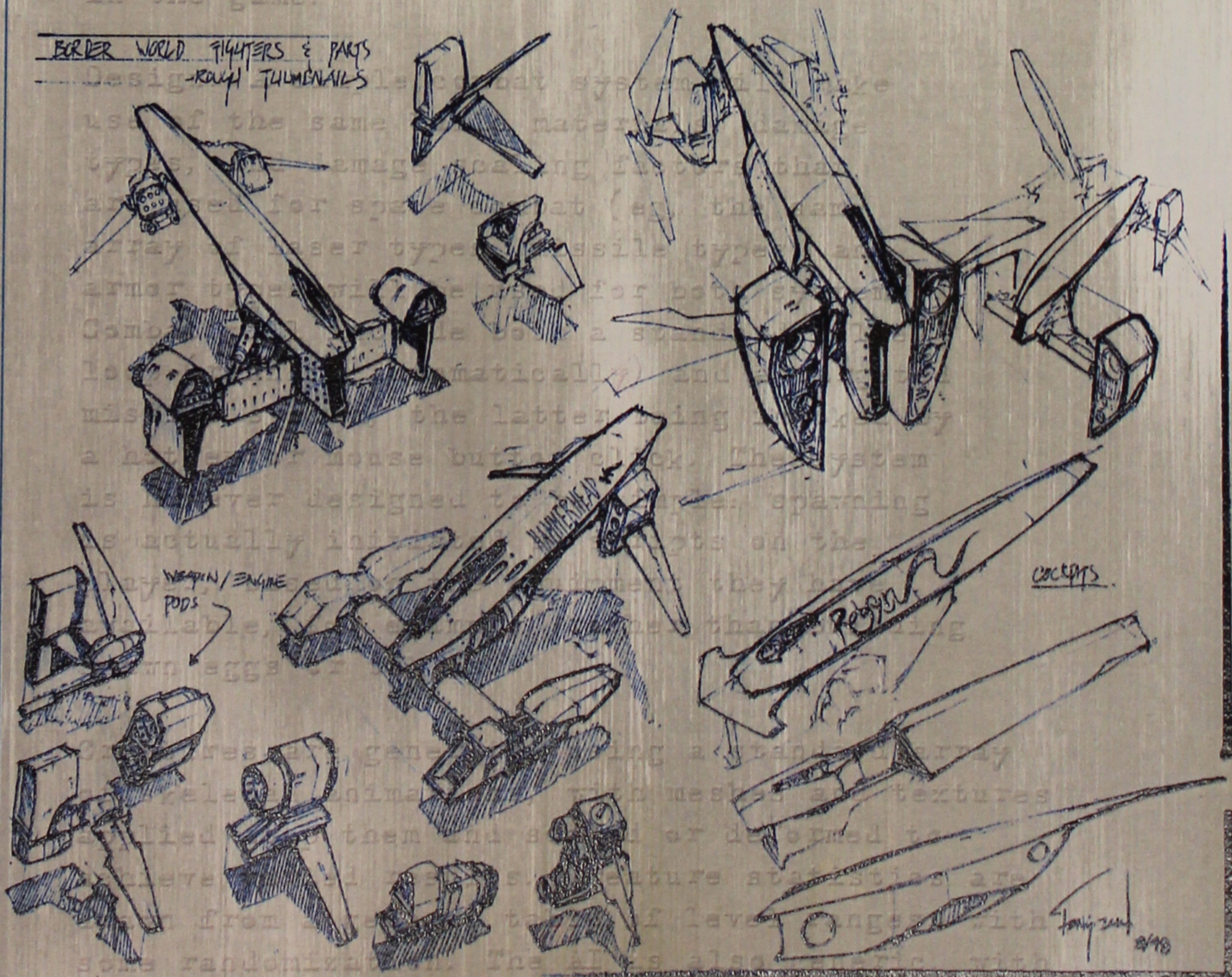


how big is the universe? you tell me! it's a big, big universe! it's a big, big universe! it's a big, big universe!



Ground combat

Goals: To provide scope for those players who are not expert pilots, and provide a setting for colonization. Also to increase the range of missions that are available in the game.



The random addition of specific flags or scripts that can provide variety in the tactics used.

Scope

The following are highly preliminary numbers:

- **Ship models**
This is the largest portion of the 3d mesh data. We would like to include models from *all* the ship types used in the history of the *Wing Commander* and *Privateer* series. Fortunately, we still have many of the models archived, in 3DStudio format. We anticipate the number of basic ship models therefore numbering well over 50.
- **Ship textures**
The use of hued textures means that we do not anticipate having to custom-texture every ship model. However, the ability to place different textures on your ships is a prime draw for the game. We therefore expect to need quite a lot of different ship textures.
- **Creature textures**
We are targeting having only a few dozen different huable textures for creatures.
- **Creature skeletal animations**
Our model for creatures involves using only a half-dozen or so skeletal animations: bipedal humanoid, quadriped, arachnid, etc. We anticipate being able to reuse many animations from *Ultima Online 2*, particularly motion capture data for the avatars.
- **Creature models**
Our plan is to use morphing and deforming meshes. We expect to have only a few meshes per animation skeleton, bringing the total meshes to a number well under 50.
- **Creature statistical data**
We intend to make use of only a few internal levels worth of creatures, allowing us to re-use the statistics with random factors.
- **Plant models**
Our intent is to use deforming meshes for plant features. Our list of meshes is currently under a dozen.
- **Plant textures**
Our list of huable textures needed for planets is currently under a dozen.
- **Other art assets**
We will require several hundred pieces of interface artwork, built modularly so that we can use them with the interface description language.

COMMODITIES EXCHANGE

4974

5783

8695

75342

876241

900

7846

86242

963

8562

972678

536

9387465

5267

7693

74632

7465

4974

5783

8695

75342

876241

900

7846

86242

963

8562

972678

536

9387465

5267

7693

74632

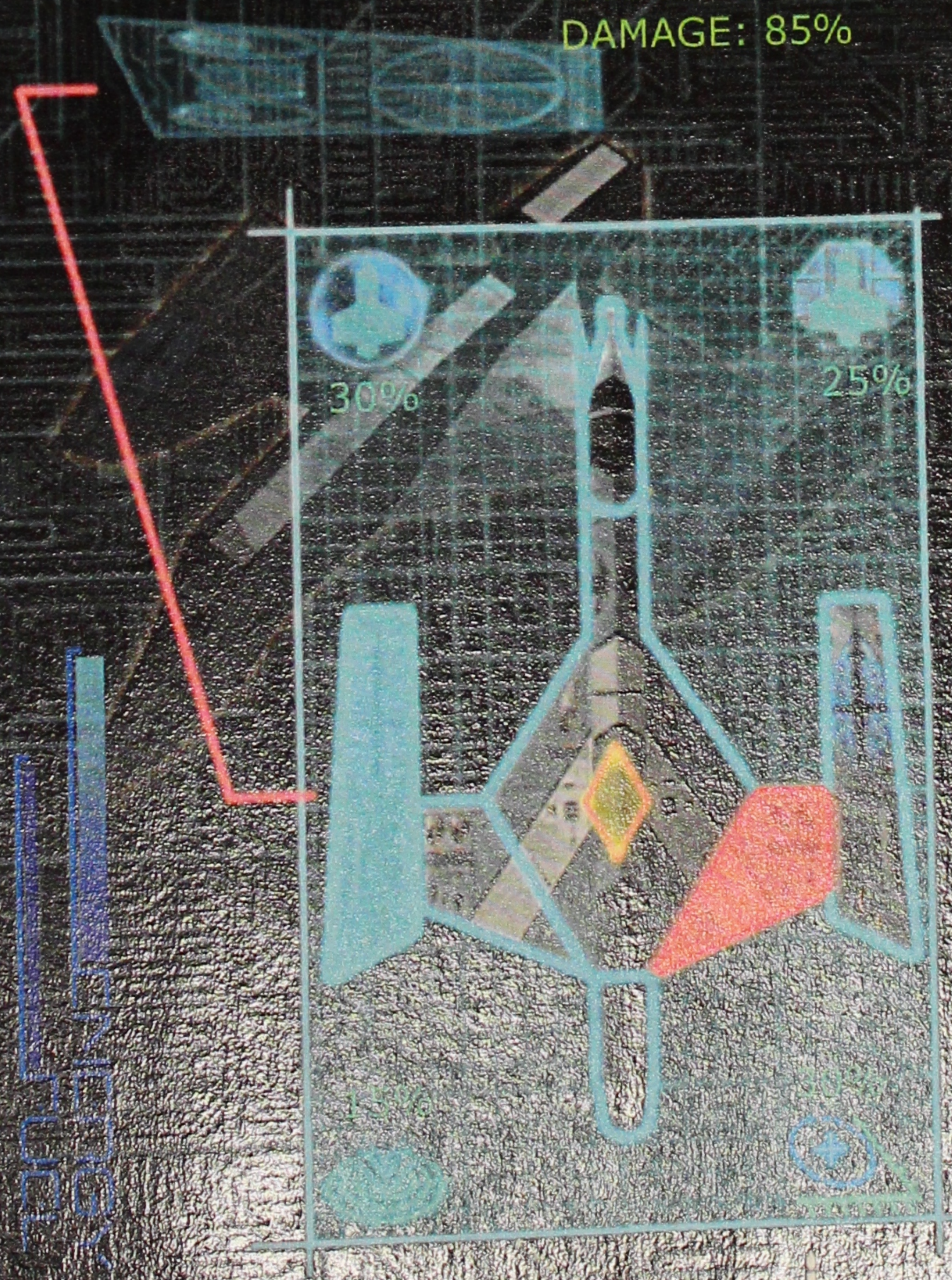
7465

SHIP

MODIFICATIONS: 0

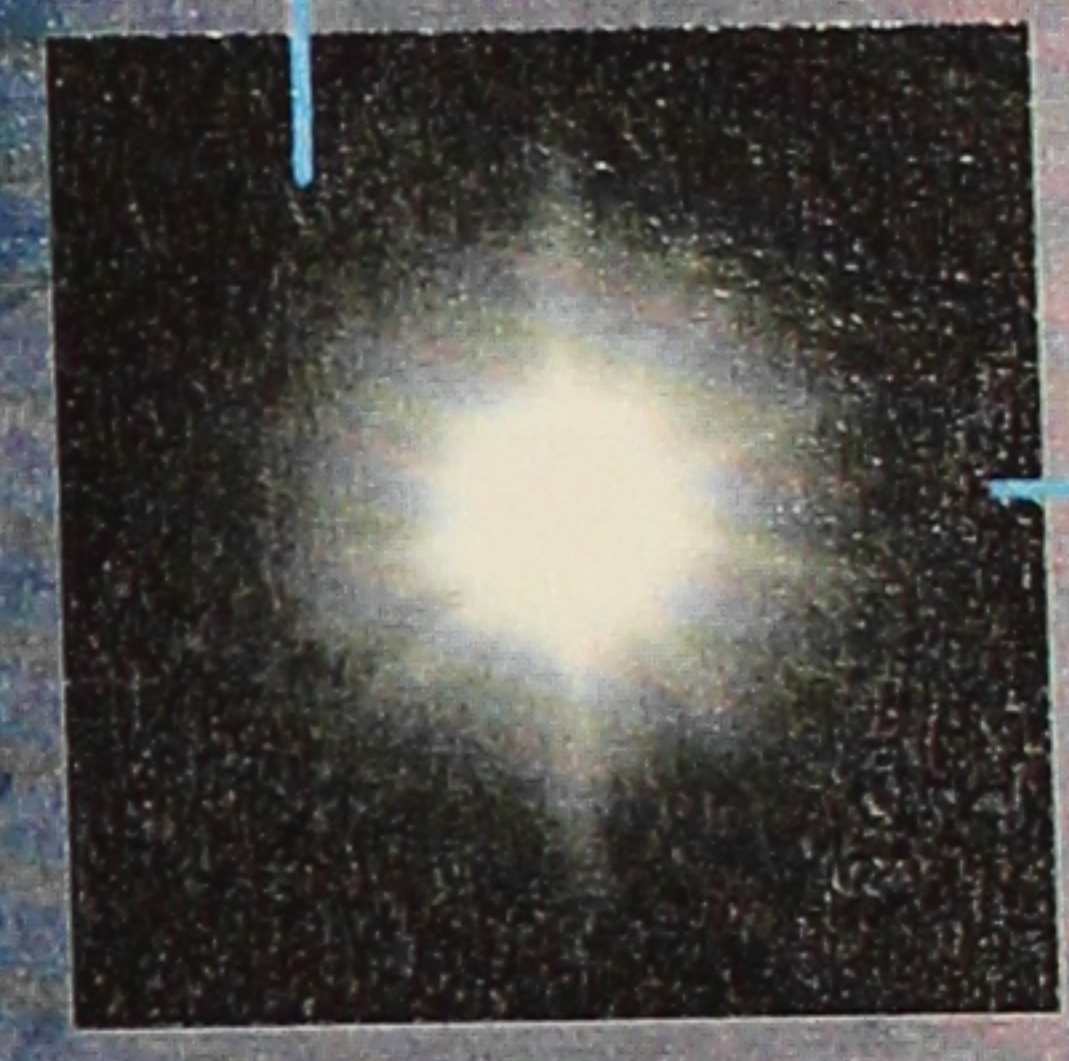
PRICE: 25000

DAMAGE: 85%



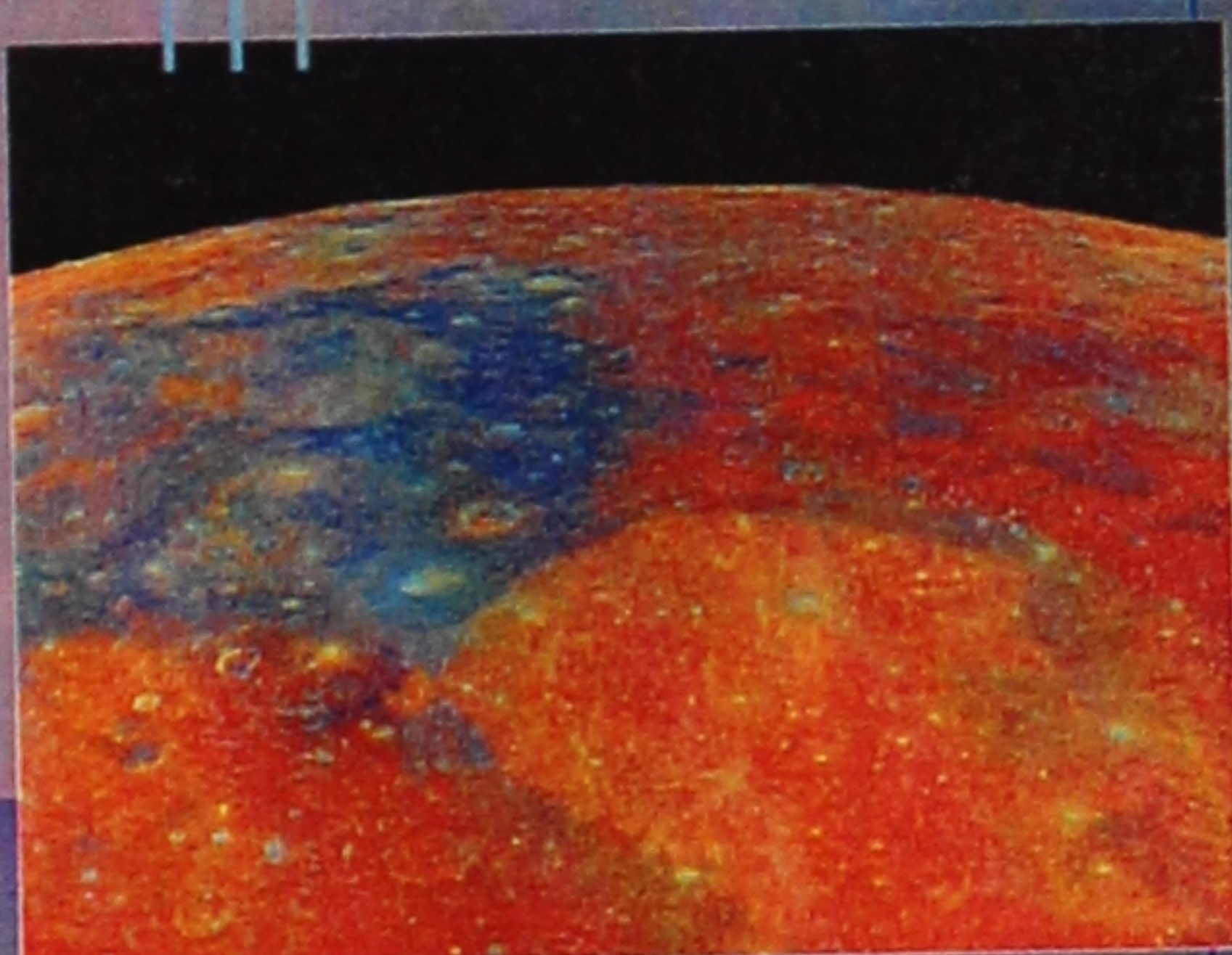
BANK 69 37500

EXPERIENCE TO BE GAINED

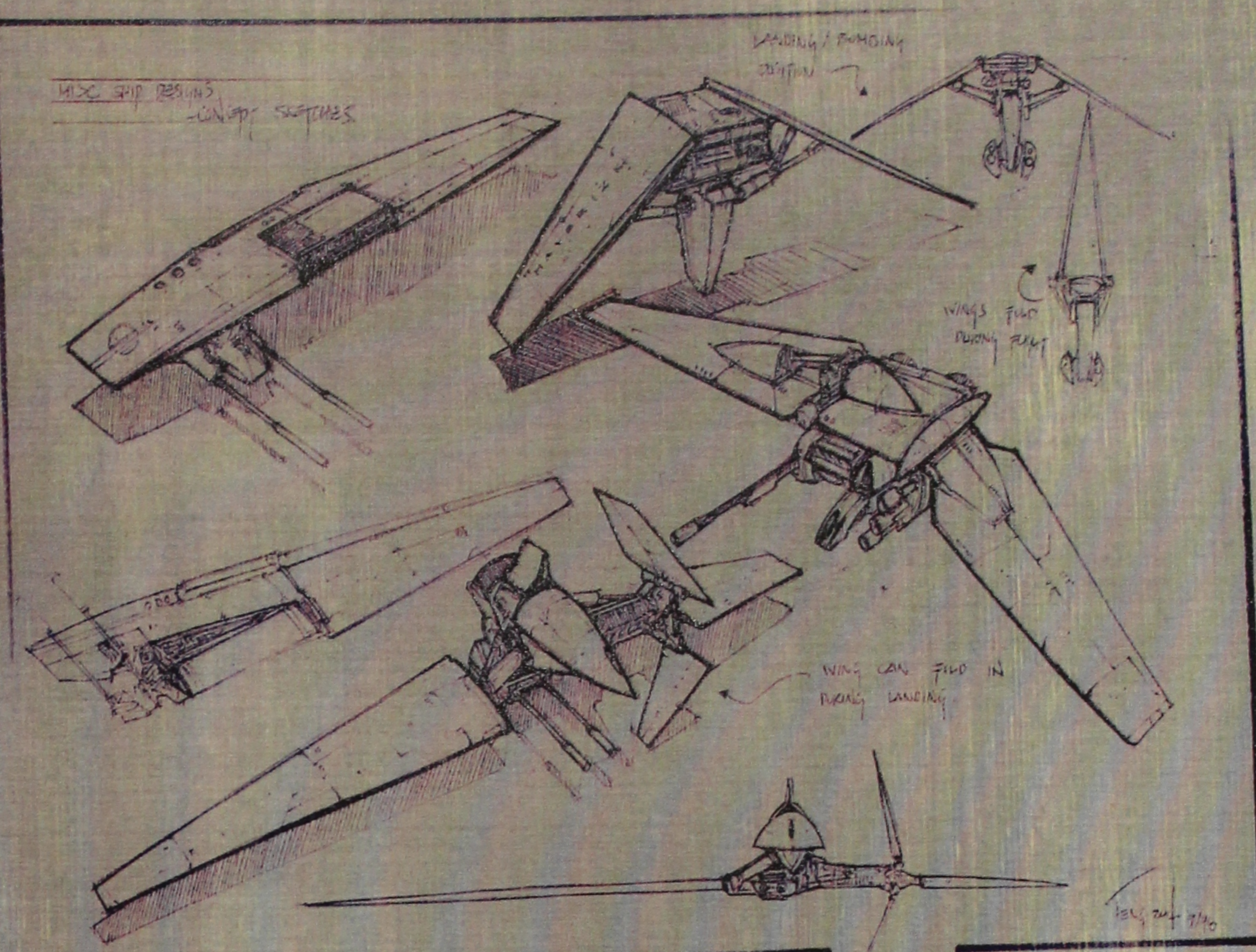




Swimming pool goes here



WARRIORS BEING TESTED TO EXTREME

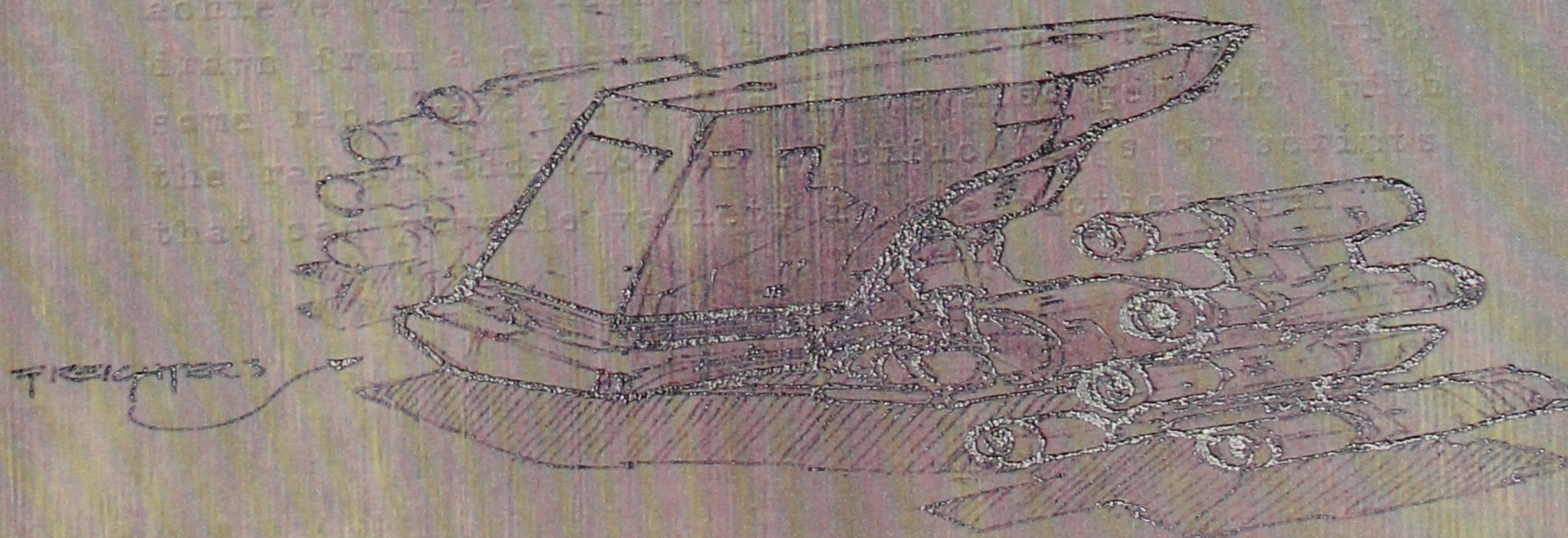


Ground combat

scale. To provide scope for those who are not expert piloted and pray setting for colonization. Also to the range of missions that are available in the game.

Design: A simple combat system will use of the same basic materials, sea types, and damage scaling factors that are used for space combat (eg, the array of laser types, missile types, armor types will be used for both systems). Combat will include both a standard wheel loop (running automatically), and a targeted missile system, the latter being invoked by a hotkey or mouse button click. The system is however designed to be simple: spawning is actually initiated by scripts on the player, based on the equipment they have available, for example, rather than building spawn eggs or tables.

Creatures are generated using a standard array of skeletal animations, with meshes and textures applied atop them and scaled or deformed to achieve varied results. Creature statistics are drawn from a standard table, with some variation in the way they are used, with the result of the creature's behavior or scripts that control its actions.



Style

Art

This isn't the dark militant world of *Wing Commander*, this is a world worthy of pioneers, adventurers, and settlers. One look around the initial planet city of New Krieger, and one can tell that the Corporate/Confed re-constructivists are intent on rebuilding a prominent universe. The buildings are sleek, clean, and, colossal. Enormous mosaic murals, central fountains, playing children, and lots of manicured plant-life create a peaceful friendly place for you to shop for gear, solicit for work, get a license, and prepare for your journey into the heavens. Simple posters promote planets and solar systems, offering both real estate and adventure packages. A small one-man hover shuttle pulls up and asks where you'd like to go. While the uniformed driver zips you across town he goes on at length about the booming economy. When it's time to leave the city, a monorail will take you to the central port, offering one last look at the city before you fly out. It's a beautiful, bright, softly shaped place, perfect in function and appearance. It's like a Syd Mead extravaganza. As the monorail enters the port, a spacecraft darts out from the opened hanger and off into the sky.

Not all of the world is going to be as perfect as New Krieger would promise, but it's guaranteed to be an adventure.

The world of *Wing Commander Online: Privateer* will be designed with a complex community in mind. The planet surfaces, vehicles, and avatars, will be as diversified as our audience. This is crucial since the greater the variance between worlds the more adventurous players are inclined to be. Some worlds will tend to be reminiscent of the hard edge military standards of the Con Fed, while others will lean more towards a harmonious environment by using special mud and wood unique to that planet's environment. Worlds governed by cruel pirates or ruthless smugglers will be downright alien in appearance. Anyone brave enough to walk around these cities, might believe they seeing through the eyes of a tapeworm. They'll be worlds every bit as exotic and far reaching as those in the *Star Wars* epics. At a later date, we may introduce new and totally bizarre planets that have been designed by a guest artist the likes of Bruval, Syd Mead, or Wayne Barlow, and then realized by our staff.

The most important factor in what determines the look of a world will be what the players make of it. Textures and meshes will decay into slums without proper upkeep. Or a structure may be upgraded using more pristine building materials. Spacecraft will range from heaps to aerodynamic mastery, and will be jazzed with color and decals to suit the attitude of its owner. That's right, Gear Heads, that chain link steering wheel will be in stock.

The job of the art staff will be to trump up excitement over a vast universe, and then create the building blocks for our audience to make of it what they will.

Audio and music

Ambiences

- The world will be full of 3D/THX/Dolby Surround encoded ambiences. Examples will include Capital Ship, cockpits, inside buildings and outdoors in the cities.

Music

- Traditional Wing Commander music (composed by George Oldziej) will connect the past with the present. There will also be remix tracks in the game from established recording artists such as Nine Inch Nails, Smashing Pumpkins, etc.
- The music for WCOP will be thematic, with certain areas or activities being connected to corresponding pieces of music. Using EAC's AEMS MIDI Player engine we will be able to provide players with fully interactive music. Controllers can be set to adjust certain elements of the music, such as tempo and instrumentation, based on the current events taking place in the game. The EAC AEMS engine combined with custom sound sets will allow us to deliver music that requires extremely low overhead and that sounds universal on all player's systems. Using these custom sound sets will also give us the ability to provide the players with new music via periodic downloadable patches.

Sound Effects

- Traditional Wing Commander weapon sounds will be enhanced and used to connect this release with previous versions of Wing Commander.
- Controllers similar to those used to make the music interactive can be applied to SFX allowing for a huge variety of sounds to be generated from a single .wav file. These controllers can also be used to "place" SFXs three dimensionally creating an extremely realistic and life-like audio environment. This will allow for incredibly realistic sound effects with very little space consumption.

Spatial Effects

- Using the EAC AEMS engine will allow us to incorporate software generated effects (reverb, chorus, delay, etc.) into the game. This will ensure that all players experience the same interactive effects regardless of which sound card they are using.

Speech

- Interactive server generated mission dialogue, voice chat between multiple players (premium shard) and "wingman style" command and reply dialogue via Macro sets will all be a part of game play.

Streaming Music

- Using Direct Music will give us the capability to stream thematic MIDI music to our players in real-time. For example, we can use this for celebrating holidays in WCOP. The MIDI files are extremely small and would take only a few seconds to download. We will also provide a tool for IGMs to use thematic music in their live events.

Intended Audience

What is the intended audience?

The intended audiences for this product are:

1. Wing Commander/space combat gamers, a segment size of 1.5 million+
2. Online RPG players, a segment size of 700,000+
3. Science fiction fans with Internet access, a segment size of 1 million+

Why will this design appeal to that audience?

1. Wing Commander/space combat gamers
 - a. Customizable modular ships
 - b. An intense combat experience with real people
 - c. Multi-position capital ships
 - d. Community building through squadron organization and leader boards
 - e. Wide range of weapons and defensive systems supporting numerous strategies.
2. Online RPG players
 - a. Highly customizable 3D character avatars.
 - b. Detailed and extensible skill systems.
 - c. Ownership of persistent and functional in-game facilities.
 - d. Dozens of non-combat professions.
 - e. A player-driven economy
 - f. Limitless game play with player generated missions
3. Science fiction fans with Internet access
 - a. Attractive and familiar high concept
 - b. Exploration of an interesting and unique science fiction setting
 - c. Low barriers to entry

Target platform

The following numbers are highly preliminary, as we have not yet done our TDR. In particular, the CPU targets as chosen based on the current bottom of the line and top of the line, and the footprint number is intentionally inflated from our expectations.

Minimum CPU: Celeron 400

Target CPU: Pentium III 600

3d card: TNT 16mb/Voodoo2 12mb

RAM: 64mb

Connectivity: 33.6K modem

Footprint: 350MB

Competitive Landscape

The following products are ranked according to the potential threat they offer us.

- *Product name:* Star Wars Online
Publisher: Presumably Lucasarts
Projected release date: No data

Rumors have been circulating that Lucas is pursuing this opportunity. This title would be a major competitor on the strength of its license, regardless of the quality of the game itself.

- *Product name:* Star Trek Online
Publisher: Paramount and an external development partner
Projected release date: No data

A similar situation applies as with *Star Wars*, although the *Star Trek* license has not enjoyed nearly as much success in the gaming arena to date.

- *Product name:* Freelancer
Publisher: Microsoft, developed by Digital Anvil
Projected release date: The Single Player is scheduled for Late 2000, with a massively multiplayer add-on '6 months after'.

Freelancer is an updated recreation of the original *Privateer*. The primary thrust of the massively multiplayer game is reported to involve an in-depth economic and trading model. Press coverage to date indicates *Freelancer* will have a much smaller scope than PO, with limited or non-existent ground and community-building elements. Some of the people who developed the original *Wing Commander* and *Privateer* games are working on this title, notably Chris Roberts.

- *Product name:* Allegiance
Publisher: Microsoft
Projected release date: Winter of 2000

Allegiance is an internally developed project that will be exclusively available on the Microsoft Gaming Zone. Focusing exclusively on space combat, it's best described as a "Tribes" analog and does not involve a persistent world environment.

Localization

We intend to fully localize both client and server. The languages targeted for localization are, in this order:

- German
- French
- Japanese
- Spanish
- Korean
- Chinese (Traditional, Simplified)

The estimated in-game text is 50,000 words. The estimated documentation text is 75,000 words. We are assuming a translation rate of 2500 words per day, per translator, and 1-3 translators on each language. The average translation cost is then \$.26 per word. The editing cost is \$.05 per word for an additional editor (optional if more than one translator is working on the text).

With these figures, the in-game text localization time and cost is 10 man-days, or \$15,500 per language. The documentation localization time and cost is 30 man-days, or \$20,750 per language. (Revisions and corrections typically add up to 50% to the final cost.)

Bottom Line: Text Localization Per Language: \$36,250- 54,375

The estimated in-game audio is 15,000 words (we are specifying a high number intentionally in order to provide a worst-case scenario). We are assuming 5000 recorded words per day, with recording studio fees of \$1,100 per day. With three voice actors at \$600 per day (this figure is variable per territory), we are looking at 6 man-days to first translate the text, at a cost of \$4650. Localized audio recording time and cost would be 3 days, for \$5100.

Bottom Line: Audio Recording Costs Per Language: \$9750.

Total Localization Costs: \$46,000-64125 per language.

Roll-out

The initial release of the game is the public beta test. We are planning on having the code complete for all game systems at this point, of course, but we may not wish to tip our hand to the competition regarding all of the systems that will be in place in the final release. For example, we may not activate all the website play elements during the public beta, reserving them for testing with a carefully selected closed beta group.

We are targeting a period of approximately six months from 1st playable to launch. The public open and closed betas are expected to be 2-3 months of this time.

At launch, we are targeting having all the features listed in our minimum feature set document available to players.

Elements targeted for roll-out post-launch as part of the Live service include:

- Apartment buildings. These are forms of housing that do not require players to buy a house of their own.
- Pets. This system is easily added post-launch assuming creatures are present in the game.
- Embedded games. These would include embedded diversions such as gambling within the game, races, fighting arenas, specialized simulations within simulator training spaces, etc.
- Additional data. This would include new ships, objects, etc.
- Expanded crafting. This would involve more items to craft and manufacture, new types of objects and capabilities, and new types of resources that are discovered.

Elements that are proposed for a potential Gold edition include:

- The ability to board capital ships. This assumes an adaptation of the ground portion of the game for use on the interior of capital ships.
- Additional player races. This would involve creating Kilrathi or other races as player options, plus providing alternate training mechanisms, missions, ships, and starting locations for players who choose this option.
- A full Tamagotchi or *Monster Rancher* system for pets.
- More database assets. This involves more ships, more planet types, more textures, etc.
- Ports to other platforms. A Linux port is an example of potential larger market penetration that could come with a Gold edition.

Priorities

For launch, we must focus initially on systems that meet the following criteria:

1. They serve as initial attractors to the product.
2. They serve to bring in the required mix of player types into the initial player base.
3. They provide sufficient retention to keep players around while other features are being developed.

Our objective is to complete all the basic systems of the game, and launch with the minimum content we feel necessary to make the game systems viable. Due to the interlocking nature of the game design, it is very difficult to remove a system whole. It is also very difficult to add a complete game system post-launch.

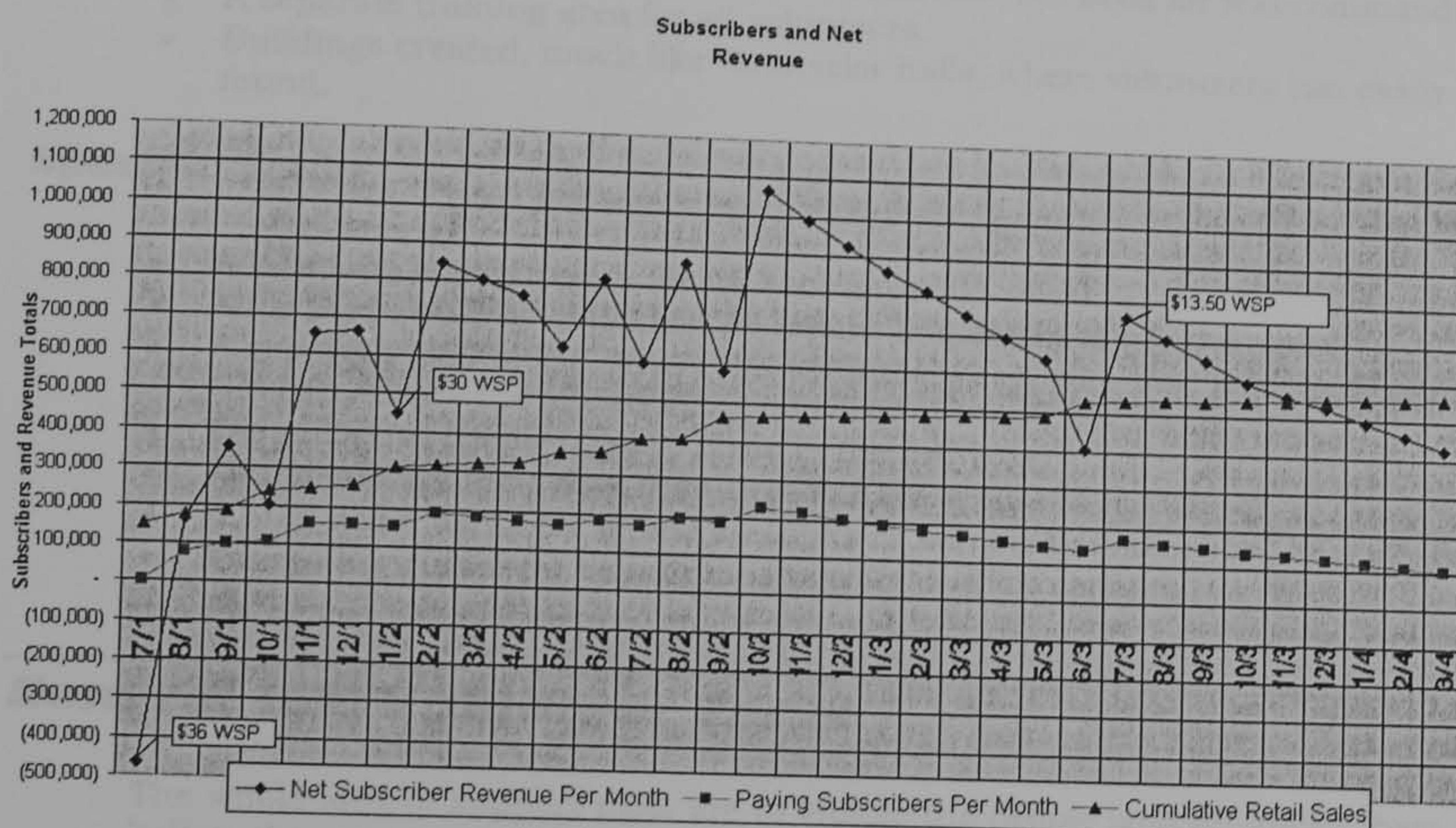
The easiest systems to slip out of the initial release are those involving web integration. They do not impact the core operation of the game, and therefore are easily removed from the initial launch. However, they also have the least impact upon the team's timeline, as the bulk of the work to be completed falls on the web group.

Capital ships, space stations, and formalized cities are the major game elements that could most easily be removed or slipped to a post-launch addition. In the event of timeline issues, these are the systems we would examine first. There are also numerous areas where the design can scale in terms of content. We are already planning to only develop the bare minimum of content required for launch.

Support

Our goal is to develop the tools and software for PO that can be easily adopted and maintained by the live team. The software and tools have to be well documented, professionally developed, and extensible. Finally, design and develop tools and game mechanics that require minimum support (Live) to run.

Our other goal is a minimum 40% contribution margin.



Bandwidth

Our goal is to achieve an average sustained bandwidth of 1.2k/user/sec. From the start our network code will be designed to provide the most efficient throughput without sacrificing game play.

Storage and Hardware Costs

The server farm will be made up of four Dell Quad Machines with one gig of Ram at a cost of ~\$12,000 each (\$48,000 total). Supporting hardware will run ~ \$50,000 (DLT, Raid Drives, UPS, etc.) The supporting hardware can be configured to support two server farms.

Live Team

Our live team will be composed of a producer, six programmers (two for network, one database, two client, and one lead), four designers (three systems/scripting/fiction, and

one lead), and a couple of artists to continually provide additional content or changes. This comes to a total headcount of thirteen people. That is ~108,000 dollars per month burn rate for the PO live team.

Volunteer Tools for Counselors and Companions

General tools

- In game chat system that the volunteer would automatically be logged into when they join the game.
- A flexible graphic interface that would alleviate the need for text commands.
- A separate training area for all volunteers.
- Buildings created, much like Counselor halls, where volunteers can easily be found.

Specific Tools

- A locations configuration file for each separate world.
- A file that would save the last 25 names and object id's of the caller.
- A separate "cell block" where the volunteers can take troublemakers until a GM can survey the situation.
- The ability to teleport to any place in the universe.
- All the current tools that they already have access to in UO.
- A drop down menu for all the locations or sub commands.
- A targeting cursor for transferring players or their corpses.

General IGM/Volunteer Tools

The whole idea of support tools are to enable the people who are using them to better the experience of the customer in the quickest and easiest manner possible. The more tools we have the better off we are. The easier to use these tools the more we will be able to affect the customers in the game in a positive manner.

1. **Easy to use interface:** The tools all come from a graphical interface that is easy to navigate and has the ability to grow with the addition of new tool sets.
2. **Scaleable:** The tool interface needs to be able to add or remove skill sets per the level of IGM or volunteer using them. Either gray out the buttons or have a different front end for each group.
3. **Uniformity:** Make the tools for all the groups utilize the same sets, therefore training will be more uniformed.
4. **Support Minded:** Create the tools to be support minded and not Development/Editor minded.

Specific Tools

1. **Teleport:** The ability to go any where in the universe in an instant.
2. **Communicate:** The ability to speak to anyone by knowing his or her name or Id #.

3. **Create:** To create any object and place them in the world to be used by the players.
4. **Spawn:** To be able to Spawn any given number of Monsters/Creatures/Enemies and be able to set them to an attach pattern or style.
5. **Region:** To mark a region to do specific things (i.e. No Guards or in the case of PO a no Confed zone).
6. **Skill/Stat/Item manipulation:** We need to have the ability to changes the stats and skills of player characters as well as monsters, creatures and vehicles.
7. **Become:** This will allow the GM to change into any body available and maybe even some that are not.
8. **Goto:** We have to have the ability to go to any player in the game by name or ID#.
9. **Player/Item/Monster Info:** Full stats and skill information read outs on all monsters, items, and players.
10. **Script/Integer Manipulation:** If scripts and integers are being used to enhance or change items or players then we need the ability to change these.
11. **World building:** We are going to need the ability to build structures and areas in the world for the use of events and quests.
12. **Player Database Access:** Quick access to the player database.
13. **Freeze/Squelch:** another tool used not only for events but to handle customers that need to be talked to.
14. **Delete:** We need to be able to delete objects in the world from a single NPC to an entire Starship.
15. **Slay:** This will give the IGM the ability to slay any NPC or player character.
16. **Duplication of created creatures or items:** After creating a special creature for an event this tool would allow the IGM to then make duplicates of the one creature with out having to create each one individually.
17. **Trigger system:** Allowing the IGM to drop switches and levers and then use them singly or in ordered or random groupings to complete a common goal (Opening a door to a secret room or unlocking a special chest).
18. **Quest Variable Script:** This script would be placed on a player once they have spoken to a quest oracle or completed a set task for the event.
19. **Information Medium:** The IGM will need to be able to set up Oracles, Criers, Message Boxes, Surveys, Bulletin boards and books with out the fear of players changing the text or using the item inappropriately to hinder others from completing the quest or event at hand.

Staffing

The budget and ROI can be seen on the following pages.

Key personnel

1. Alan Pavlish, Executive Producer
2. Rich Vogel, Producer
3. Bryan Walker, Producer
4. Cody Loyd, Admin
5. J. Allen Brack, Assistant Producer
6. Raph Koster, Lead Designer
7. Pete Shelus, Lead Programmer
8. Chris Mayer, Lead Server Programmer
9. Andy Sommers, Lead Client Programmer
10. Nick Newhard, Senior Designer
11. Hal Milton, Designer
12. John Moreland, Designer
13. Will Kier, Artist
14. Justin Randall, Network Programmer
15. Francine Hunter, Client Programmer
16. Clark Janes, Client Programmer

To be hired

1. Art director
2. Database programmer
3. Server programmer
4. Space sim designer
5. Resource management designer
6. Artist/Animator (5)
7. Programmer (2)

FISCAL YEAR 2000

	Apr-99	May-99	Jun-99	Jul-99	Aug-99	Sep-99	Oct-99	Nov-99	Dec-99	Jan-00	Feb-00	Mar-00	Q1	Q2	Q3	Q4	FY00
CO																	
Salaries	0	0	0	38,154	58,532	101,050	99,148	105,302	142,685	134,148	134,148	173,935	0	197,737	347,134	442,231	987,101
Overtime Salaries	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bonuses	0	0	0	3,624	4,682	5,686	29,599	30,614	31,679	33,629	33,629	33,816	0	13,992	91,891	101,073	206,956
PTO	0	0	0	763	1,171	2,021	1,983	2,106	2,854	2,683	2,683	3,479	0	3,955	6,943	8,845	19,743
Payroll Tax	0	0	0	3,196	4,836	7,181	7,734	7,913	8,762	13,201	13,058	9,418	0	15,212	24,409	35,676	75,298
Employee Insurance	0	0	0	1,375	2,750	4,125	4,950	5,225	5,775	6,875	6,875	7,150	0	8,250	15,950	20,900	45,100
Pension/401k match	0	0	0	763	1,171	2,021	1,983	2,106	2,854	2,683	2,683	3,479	0	3,955	6,943	8,845	19,743
Insurance-General	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HEADCOUNT FIXED	0	0	0	47,875	73,143	122,084	145,396	153,266	194,608	193,218	193,075	231,277	0	243,101	493,271	617,570	1,353,941
Employee Benefits	0	0	0	100	200	300	360	380	420	500	500	520	0	600	1,160	1,520	3,280
Travel & Lodging	0	0	0	6,750	2,640	300	4,100	875	5,100	2,000	6,100	3,300	0	9,690	10,075	11,400	31,165
Business Meals	0	0	0	200	200	200	200	200	200	200	200	200	0	600	600	600	1,800
Crunch Meals	0	0	0	0	0	0	1,080	0	0	0	0	0	0	0	1,080	0	1,080
Personnel Recruiting (20% recruiter fee)	0	0	0	0	0	0	30,000	15,000	22,000	47,000	0	13,000	0	0	67,000	60,000	127,000
Relocation Expense (see bottom of sheet)	0	0	0	9,600	6,000	9,375	15,875	16,625	16,750	21,000	17,250	15,000	0	24,975	49,250	53,250	127,475
Business Entertainment	0	0	0	200	200	200	200	200	200	200	200	200	0	600	600	600	1,800
Offsite Meetings	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Temporary Help (In-House)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Temporary Help (Agency)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Trade Shows	0	0	0	0	0	2,400	0	0	0	0	0	4,050	0	2,400	0	4,050	6,450
Training & Conferences	0	0	0	0	0	0	1,000	0	400	400	400	400	0	0	1,400	1,200	2,600
Corporate Training (HR Only)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HEADCOUNT DISCRETIONARY	0	0	0	16,850	9,240	12,775	52,815	33,280	45,070	71,300	24,550	36,670	0	38,865	131,165	132,620	302,650
Office Supplies	0	0	0	175	350	989	3,230	665	1,935	875	1,475	1,510	0	1,514	5,830	3,860	11,204
Dues, Fees & Subscriptions	0	0	0	0	0	50	50	0	0	50	0	0	0	50	50	50	150
Postage/Business Reply	0	0	0	25	50	75	90	95	105	125	125	130	0	150	290	380	820
Overnight Mail/Freight	0	0	0	200	200	200	200	200	200	200	200	200	0	600	600	600	1,800
Research Samples (books, etc...)	0	0	0	200	0	0	200	0	0	200	0	0	0	200	200	200	600
Software Purchases (<\$1,000)	0	0	0	0	0	5,325	2,835	1,100	1,700	5,000	1,700	345	0	5,325	5,635	7,045	18,005
Hardware Accessories(<\$1,000)	0	0	0	0	1,030	500	1,250	1,250	1,250	1,250	1,250	1,250	0	1,530	3,750	3,750	9,030
Depreciation	0	0	0	354	632	1,220	1,648	1,690	1,900	2,621	2,831	2,952	0	2,206	5,238	8,405	15,849
Amortization	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Outside Services - Development	0	0	0	0	0	0	0	0	0	0	50,000	54,600	0	0	0	104,600	104,600
Outside Services - Non Development	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Artist Advance Writeoffs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Artist Advance Reserves	0	0	0	0	0	5,000	27,000	44,500	27,000	27,000	27,000	27,000	0	5,000	98,500	81,000	184,500
Charge-backs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Entertainment Committee (EC Use Only)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Charitable Contributions	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
OTHER EXPENSE	0	0	0	954	2,262	13,359	36,453	49,500	34,140	37,271	84,581	88,037	0	16,575	120,093	209,890	346,558
Equipment Maintenance	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Miscellaneous Taxes	0	0	0	120	120	120	120	120	120	120	120	120	0	360	360	360	1,080
Telephone/Pager	0	0	0	175	350	525	630	665	735	875	875	910	0	1,050	2,030	2,660	5,740
Cell Phone / pager	0	0	0	100	150	150	150	150	150	150	150	150	0	400	450	450	1,300
Equipment Rental	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bandwidth - T1 & T3 lines	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Facilities Expenses	0	0	0	395	620	795	900	935	1,005	1,145	1,145	1,180	0	1,810	2,840	3,470	8,120
FACILITIES	0	0	0	395	620	795	900	935	1,005	1,145	1,145	1,180	0	1,810	2,840	3,470	8,120
TOTAL before O/H allocations	0	0	0	66,074	85,264	149,013	235,564	236,981	274,823	302,934	303,451	357,164	0	300,361	747,369	963,550	2,011,270
O/H ALLOCATIONS	0	0	0	963	1,119	1,882	1,882	1,900	1,882	1,909	2,490	2,466	0	3,964	5,564	6,865	16,493
TOTAL	0	0	0	67,037	86,383	150,895	237,447	238,881	276,705	304,843	305,942	359,630	0	304,315	753,033	970,415	2,027,763

FISCAL YEAR 2001

	Apr-00	May-00	Jun-00	Jul-00	Aug-00	Sep-00	Oct-00	Nov-00	Dec-00	Jan-01	Feb-01	Mar-01	Q1	Q2	Q3	Q4	FY01
Salaries	139,148	144,148	155,394	155,394	155,394	155,394	155,394	155,394	155,394	155,394	155,394	155,394	468,288	505,030	505,030	505,030	1,983,377
Overtime Salaries	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bonuses	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PTO	2,783	2,883	3,700	3,108	3,108	3,108	3,108	3,108	3,108	3,108	3,108	3,108	9,366	43,229	194,529	194,529	432,287
Payroll Tax	9,429	9,845	11,161	10,758	10,127	11,242	10,667	10,667	11,973	15,637	15,637	11,973	30,436	10,101	10,101	10,101	39,669
Employee Insurance	7,150	7,425	7,700	7,700	7,700	7,700	7,700	7,700	7,700	7,700	7,700	7,700	22,275	32,126	33,306	43,246	139,113
Pension/401k match	2,783	2,883	3,700	3,108	3,108	3,108	3,108	3,108	3,108	3,108	3,108	3,108	9,366	10,101	10,101	10,101	39,669
Insurance-General	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HEADCOUNT FIXED	161,293	167,184	211,254	194,477	193,846	235,363	244,819	244,819	244,819	249,789	249,789	249,789	539,731	623,686	776,167	786,107	2,725,690
Employee Benefits	520	540	560	560	560	560	560	560	560	560	560	560	1,620	1,680	1,680	1,680	6,660
Travel & Lodging	5,500	1,000	6,075	1,000	5,500	1,000	5,500	5,500	5,500	4,500	4,500	4,500	12,575	7,500	10,575	4,500	35,150
Business Meals	200	200	200	200	200	200	200	200	200	200	200	200	600	600	600	600	2,400
Crunch Meals	1,000	0	1,000	0	1,000	0	1,000	0	1,000	0	1,000	0	2,000	1,000	2,000	1,000	6,000
Personnel Recruiting (20% recruiter fee)	0	13,000	10,000	0	0	0	0	0	0	0	0	0	23,000	0	0	0	23,000
Relocation Expense (see bottom of sheet)	3,250	6,500	5,750	10,750	2,500	0	0	0	0	0	0	0	15,500	13,250	0	0	28,750
Business Entertainment	50	50	50	50	50	50	50	50	50	50	50	50	150	150	150	150	600
Offsite Meetings	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Temporary Help (In-House)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Temporary Help (Agency)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Trade Shows	0	0	3,350	0	0	2,400	0	0	0	0	0	0	3,350	2,400	0	4,050	9,800
Training & Conferences	400	400	400	400	400	400	400	400	400	400	400	400	1,200	1,200	1,200	1,200	4,800
Corporate Training (HR Only)																	
HEADCOUNT DISCRETIONARY	10,320	21,690	27,385	12,960	10,210	4,810	7,710	1,785	6,710	1,210	6,710	5,260	59,995	27,780	16,205	13,180	117,160
Office Supplies	1,510	1,545	2,780	980	980	980	980	980	980	980	980	980	5,835	2,940	2,940	2,940	14,655
Dues, Fees & Subscriptions	50	0	0	50	0	0	0	50	0	0	0	0	50	50	50	50	200
Postage/Business Reply	130	135	140	140	140	140	140	140	140	140	140	140	405	420	420	420	1,665
Overnight Mail/Freight	200	200	200	200	200	200	200	200	200	200	200	200	600	600	600	600	2,400
Research Samples (books, etc...)	0	0	200	0	0	0	0	0	0	0	0	0	200	200	200	200	800
Software Purchases (<\$1,000)	1,700	3,745	0	0	0	0	0	0	0	0	0	0	5,445	0	0	0	5,445
Hardware Accessories(<\$1,000)	1,250	1,250	1,250	1,250	1,250	1,250	1,250	1,250	1,250	1,250	1,250	1,250	3,750	3,750	3,750	3,750	15,000
Depreciation	2,460	3,094	3,094	4,637	4,637	4,637	4,637	4,637	4,637	4,637	4,637	4,845	8,647	13,910	13,910	14,327	50,794
Amortization	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Outside Services - Development	50,000	50,000	50,000	50,000	5,000	0	0	0	0	15,000	0	10,000	150,000	55,000	0	0	230,000
Outside Services - Non Development	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Artist Advance Writeoffs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Artist Advance Reserves	27,000	27,000	27,000	27,000	27,000	27,000	27,000	27,000	27,000	27,000	27,000	27,000	81,000	81,000	81,000	81,000	324,000
Charge-backs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Entertainment Committee (EC Use Only)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Charitable Contributions	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
OTHER EXPENSE	84,300	85,969	84,864	84,257	39,207	34,407	34,257	34,207	34,407	49,257	34,415	44,615	255,932	157,870	102,870	128,287	644,959
Equipment Maintenance	120	150	150	150	150	150	150	150	150	150	150	150	420	450	450	450	1,770
Miscellaneous Taxes	910	945	980	980	980	980	980	980	980	980	980	980	2,835	2,940	2,940	2,940	11,655
Telephone/Pager	150	150	150	150	150	150	150	150	150	300	300	300	450	450	450	900	2,250
Cell Phone / pager	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Equipment Rental	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bandwidth - T1 & T3 lines	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Facilities Expenses	1,180	1,245	1,280	1,280	1,280	1,280	1,280	1,280	1,280	1,430	1,430	1,430	3,705	3,840	3,840	4,290	15,675
FACILITIES																	
TOTAL before O/H allocations	257,693	277,088	324,582	292,974	244,542	275,660	288,066	282,091	328,925	301,686	292,344	337,833	859,363	813,176	899,082	931,863	3,503,484
O/H ALLOCATIONS	6,500	6,760	7,000	7,000	7,000	7,000	7,000	7,000	7,000	7,000	7,000	7,000	20,250	21,000	21,000	21,000	83,250
TOTAL	264,193	283,838	331,582	299,974	251,542	282,660	295,066	289,091	335,925	308,686	299,344	344,833	879,613	834,176	920,082	952,863	3,586,734

Origin Sys
FY00 Annual Plan
Privateer Online

FISCAL YEAR 2002

	Apr-01	May-01	Jun-01	Jul-01	Aug-01	Sep-01	Oct-01	Nov-01	Dec-01	Jan-02	Feb-02	Mar-02	Q1	Q2	Q3	Q4	FY02	TOTAL	%
Salaries	155,394	155,394	184,242	163,163	0	0	0	0	0	0	0	0	505,030	163,163	0	0	668,193	3,638,671	53.31%
Overtime Salaries	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Bonuses	0	0	0	0	0	0	0	0	0	0	0	0	0	4,848	0	0	4,848	644,091	9.44%
PTO	3,108	3,108	3,885	3,263	0	0	0	0	0	0	0	0	10,101	3,263	0	0	13,364	72,776	1.07%
Payroll Tax	9,726	9,726	11,033	10,058	0	0	0	0	0	0	0	0	30,485	10,058	0	0	40,543	254,954	3.73%
Employee Insurance	7,700	7,700	7,700	7,700	0	0	0	0	0	0	0	0	23,100	7,700	0	0	30,800	167,475	2.45%
Pension/401k match	3,108	3,108	3,885	3,263	0	0	0	0	0	0	0	0	10,101	3,263	0	0	13,364	72,776	1.07%
Insurance-General	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
HEADCOUNT FIXED	179,036	179,036	220,745	192,235	0	0	0	0	0	0	0	0	578,817	192,235	0	0	771,112	4,850,743	71.05%
Employee Benefits	560	560	560	560	0	0	0	0	0	0	0	0	1,680	560	0	0	2,240	12,180	0.18%
Travel & Lodging	4,500	0	8,875	0	0	0	0	0	0	0	0	0	13,375	0	0	0	13,375	79,690	1.17%
Business Meals	200	200	200	200	200	200	200	200	200	200	200	200	600	600	600	600	2,400	6,600	0.10%
Crunch Meals	0	4,000	4,000	4,000	0	0	0	0	0	0	0	0	8,000	4,000	0	0	12,000	19,080	0.28%
Personnel Recruiting (20% recruiter fee)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150,000	2.20%
Relocation Expense (see bottom of sheet)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	156,225	2.29%
Business Entertainment	50	50	50	50	0	0	0	0	0	0	0	0	150	50	0	0	200	2,600	0.04%
Offsite Meetings	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Temporary Help (In-House)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Temporary Help (Agency)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Trade Shows	0	0	3,350	0	0	2,400	0	0	0	0	0	0	3,350	2,400	0	0	5,750	22,000	0.32%
Training & Conferences	400	400	400	400	0	0	0	0	0	0	0	0	1,200	400	0	0	1,600	9,000	0.13%
Corporate Training (HR Only)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
HEADCOUNT DISCRETIONARY	5,710	5,210	17,435	5,210	200	2,600	200	200	200	200	200	200	28,355	8,010	600	600	37,565	457,375	6.70%
Office Supplies	980	980	980	980	0	0	0	0	0	0	0	0	2,940	980	0	0	3,920	29,779	0.44%
Dues, Fees & Subscriptions	50	0	0	50	0	0	0	0	0	0	0	0	50	0	0	0	100	450	0.01%
Postage/Business Reply	140	140	140	140	0	0	0	0	0	0	0	0	420	140	0	0	560	3,045	0.04%
Overnight Mail/Freight	200	200	200	200	0	0	0	0	0	0	0	0	600	200	0	0	800	5,000	0.07%
Research Samples (books, etc...)	0	0	200	0	0	0	0	0	0	0	0	0	200	0	0	0	200	1,600	0.02%
Software Purchases (<\$1,000)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23,450	0.34%
Hardware Accessories (<\$1,000)	1,250	1,250	1,250	1,250	0	0	0	0	0	0	0	0	3,750	1,250	0	0	5,000	29,030	0.43%
Depreciation	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	14,535	14,535	14,535	14,535	58,141	124,784	1.83%
Amortization	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Outside Services - Development	15,000	10,000	10,000	0	0	0	0	0	0	0	0	0	35,000	0	0	0	35,000	369,600	5.41%
Outside Services - Non Development	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Artist Advance Writeoffs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Artist Advance Reserves	97,000	27,000	27,000	114,500	0	0	0	0	0	0	0	0	151,000	114,500	0	0	265,500	774,000	11.34%
Charge-backs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Entertainment Committee (EC Use Only)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Charitable Contributions	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
OTHER EXPENSE	119,465	44,415	44,615	121,965	4,845	4,845	4,845	4,845	4,845	4,845	4,845	4,845	208,495	131,655	14,535	14,535	369,221	1,360,736	19.93%
Equipment Maintenance	150	150	150	150	0	0	0	0	0	0	0	0	450	150	0	0	600	3,450	0.05%
Miscellaneous Taxes	980	980	980	980	0	0	0	0	0	0	0	0	2,940	980	0	0	3,920	21,315	0.31%
Telephone/Pager	300	300	300	300	0	0	0	0	0	0	0	0	900	300	0	0	1,200	4,750	0.07%
Cell Phone / pager	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Equipment Rental	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Bandwidth - T1 & T3 lines	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00%
Facilities Expenses	1,430	1,430	1,430	1,430	0	0	0	0	0	0	0	0	4,290	1,430	0	0	5,720	29,515	0.43%
FACILITIES	1,430	1,430	1,430	1,430	0	0	0	0	0	0	0	0	4,290	1,430	0	0	5,720	29,515	0.43%
TOTAL before O/H allocations	305,641	230,091	284,225	326,900	5,045	5,045	5,045	5,045	5,045	5,045	5,045	5,045	819,957	333,390	15,135	15,135	1,183,617	6,596,371	98.13%
O/H ALLOCATIONS	7,000	7,000	7,000	7,000	0	0	0	0	0	0	0	0	21,000	7,000	0	0	28,000	127,743	1.87%
TOTAL	312,641	237,091	291,225	327,900	5,045	5,045	5,045	5,045	5,045	5,045	5,045	5,045	840,957	340,390	15,135	15,135	1,211,617	6,825,114	100.00%

EA Confidential

	A	B	C	D	E	F	G
1	Title	Wing Commander Online: Privateer					
2	Platform	PC/Online					
3	Exec. Producer	Alan Pavlish					
4	Duration	24 Months					
5	Master Date	30-Jun-01					
6	Producer	Richard Vogel					
7							
8	ADVANCES	AMOUNT					
9	Licensor (Simis)	\$796,000					
10	Developer	\$0					
11	Music	\$0					
12	Other	\$0					
13	SUBTOTAL	\$796,000					
14							
15	Business Variables			Wholesale Price:			
16	WW Sales Forecast:	569,000		1-6 Months	7-24 Months	25+ Months	
17	Return %	13%	US	\$ 36.00	\$ 30.00	\$ 13.50	
18	Sales and Marketing:	8%	UK	\$ 36.00	\$ 30.00	\$ 13.50	
19	Cost of Goods:	\$5.82	France	\$ 36.00	\$ 30.00	\$ 13.50	
20			Germany	\$ 36.00	\$ 30.00	\$ 13.50	
21			Japan	\$ 36.00	\$ 30.00	\$ 13.50	
22			Korea	\$ 36.00	\$ 30.00	\$ 13.50	
23	Output Summary						
24	Net Retail Sales:	\$15,998,430					
25	Net Units	495,030					
26	Total Retail CM:	\$13,683,197	76.78%				
27	Total Costs:	\$9,253,671					
28	Net Retail Profit:	\$4,429,526					
29	Break Even (Retail Units):	334,779					
30	Subscription Net Revenue:	\$ 16,062,769					
31							\$150,000
32	QA - Functionality						\$0
33	QA - Compatibility						\$6,272,282
34	Product Development						\$6,422,282
35	DEVELOPMENT COST SUBTOTAL:						
36							
37	LOCALIZATION COSTS:						
38	Total Words	140,000					
39	Text	125,000					
40	Audio	15,000					
41	Translation Cost per Word	\$ 0.26					
42	Editing Cost per Word	\$ 0.05					
43	Recording Studio per Day	\$ 1,100.00					
44	Voice Actor Fees per Day	\$ 600.00					
45	Number of Voice Actors	3					
46		Translation	Editing	TEXT	AUDIO	IMPLEM.	TOTAL
47	German	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
48	French	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
49	Italian	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
50	Spanish	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
51	Japanese	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
52	Korean	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
53	Taiwanese	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
54	Chinese	\$ 0.26	\$ 0.05	\$ 43,400	\$ 5,100	\$ -	\$ 48,500
55	LOCALIZATION COST SUBTOTAL			\$ 347,200	\$ 40,800	\$ -	\$ 388,000
56	Audio Costs (Soundtrack Musicians):						\$ 192,000
57	NATS/MIS Support Costs						\$ 7,726,282.00
58	DEVELOPMENT COST TOTAL						

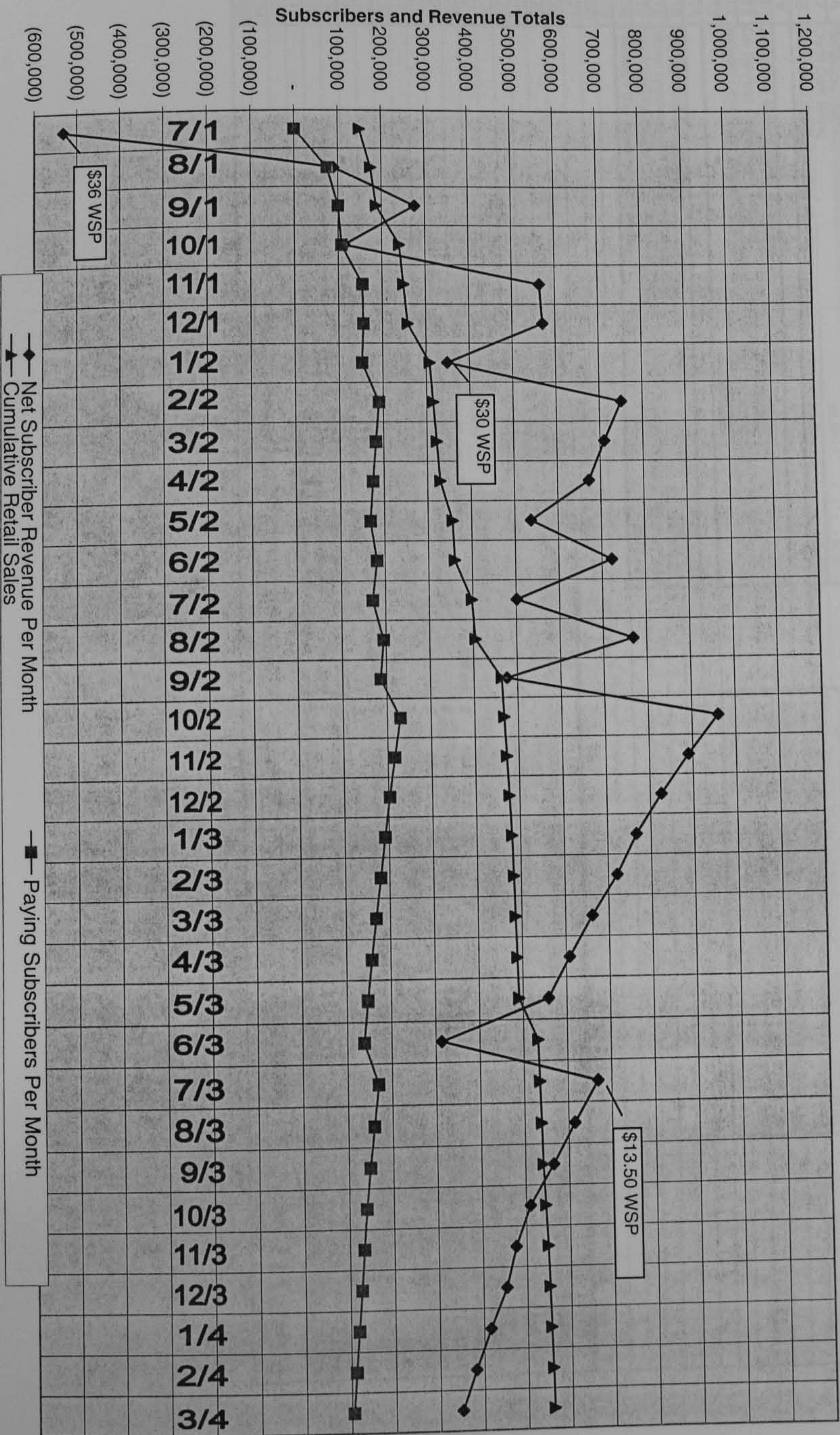
EA Confidential

	H	I	J	K	L	M
1	ROYALTIES & OTHER FIXED UNIT COSTS					
2						
3	ROYALTY CALCULATION PARAMETERS					
4	Licensors (Systans)					
5	Developer (Simis)	\$0.50	per box			
6		2.50%	of Net Sales			
7			minus Advances			
8	ROYALTIES			AMOUNT		
9	Licensors (Systans)	%	WSP	PER UNIT	NET UNITS	EARNED AMT
10	Developer (Simis)	0.00%		\$36	\$0.50	495,030
11	Music	2.50%		\$36	\$0.00	495,030
12	Other	0.00%		\$36	\$0.00	495,030
13	ROYALTY TOTAL	2.50%		\$36	\$0.00	495,030
14				\$0.50		\$247,515
15	UNITS TO RECOUP ADVANCES					
16	Developer (Simis)		Advances	Units		
17			\$796,000	884,444		
18	Cost of Goods (COGS)					
19	Box					Cost @ 569,000
20	Manual			\$ 1.04		\$ 591,760.00
21	Easy Play Guide			\$ 1.50		\$ 853,500.00
22	Reference Card			\$ 0.52		\$ 295,880.00
23	Install Guide			0.14		79660.00
24	SKU Label			\$ -		\$ -
25	2 CDs (\$0.45 each)			\$ 0.45		\$ 256,050.00
26	Jewel Case			\$ 0.90		\$ 230,445.00
27	CD Inlay			\$ 0.45		\$ 256,050.00
28	Back Liner			\$ 0.05		\$ 12,802.50
29	Registration Card			\$ 0.08		\$ 45,520.00
30	Catalog			\$ 0.01		\$ 5,690.00
31	Sticker Seal			\$ 0.10		\$ 56,900.00
32	Corrugated Platform Insert			\$ 0.02		\$ 11,380.00
33	Outside Processing			\$ 0.20		\$ 113,800.00
34	COGS SUBTOTAL			\$ 0.67		\$ 381,230.00
35	Minus COGS SRA	5%		\$ 6.13		\$ 3,190,667.50
36				\$ 0.31		\$ 159,533.38
37	NET COGS			\$ 5.82		\$ 3,031,134.13
38						
39	OTHER RESERVES					
40	Bad Debt (Net Revenue)			Net Sales @	\$36	
41	Co-op Ads (Net Revenue)	0.30%			\$ 0.108	
42	Period Costs (Net COGS)	5%			\$ 1.800	
43	Warranty (Net Revenue)	1.50%			\$ 0.087	
44	Total Other Reserves	1.50%			\$ 0.540	
45		8.30%			\$2.535	
46	Cost of Sales (Per Box)					
47				\$ 8.36	23.22%	
48	Contribution Margin					
49				\$27.64	76.78%	
50	Cost of Sales (Sales Units Goal):					
51						
52	569,000	Units:	\$	4,756,187.07		
53						
54						
55						
56						
57						
58						

	N	O	P	Q	R	S	T	U	V
1									
2	SALES FORECASTS		GROSS UNITS	RETURN %	RETURNS	NET UNITS	WSP	NET SALES	S&M Costs
3	(Frontline)								8%
4	North America								
5	UK		116,100	13.00%	15,093	101,007	\$ 36.00	\$3,636,252	\$290,900
6	France		25,800	13.00%	3,354	22,446	\$ 36.00	\$808,056	\$64,644
7	Germany		20,640	13.00%	2,683	17,957	\$ 36.00	\$646,445	\$51,716
8	Japan		30,960	13.00%	4,025	26,935	\$ 36.00	\$969,667	\$77,573
9	Korea		51,600	13.00%	6,708	44,892	\$ 36.00	\$1,616,112	\$129,289
10	WW FRONTLINE SUBTOTAL		258,000	13.00%	1,677	11,223	\$ 36.00	\$404,028	\$32,322
11	(Charter Edition)				33,540	224,460		\$8,080,560	\$646,445
12	North America		10,000	13.00%	1,300	8,700	\$ 50.00	\$435,000	\$34,800
13									
14	SALES FORECASTS		GROSS UNITS	RETURN %	RETURNS	NET UNITS	WSP	NET SALES	S&M COSTS
15	(Frontline)								
16	North America		123,750	13.00%	16,088	107,663	\$ 30.00	\$3,229,875	\$258,390
17	UK		27,500	13.00%	3,575	23,925	\$ 30.00	\$717,750	\$57,420
18	France		22,000	13.00%	2,860	19,140	\$ 30.00	\$574,200	\$45,936
19	Germany		33,000	13.00%	4,290	28,710	\$ 30.00	\$861,300	\$68,904
20	Japan		55,000	13.00%	7,150	47,850	\$ 30.00	\$1,435,500	\$114,840
21	Korea		13,750	13.00%	1,788	11,963	\$ 30.00	\$358,875	\$28,710
22	WW FRONTLINE SUBTOTAL		275,000		35,750	239,250		\$7,177,500	\$574,200
23									
24	SALES FORECASTS		GROSS UNITS	RETURN %	RETURNS	NET UNITS	WSP	NET SALES	S&M COSTS
25	(Frontline)								
26	North America		11,700	13.00%	1,521	10,179	\$ 13.50	\$137,417	\$10,993
27	UK		2,600	13.00%	338	2,262	\$ 13.50	\$30,537	\$2,442.96
28	France		2,080	13.00%	270	1,810	\$ 13.50	\$24,430	\$1,954.37
29	Germany		3,120	13.00%	406	2,714	\$ 13.50	\$36,644	\$2,932
30	Japan		5,200	13.00%	676	4,524	\$ 13.50	\$61,074	\$4,886
31	Korea		1,300	13.00%	169	1,131	\$ 13.50	\$15,269	\$1,221
32	WW FRONTLINE SUBTOTAL		26,000		3,380	22,620		\$305,370	\$24,430
33	TOTALS		569,000			495,030	\$	15,998,430	\$ 1,279,874
34									
35	Costs Summary:								
36	Project Development:		\$7,726,282						
37	Systrans Licensing:		\$247,515						
38	Sales and Marketing:		\$1,279,874						
39	Total Costs:		\$9,253,671						
40									
41									
42									
43									
44									
45									
46									
47									
48									
49									
50									
51									
52									
53									
54									
55									
56									
57									
58									

Total CM:	\$	13,683,197
UNITS TO BREAK EVEN:		334,779
NET PROFIT:	\$	4,429,526

Subscribers and Net Revenue



	A	B	C	D	E	F	G	H	I	
1	Wing Commander Online: Privateer Online Subscription Revenue									
2	Monthly Fixed Cost Assumptions:				<div>Totals Summary July 02-March 04</div> <div>Gross Subscription Revenue: \$ 50,011,747</div> <div>Net Subscription Revenue: \$ 16,062,769</div> <div>Average Monthly Subscribers: 152,312</div> <div>Average Monthly Cost per Subscriber: 5.69</div> <div>Average Monthly Profit per Subscriber: 2.94</div> <div>Contribution Margin (Paying Accounts): 41.77%</div> <div>Contribution Margin (All Accounts): 31.70%</div> <div>FY02 Net Profit: 2,580,713</div> <div>FY03 Net Profit: 8,191,268</div> <div>FY04 Net Profit: 5,601,414</div>					
3	Monthly Subscriber Fee	\$9.95	% of Monthly Fee							
4	Bandwidth (9600 bps)	\$2.50	25.13%							
5	Customer Support	\$1.00	10.05%							
6	Portal	\$1.00	10.00%							
7	Credit Card Billing	\$0.50	5.03%							
8	Fixed Cost per Subscriber Month	\$5.00	50.20%							
9										
10	Additional Monthly Cost Assumptions:									
11	QA Services (Assuming 3 Testers)	\$25,000					\$ 2,580,713			
12	Network Operations (Assuming 3 Titles)	\$24,000					\$ 8,191,268			
13							\$ 5,601,414			
14										
15	Startup Cost Assumptions:									
16	Server Hardware and Support Startup Costs:		\$1,095,000.00							
17	Live Team Hardware Costs:		\$147,500							
18	Hardware Depreciation per Annum		\$310,625							
19										
20										
21	Plan: 600K Sell-In, 8% Churn Rate			FY02						
22				Jul						
23	Worldwide Retail Sell-In by Month	300,000	50,000	25,000			25,000	15,000	15,000	
24	Worldwide Total Sell-In	300,000	350,000	375,000			400,000	415,000	430,000	
25	Worldwide Sell-Through	150,000	175,000	187,500			240,000	249,000	258,000	
26	New/Renewed Accounts	75,000	25,000	12,500			52,500	9,000	9,000	
27	Churn Rate	0%	0%	0%			5%	5%	5%	
28	Active Subscribers	0	75,000	100,000			107,500	154,625	155,894	
29	Subscriber Fixed Cost Per Month	\$ 374,625.00	\$ 499,500.00	\$ 561,937.50			\$ 799,200.00	\$ 817,306.88	\$ 823,644.28	
30	Monthly Gross Subscriber Revenue	\$ -	\$ 746,250.00	\$ 995,000.00			\$ 1,069,625.00	\$ 1,538,518.75	\$ 1,551,142.81	
31	Monthly Net Revenue	\$ (531,625.00)	\$ 89,750.00	\$ 276,062.50			\$ 113,425.00	\$ 564,211.88	\$ 570,498.53	
32	Monthly Fee	\$ -	\$ 9.95	\$ 9.95			\$ 9.95	\$ 9.95	\$ 9.95	
33	Fixed Cost per Subscriber Month	\$5.00	\$5.00	\$5.00			\$5.00	\$5.00	\$5.00	
34	Live Team Cost per Month	\$ 108,000.00	\$ 108,000.00	\$ 108,000.00			\$ 108,000.00	\$ 108,000.00	\$ 108,000.00	
35	Total Monthly Cost Per Subscriber	\$ 6.44	\$ 6.08	\$ 5.96			\$ 5.67	\$ 5.66	\$ 5.65	
36	Net Monthly Revenue Per Subscriber	\$ (7.09)	\$ 0.90	\$ 2.45			\$ 0.71	\$ 3.45	\$ 3.46	
37	Contribution Margin (Paying Accounts):	0.00%	38.94%	40.15%			43.02%	43.17%	43.22%	
38	Contribution Margin (All Accounts):	0.00%	9.02%	24.66%			7.12%	34.66%	34.77%	
39	Retail Sell-Through Rate	50.00%	50.00%	50.00%			60.00%	60.00%	60.00%	
40	Subscriber Churn Rate:	0.00%	0.00%	0.00%			5.00%	5.00%	5.00%	

COMING

	A	B	C	P	Q	R	S	T	U
1	Wing Commander Online: Privateer Online Subscription Revenue								
2	Monthly Fixed Cost Assumptions:			% of Monthly Fee					
3	Monthly Subscriber Fee	\$9.95							
4	Bandwidth (9600 bps)	\$2.50	25.13%						
5	Customer Support	\$1.00	10.05%						
6	Portal	\$1.00	10.00%						
7	Credit Card Billing	\$0.50	5.03%						
8	Fixed Cost per Subscriber Month	\$5.00	50.20%						
9									
10	Additional Monthly Cost Assumptions:								
11	QA Services (Assuming 3 Testers)	\$25,000							
12	Network Operations (Assuming 3 Titles)	\$24,000							
13		\$49,000							
14									
15	Startup Cost Assumptions:								
16	Server Hardware and Support Startup Costs:		\$1,095,000.00						
17	Live Team Hardware Costs:		\$147,500						
18	Hardware Depreciation per Annum		\$310,625						
19									
20									
21									
22		Plan: 600K Sell-In, 8% Churn Rate							
23		Worldwide Retail Sell-In by Month							
24		Worldwide Total Sell-In							
25		New/Renewed Accounts							
26		Churn Rate							
27		Active Subscribers							
28		Subscriber Fixed Cost Per Month							
29		Monthly Gross Subscriber Revenue							
30		Monthly Net Revenue							
31		Monthly Fee							
32		Fixed Cost per Subscriber Month							
33		Live Team Cost per Month							
34		Total Monthly Cost Per Subscriber							
35		Net Monthly Revenue Per Subscriber							
36		Contribution Margin (Paying Accounts):							
37		Contribution Margin (All Accounts):							
38									
39		Retail Sell-Through Rate							
40		Subscriber Churn Rate:							

A			B	C	V	W	X	Y	Z	AA
1	Wing Commander Online: Privateer Online Subscription Revenue									
2	Monthly Fixed Cost Assumptions:									
3	Monthly Subscriber Fee		\$9.95	% of Monthly Fee						
4	Bandwidth (9600 bps)		\$2.50	25.13%						
5	Customer Support		\$1.00	10.05%						
6	Portal		\$1.00	10.00%						
7	Credit Card Billing		\$0.50	5.03%						
8	Fixed Cost per Subscriber Month		\$5.00	50.20%						
9										
10	Additional Monthly Cost Assumptions:									
11	QA Services (Assuming 3 Testers)		\$25,000							
12	Network Operations (Assuming 3 Titles)		\$24,000							
13			\$49,000							
14										
15	Startup Cost Assumptions:									
16	Server Hardware and Support Startup Costs:			\$1,095,000.00						
17	Live Team Hardware Costs:			\$147,500						
18	Hardware Depreciation per Annum			\$310,625						
19										
20										
21										
22		Plan: 600K Sell-In, 8% Churn Rate			Jan	Feb	Mar	FY04 Apr	May	Jun
23		Worldwide Retail Sell-In by Month			5,000	3,000	3,000	3,000	3,000	50,000
24		Worldwide Total Sell-In			565,000	568,000	571,000	574,000	577,000	627,000
25		Worldwide Sell-Through			480,250	482,800	485,350	487,900	490,450	532,950
26		New/Renewed Accounts			4,250	2,550	2,550	2,550	2,550	42,500
27		Churn Rate			8%	8%	8%	8%	8%	8%
28		Active Subscribers			190,640	179,639	167,818	156,942	146,937	137,732
29		Subscriber Fixed Cost Per Month			\$ 973,476.65	\$ 910,034.06	\$ 850,987.57	\$ 796,664.79	\$ 746,687.84	\$ 900,259.29
30		Monthly Gross Subscriber Revenue			\$ 1,896,870.18	\$ 1,787,408.07	\$ 1,669,787.92	\$ 1,561,577.39	\$ 1,462,023.70	\$ 1,370,434.30
31		Monthly Net Revenue			\$ 766,393.54	\$ 720,374.00	\$ 661,800.35	\$ 607,912.59	\$ 558,335.86	\$ 313,175.01
32		Monthly Fee			\$ 9.95	\$ 9.95	\$ 9.95	\$ 9.95	\$ 9.95	\$ 9.95
33		Fixed Cost per Subscriber Month			\$5.00	\$5.00	\$5.00	\$5.00	\$5.00	\$5.00
34		Live Team Cost per Month			\$ 108,000.00	\$ 108,000.00	\$ 108,000.00	\$ 108,000.00	\$ 108,000.00	\$ 108,000.00
35		Total Monthly Cost Per Subscriber			\$ 5.55	\$ 5.59	\$ 5.63	\$ 5.67	\$ 5.72	\$ 5.59
36		Net Monthly Revenue Per Subscriber			\$ 3.93	\$ 3.95	\$ 3.88	\$ 3.81	\$ 3.74	\$ 1.74
37		Contribution Margin (Paying Accounts):			44.23%	43.84%	43.43%	42.99%	42.54%	43.78%
38		Contribution Margin (All Accounts):			39.52%	39.74%	39.04%	38.31%	37.54%	17.46%
39		Retail Sell-Through Rate			85.00%	85.00%	85.00%	85.00%	85.00%	85.00%
40		Subscriber Churn Rate:			8.00%	8.00%	8.00%	8.00%	8.00%	8.00%

Garrett (Rich)

EA Confidential

A			B	C	AB	AC	AD	AE	AF	AG
1	Wing Commander Online: Privateer Online Subscription Revenue									
2	Monthly Fixed Cost Assumptions:									
3	Monthly Subscriber Fee		\$9.95	% of Monthly Fee						
4	Bandwidth (9600 bps)		\$2.50	25.13%						
5	Customer Support		\$1.00	10.05%						
6	Portal		\$1.00	10.00%						
7	Credit Card Billing		\$0.50	5.03%						
8	Fixed Cost per Subscriber Month		\$5.00	50.20%						
9										
10	Additional Monthly Cost Assumptions:									
11	QA Services (Assuming 3 Testers)		\$25,000							
12	Network Operations (Assuming 3 Titles)		\$24,000							
13			\$49,000							
14										
15	Startup Cost Assumptions:									
16	Server Hardware and Support Startup Costs:			\$1,095,000.00						
17	Live Team Hardware Costs:			\$147,500						
18	Hardware Depreciation per Annum			\$310,625						
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										
37										
38										
39										
40										

Subs ROI

0/5/99

A			B	C	AH	AI	AJ	AK
1	Wing Commander Online: Privateer Online Subscription Revenue							
2	Monthly Fixed Cost Assumptions:							
3	Monthly Subscriber Fee		\$9.95	% of Monthly Fee				
4	Bandwidth (9600 bps)		\$2.50	25.13%				
5	Customer Support		\$1.00	10.05%				
6	Portal		\$1.00	10.00%				
7	Credit Card Billing		\$0.50	5.03%				
8	Fixed Cost per Subscriber Month		\$5.00	50.20%				
9								
10	Additional Monthly Cost Assumptions:							
11	QA Services (Assuming 3 Testers)		\$25,000					
12	Network Operations (Assuming 3 Titles)		\$24,000					
13			\$49,000					
14								
15	Startup Cost Assumptions:							
16	Server Hardware and Support Startup Costs:			\$1,095,000.00				
17	Live Team Hardware Costs:			\$147,500				
18	Hardware Depreciation per Annum			\$310,625				
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								
33								
34								
35								
36								
37								
38								
39								
40								

Pile of Thought & Table of Contents

The "Pile of Thought" is our term for the more detailed design documentation and work that has gone into shaping the high-level DDR document you just read. Developing the Pile of Thought fully is normally an endeavor that takes many months. We have focused our attention on finishing those documents most critical for shaping the DDR proper, as well as those documents most necessary for proceeding with a Technical Design Review phase. However, because of the timeline limitations, there are not yet Pile of Thought documents for every system in the game, nor are these documents all considered final as of this writing.

The Pile of Thought work done thus far on *WCO:P* resides on our internal website at <http://figmund/devteams/PO/po-index.html>. However, this site requires a password, for security reasons. For your convenience, we have printed out copies of the Pile of Thought documents completed at this time and inserted them into this binder for those people who are reviewing the game design proper (as opposed to just reading the DDR document). Please refer to this section if you are looking for further detail on particular systems within the game design.

PILE OF THOUGHT & TABLE OF CONTENTS.....	1
APPROACHES TO DESIGN.....	3
WILD SPACE - TAME SPACE - SAFE SPACE	5
MISSION BOARD	11
CLONING AND INSURANCE	15
GROUND COMBAT	19
JUMP POINTS, JUMP DRIVE AND JUMPING.....	25
SHIP CUSTOMIZATION.....	31
SHIPS - WEAPONS.....	37
SHIPS - DEFENSES - SHIELDS	47
SHIPS - DEFENSES - ARMOR	53
SHIPS - DEFENSES - MISSILE DEFENSES	59
PHYSICS, MANEUVERING, POWER - FIGHTERS.....	69
GENERATED MISSIONS GROUND AND SPACE	76
CAPITAL SHIPS.....	84

GROUPING	100
CRAFTING.....	103
THE STOCK MARKET.....	109
THE COMMODITIES EXCHANGE.....	114
COMPANIES	120
COMPANIES - TOOLS	125
AUTOMATED INSTALLATIONS.....	131
OWNERSHIP.....	136
RESOURCES	141
BASE STATISTICS AND ADVANCEMENT	144
AVATAR CUSTOMIZATION	147
ADVANCEMENT: STARTING ACTIVITIES.....	151
ADVANCEMENT: PROFESSION TREE	154
ADVANCEMENT: APPRENTICESHIP	158
REPUTATION.....	161
LEADER BOARDS	165
PLANET SPAWNING.....	167
FLORA AND FAUNA GENERATION.....	171
PLAYER HOUSING	175
CITIES.....	181
CAMERAS, CONTROLS, AND INTERFACE.....	186
BACKSTORY	193
OWO MAIL.....	195
RENDERING REQUIREMENTS AND SPECIFICATIONS	200
ANCILLARY REVENUE	203
DEMO	208
TOP MARKETING HOOKS	209
PLANS FOR EXTENSIBILITY (BRAINSTORM)	210

Approaches to Design

The following tenets guided our design process as we worked on the documents herein.

Plan for the Internet medium:

- Assume delayed responses to everything – don't make the system too reliant on time-critical responses.
- Design systems to have fast and significant reward cycles, but be aware that every system you design should attempt to give 50+ hours of game play to each player.
- Design systems to be easily expandable and editable on the server side.
- Avoid static data.

Plan for low backend costs:

- Avoid game design approaches that rely on human administrative intervention. *EverQuest* incurred a huge cost by requiring manual approval of names and guilds.
- Don't design stuff that requires high bandwidth.
- Minimize storage on server-side objects. But don't trust the client either.

Plan for what players regard as fun:

- Don't get hung up on what you personally view as fun. One of the most popular features in *Ultima Online* was fishing. There are many types of players out there.
- Community features offer more value than the game aspects do (distressing to hear as a game designer, I know). Time spent on adding a wedding dress to the game is more valuable than time spent adding a new ship type.
- Design for variety. Players will be living here. Having only one type of laser grows boring.
- Design for player empowerment. Better to let players lay out their floor plan than for us to provide stock ones.
- Strive to provide a sense of ownership.

Plan for simplicity:

- Make it easy to use.
- Make it easy to develop further.
- Make it easy to understand.

Places to go to get more info on this stuff:

- Laws of Online World Design: <http://mud.sig.net/raph/gaming/laws.html>
- Journal of MUD Research: <http://journal.tinymush.org>
- MUD-Dev discussion list: <http://www.kanga.nu/lists/listinfo/mud-dev/>
- Richard Bartle's page: <http://www.mud.co.uk/richard/>
- Habitat papers: <http://www.communities.com/people/crock/habitat.html>
- Julian Dibbell's essays: http://www.levity.com/julian/index_noframes.html
- Lydia Leong's MUD Resource Collection: <http://www.godlike.com/muds/>
- Amy Jo Kim's Naima page: <http://www.naima.com>

Wild Space - Tame Space - Safe Space

Overview

In Privateer space can be a very dangerous place. In many cases, space can be too dangerous for the starting or casual player. We need to provide safe, tame, and completely wild areas for players to explore dependent on their needs/desires. Wild, Tame, and Safe space systems are to frame PVP and PVE (player vs. environment/AI) within areas that can be easily recognized and avoided by new players or non-combat oriented players.

Safe space represents our newbie start systems and is completely safe (no weapon or collision damage). Tame space represents our intermediate systems and encourages PVE conflict, but PVP is restricted to Bounty Hunters and Pirates. Wild Space represents the lawless systems, Dead Space, and other "advanced" areas. PVP & PVE conflict is present in Wild Space.

Vision and goals

Goals

- ☐ Provide newbies safe environments to explore the initial skill sets they wish to pursue without the threat of being harmed in any way.
- ☐ Provide intermediate spaces with limited PVE conflict.
- ☐ Provide wild spaces for free PVP/PVE conflict.
- ☐ Provide systems for monitoring the activity within Wild and Tame Space.

Vision Doc

- ☐ Must offer a strong support system so that players feel welcomed into the game and so the game atmosphere is friendly
- ☐ Should empower players to take action against those who are obnoxious
- ☐ Should expose new strategies throughout the game to the player
- ☐ Should make the player feel smart
- ☐ Should never make the player feel ripped off by the game mechanics

Walkthrough

To be covered:

- Definition/Function of Safe Space
- Devices, both fictional and practical, to keep Safe space Safe
- Definition/Function of Tame Space
- Sensor Disruptive Qualities [SDQs]
- Devices, both fictional and practical to moderate Tame Space
- Definition/Function of Wild Space
- The Taming of Wild Space
- Differentiating Safe/Tame/Wild on the Nav map
- Tracking System Activity

With each section there will be a small table at the beginning that displays who can attack whom in each section. Under AI in tame and wild space there are “??”, These are for future discussion of any AI rivalries the player may land in the middle of.

		V I C T I M			
A T T A C K E R	SAFE SPACE				
		Bounty Hunter	Pirate/ PK	Neutral/ Non-PK	AI
	Bounty Hunter	NO	NO	NO	N/A
	Pirate/ PK	NO	NO	NO	N/A
	Neutral/ Non-PK	NO	NO	NO	N/A
	AI	N/A	N/A	N/A	N/A

Systems design

Safe Space/Ground

Safe space is exactly that. All Newbie start locations are locked down safe space. This includes both Ground and Space. Weapons do not function, and collisions deal no damage. This is where payers get their feet wet with the numerous and intricate systems

and paths available for them.

No newbie should have to look over their shoulder for an evil player coming to kill them. They have enough to worry about already. We want to ensure that Safe Space offers a large area for newbies to practice their piloting skills, get used to their starting craft, perform simple missions, and a host of other activities.

Safe Ground areas are terrestrial zones (newbie start city, etc.) that are also devoid of any form of physical conflict.

Smuggling of contraband is also highly restricted within safe space. Any ship attempting to smuggle illegal goods will first receive a warning when they attempt to plot a jump into a Safe system. If they proceed into Safe space anyway, they must then avoid getting within x range of any Confederation patrol, and then pass the die roll of “customs” when they attempt to land on a station or planet. Not a task for the amateur.

Safe Space Devices

Weapon Lock-Down A.K.A Safety.Net

All ships, and planet side players, will not be able to utilize any offensive weaponry. The fiction around this will involve a Confed variable transponder signal that “locks down” all weapons. Although it is a questionable device fictionally, some form of lock-down is necessary. If any form of active weapon is allowed into safe space, newbies will die. No penalty (guards, fines, etc.) will change this. Therefore all aggressive potential must be eliminated.

Players will forgive and forget this inconvenience to random violence with the knowledge that their guns will rain electric death on foes elsewhere.

Here’s the fiction for the players:

All prospective pioneers embarking on the journey to rebuild the Confederation are considered to be good citizens tried and true. Because accidents happen, an organization was formed called Safety.Net to protect these pioneers against accidents. Utilizing **Jump Beacon Data Nodes** to transmit the carrier signal to all craft within Confederation controlled space; instances of celestial friendly fire have been greatly reduced. Safety.Net is able to ID and authenticate weapons fire against targets. All citizens’ shipboard weaponry will not fire in Safe Space systems unless they are configured for training. In tame space you may fire

your weapons, but only at hostile (non-Safety.Net member) targets. Safety.Net members ship signatures will prevent your craft from “accidentally” firing energy or ballistic weapons at them. In wild space there is not any Safety.Net carrier signal and weapons are free.

Collision Off or Traffic Control

A possible solution for “ramming kills” within safe space could be as simple as ignoring any collision damage caused within safe space. The player would still be bumped, and the ship reacts to the physics of impact, but no internal or external damage would be recorded.

V I C T I M		TAME SPACE			
A T T A C K E R		Bounty Hunter	Pirate/ PK	Neutral/ Non-PK	AI
	Bounty Hunter	YES	YES	NO	YES
	Pirate/ PK	YES	YES	NO	YES
	Neutral/ Non-PK	NO	NO	NO	YES
	AI	YES	YES	YES	??

Or we could implement a “remote space traffic control system.” This system prevents any ship from getting within **X** meters of another.

Either way a collision solution is TBD.

Tame Space/Ground

Tame space is an area designed for players looking for more risk than Safe Space, but still have a reasonable level of comfort. Basically, these are systems with the lure of environmental threats (AI Pilots, etc.).

These are the threshold systems around newbie start systems. They represent a greater challenge to the player as they endeavor to pursue their goals, but have devices in place to reduce and/or discourage PVP conflict.

SDQ Phenomena

Within Tame Safe systems there is the chance of celestial phenomena that contain threats. Special Asteroid Fields, Nebulae, and other celestial features with SDQs (sensor disruptive qualities) prohibit Confed from turning These are the happy hunting grounds for players, who are “helping” Confed and the Corporations to “lock down” these systems. Of course, they aren’t, but it is a nice way to frame the conflict.

SDQ celestial phenomena will also house valuable/rare resources. Any system with an SDQ can never be rendered completely “Safe” Space. This keeps the game from becoming a consequence free environment. The highest level of safety a system with an SDQ can attain is *Tame*. Not all SDQs affect individual shipboard sensors, only the long-range sensor/scanners used by the Confederation. This is another fictional device.

There are two modes of thought regarding the handling of PVP within Tame Space. The first is to simply not allow it, like Safe Space. This means the player must find a Wild Space system in order to engage in PVP activity.

Tame ground represents an area where there is dangerous wild life on the planet, but ground defenses have been established to protect citizens within certain limits. These ground defenses could be as elaborate as a turret sentry system, or as simple as a wall. Safety.Net measures have been implemented here, but Bounty Hunters and Pirates/PKs can duke it out old west style with the innocent onlookers safely shielded from any harm.

Tame Space Moderation**Confederation Patrols**

In an active tame system (players present) there will be one or more confederation patrols moving about a system. They will skip drive from celestial feature to celestial feature hunting hooligans.

A smuggler with illicit cargo or wanted criminal with a black record will want to stay outside of x range of these patrols or they face a die roll that will determine if the patrol notices them. Just like when you ran that Red Light after having two beers and then notice the cop on the corner...you cross your fingers hard...

In the case of a murderer/pirate: If a player loses the roll against the percentage they will be greeted by the patrol. 1) This Patrol will communicate to the player that they will be escorted to the nearest jump point. 2) Upon arriving at the jump point they have x time to exit the system or the patrol will open fire. X must be an adequate amount of time to leave the system. If the player is already at a jump point then skip 1.

If the player did meet a patrol and managed to leave the system he/she will be flagged for x amount of time before Spawn_Patrol itself is reset. This means that if they try to just jump back into the system there will be a patrol spawn waiting for them to say, "Hey, didn't we tell you to leave??!"

A player who has illegal contraband, and is then stopped by a patrol, still has a chance to escape. The Patrol will "Scan" the player's cargo for contraband that is a %check. This check for scan success or failure is modified by any cargo modifications/upgrades the player installs (cloaking, shielding, camouflage, etc.).

If the scan is "successful" then the player will have to dump their cargo in x time or face the consequences. If the player decides not to dump their cargo, Confed will open fire. If the player does dump the cargo, Confed will then switch to steps 1 and 2 (as if they were a murderer/criminal).

Even if Confed opens fire, the Player can still try to make a run for the jump point. Odds are very few people will be able to pull this off. This is as it should be. Only a serious long-term player will be able to have a ship that can move fast enough/be tough enough to escape a Confed Mil Spec ship patrol. Patrols will "follow" them through their skips until they either destroy the player or the player manages to jump.

V I C T I M					
A T T A C K E R	WILD SPACE				
		Bounty Hunter	Pirate/ PK	Neutral/ Non-PK	AI
	Bounty Hunter	YES	YES	YES	YES
	Pirate/ PK	YES	YES	YES	YES
	Neutral/ Non-PK	YES	YES	YES	YES
	AI	YES	YES	YES	??

Wild Space/Ground

Wild Space is the danger zone, an unmediated system of PVP and PVE conflict. There is no law here. Wild Space will of course become more and more dangerous (from a PVE perspective) the further away from a newbie start zone you are. By more dangerous, I mean, that not only will violent players be present, but also the number,

type, and frequency of AI aggressors will scale.

Wild Ground is similarly dangerous. No ground defenses have been established and the local fauna and enemy player raiding/scavenger parties can proceed unchecked.

This is where the conflict hounds come to do business. A player uninterested in this type of game play need never experience it...unless they discover a new system they wish to colonize and civilize.

There are still reputation hits in Wild Space...these are covered more thoroughly in the Reputation doc.

The Taming of Wild Space

With the exception of systems within the Dead Zone (a sector wide DMZ between Human, Kilrathi, and Firekkan space), players will have the opportunity to civilize wild spaces.

Wild Spaces can become civilized (tame or safe) as players colonize the systems planets, asteroids, and/or moons and develop a population large enough to qualify for Confederation support.

Initially this support may be the Tame Safe system patrols. The players may enhance this with a wide array of celestial, orbital, and terrestrial defenses. When the colonization process reaches certain tolerances, the system can be converted to safe space and after that a potential newbie start area (based upon the condition of your system and the surrounding systems). This is a huge game reward that is covered more thoroughly in the Cities and Housing documentation.

However, any system that contains an SDQ can never rise above TAME.

Exploring the universe, taming a wild system, then rendering it an official part of the Confederation, and then making it a hub of new exploration...Holy crap, this is one of the coolest long term reward systems in the history of online gaming.

Differentiating Safe/Tame/Wild Space On the Nav Map

On the player's Quadrant/Sector Nav map Safe/Tame/Wild systems will be represented by the colors Green (or Blue to stick to the WC paradigm)/Yellow/Red respectively.

Before you enter a Wild Space system or tame system the player will be alerted as to the danger of the system (see Tracking System Activity). Upon entry of a Wild Space system the player will again be alerted as to the danger of their current location.

Tracking System Activity

Dependent on a player's DataNode access they will be able to gain varying levels of resolution regarding the activity within a system.

This will help a player determine whether or not they wish to risk entering the system. Examples of information available:

- Player kills reported in the system over x time period
- Presence/Type/Frequency/Composition of enemies
- Presence/Frequency/Composition of Confed Patrols
- Confed Patrol reports (Number of murders, number of Pirates, etc.)

The potential exists for players to have technical upgrades to track energy signatures entering or exiting a system. Other system info options include recommended equipment/certification levels for survival.

In short, we must provide a wide variety of system tracking that act as both player info sources, and tactical tools. A good Pirate will always have a fairly solid idea of what they are in for when they enter a zone.

However no specific information about the current enemies within an SDQ will be available. This keeps these areas fictionally separate from the long range scans possible within normal space and practically provides players with surprises within celestial hunting grounds.

Data Nodes and the information they carry will be more thoroughly examined in **Space - Jump Beacons/Data Nodes.**

Interface design

TBD.

Tasks

TBD

To be addressed

TBD

- Basic Look and Feel of Board
- Data Fields / Parameters / Filters
- Rules Wizard
- How to Post a Mission
- How to extend/kill on/accept a mission
- Post Delay mechanisms
- Ownership delay mechanisms
- Premium Account Perks

System design

Look and Feel

The mission board will be a very clean, labeled view that provides an abstract and all necessary field info.

Mission Board

Overview

The Mission Board is where players post, bid for, and accepts missions. Our Mission Generator also posts to this board. This is a highly player customizable board that allows them to filter/sort for their personal preference/abilities. It also contains components complimentary to the notion of Platinum and Standard accounts. This does not cover the mechanics of mission ownership for completion, mission objective security, safeguards, etc. Those issues will be covered in a separate document.

Vision and goals

Goals

- Create a simple, highly configurable location for player postings
- Should be able to access mission boards on or off line.
- Must be easy to sort, and scan.
- Must have a strong obscenity filtering routines.
- Provide players the ability to flag mission descriptions as inappropriate/offensive for GM review. Self-Police.
- Add value to Premium Accounts

Vision

- Must give tools to players to shape their own experiences
- Must support many different styles of play
- Should encourage specialization
- Must be playable via the Web in some fashion
- Must minimize human administrative intervention
- Should empower players to take action against those who are obnoxious
- Should have an intuitive user interface

Walkthrough

This document will step through:

- Basic Look and Feel of Board
- Basic Fields/Parameters/Filters
- Rules Wizard
- How to Post a Mission
- How to examine/bid on/accept a mission
- Post Decay mechanisms
- Ownership decay mechanisms
- Premium Account Perks

Systems design

Look and Feel

The mission board will be a very clean, tabled view that provides an abstract and all necessary field info.

A good example of this is the Window's Explorer and Window's Outlook Tabs/Fields. In Outlook you can customize what fields are available with the Field Chooser, and sort just by clicking on a particular tab.

So a player can have as many or as few tabs available to sort by. Players can also setup rules for auto filtering. This is covered briefly in its own section.

Basic Fields and Parameters

The basic fields for a mission board entry include:

- **Title:** The title of the mission. Dbl Click for mission detail.
- **Client:** Who is posting the mission? In detailed mode you can access our EBAY style player profile system to see if this client is on the level.
- **Mission Type:** Delivery, Escort, Patrol, Assault, Acquire, Survey, etc.
- **Difficulty level:** Easy, Medium, Hard, Very Hard
- **Recommended Primary Certification and Level:**
- **Recommended Secondary Certification and Level:** Optional
- **Recommended Tertiary Certification and Level:** Optional
- **Sector:** (automatic based on System specified)
- **Quadrant:** (automatic based on System specified)
- **Start System (if any):** Where the prospective employee must go to pick up goods.
- **Objective System:**
- **Current Distance from System:** In jumps (automatic). Player can click to connect to his/her Nav map to get a route drawn.
- **Mission Location Security:** Wild or Safe (automatic)
- **Ground or Space Based:**
- **Time Posted:** When the mission was posted, automatic.
- **Time Limit:** One day from pickup, two hours from posting, etc.
- **Client Approval Required (y/n):** Client approval is the process by which the player who posted the mission has the option of screening prospective applicants.
- **Value:** Either fixed value or an initial Bid value that prospective Privateers can fight it out over E-Auction style. This might be an item or items as opposed to credits.
- **Abstract:** One line additional description. If a mission is blown up this is replaced by Detail.
- **Detail:** Detailed instructions, replaces Abstract if a mission is blown-up.

All of these fields/tabs sort missions in either ascending or descending order.

Rules Wizard

We will also enable the player to set up simple rules to automatically filter the missions shown on their board. While this is TBD, example filters could be:

- Only display available missions within x number of jumps
- Only display missions in Safe Space (or only display Wild Space)
- Only display missions that pay x or more
- Only display missions for x certification(s) skill(s) (check all that apply).
- Only display missions that do not require Client Approval
- Only display missions that do not have a time limit.
- Only display missions posted within x number of days, hours.

How to Post a Mission

A player will be able to post a mission at any time. They must fill out all of the non-automated and/or optional fields; assign a time limit (if any), select if they want to approve a potential applicant, recommended skills needed, arbitrary difficulty level, value or bid, abstract and detail.

In the case of a Client Approval mission, they will be contacted by message whenever a prospective employee answers the call. The client will then be able to examine that player's profile and determine whether or not they should be hired. In either case the prospective employee will be messaged with an Accepted or Declined message.

Although the specifics of mission acceptance, completion, etc. are to be covered in a separate document it is worth mentioning that when a Client posts a mission with an assigned value, or agrees to a bid, those funds are frozen until either the mission is completed or the employee defaults on the mission (time limit, etc.).

Concepts of payments up front, etc. are still being discussed.

How to Examine/Bid On/Accept a Mission

A player will first be greeted with a filtered and sorted list of available missions that contain the one line abstract

By holding over a section of the current mission a drop down will appear containing a longer detail hinted at by the abstract.

By clicking on the mission they pull up a separate window with the accept/and bid options, Client Profile, etc. They then have five minutes to accept the mission, accept and wait for approval (if Client Approval Required), place a bid, or return to the mission board.

During this time that mission is locked out of the mission board so that another player can't come and accept out from under him/her. For people viewing the board that attempt to view a mission currently viewed by another player, they will receive a message, "This mission is under evaluation by another player". Once a mission is accepted it is removed from the message board.

A player can only view the detail window of **one** mission at a time. This is to ensure a single player can't lock out multiple available missions from being viewed.

A player can only accept **three** missions at any given time. This is to keep players from hogging or draining the mission board.

Self-Policing

If a player reads a mission description that is offensive and managed to pass through our obscenity filter, they can flag that mission as offensive. The method for flagging the mission is TBD. Once flagged, this calls that mission out for a GM to examine during standard Mission Board oversight. If the mission is truly offensive, it will be removed and the person who posted contacted. If it is not, the mission will be left as is.

If any player flags more than five missions in a given play session as offensive, an automatic call to a GM is placed to see if that player is abusing their power. This encourages players to take action and self-police.

The player-to-player mechanics of accepting a mission and then picking up goods or personnel to be transferred are TBD. Ideas include: a client flags the employee with a unique "transponder" code (basically a permission) to pick up and drop off goods.

Post Decay Mechanisms

After a mission is posted it will remain on the board for x amount of time before the player who posted is sent a message indicating that the time limit for mission acceptance has expired.

They will then be given the option to resubmit the mission to the board.

Other ways of managing out of date mission data is being discussed.

Ownership Decay Mechanisms

After a player accepts a mission they have x amount of time to complete it. This is either established by the Client under the Time Limit field or is x amount of time before the player defaults and the mission is either reposted to the board, or deleted and the Client notified if it has exceeded the Post Decay limit.

Other ways of managing out of ownership decay is being discussed.

Premium Account Perks

Premium account members could have rules set up that notify them when a mission that meets their specifications becomes available. These notifications could be sent via email to the player.

Other reasons for Email notification include:

- A player has accepted a mission that requires the Client approve.
- A mission has been removed from the board due to decay.

This and other facets of an "offline" email strategy could make the premium account a highly desirable commodity.

Interface design

TBD

Tasks

- Mission Board Interface/Art
- Individual Mission detail interface/Art
- Client Posting Interface/Art
- Message Board Rules Wizard
- Email design

To be addressed

- Mission ownership
- Objective safeguards for AI generated missions
- Player to Player cargo pickup and delivery methods.

Cloning and Insurance

Overview

Death and loss are inevitabilities in Privateer Online. However, permanent death and total loss are not. This is not fun after one has spent a month or six developing their character. The two systems to handle this are Cloning and Insurance. Cloning ensures the player will have a body to return to when their original is blown to bits. Insurance provides for a percentage recoup of any and all items lost when the player's space or ground belongings are destroyed.

Goals and vision

GOALS

- The cloning system is a way to allow people to maintain their character and his/her abilities in the event of a catastrophe.
- Provide a means for players to insure their belongings against loss due to death, etc.
- It will have levels of resolution that increase in cost as the character grows stronger and has more need of a stronger safety net.
- These levels of resolution should provide a tactical/game element for disaster preparation.

VISION

- Death should be a situation with penalties to avoid, but not at the expense of having fun and exploring the universe.
- Players will be able to explore without the annoying fear of having to start from scratch or go madly scrabbling for their corpse.
- Players may actually confront a version of themselves in combat!

Walkthrough

Cloning

Cloning replaces a new character's avatar with the same physical body model at its current state of development. As a player gains experience the cost of a clone will go up dependent upon:

- Level of Certification
- Reputation status

Insurance

Insurance provides all players with a base minimum of coverage. In addition to this players can purchase Insurance plans that cover a percentage of their worth above this minimum. The elements to be described:

- Basic "Newbie" Insurance

- Insurance Plan Scale
- Ground Insurance
- Insurance cost calculation
- Ship/Goods replacement scheme

Systems design

Fictional Background for Cloning

Cloning was developed as the method by which the Confederation was able to push back the Kilrathi attack fleets that nearly eliminated all of humankind.

The drawback to this was the introduction of the *Ghouls*, rogue clones accidentally released after a Kilrathi guerilla attack on a Confederation facility. These accidentally awakened clones were to be eliminated since their "templates" still existed, however being extremely skilled Confederation pilots/soldiers, they managed to escape before this occurred. They took refuge in Dead Space and have terrorized all who cross their path, Human, Kilrathi, and Firekkan.

A cry went out to cancel the program, but the rebuilding effort took precedence and new explorers were given access to cloning facilities to allow them to make the dangerous leaps necessary to ensure humankind's sustained presence in the galaxy.

Even now, clones have been known to escape with the assistance of covert Ghouls seeking to increase their ranks/liberate fellow brethren. Many a terrified tale of pilots confronting themselves in dogfights have twisted the ear of anxious new explorers.

Cloning - The How-To

Cloning is a very important system. It is the means by which the players start again when they "die". All players will have access to a basic clone. This is a simple template of the character at his/her starting state with a hit to current certification and reputation levels. Although the player may have received numerous certifications and a stellar reputation up to that point, those facets of the player's persona will be impacted by a clone's **memory**.

Memory involves the **flash mapping** of neural constructs (ganglia, dendritic spikes, etc.) across all of the body. Clone **flashmaps** vary in their level of resolution.

Low level certifications are the easiest to retain, but as the number of total certification disciplines increase and/or their certification value increases the chance of loss are raised. Reputation will also be impacted; this is TBD.

There will be four levels of Flashmaps available to player that will reduce these penalties. The levels of reduction as well as the basic certification hits that come with dying are TBD.

Ghoulish Potential

There is the potential that a player's "template" could spawn a Ghoul. The more often a player dies and is cloned, the higher percentage chance of a Ghoul being spawned exists. This leads to the possibility of a player confronting his/her own irate double in a dogfight or ground conflict.

This doesn't mean a persistent copy of the character will be generated to roam freely, it means that if the player encounters Ghouls craft or ground personas there is x% chance that one of them will be a copy of the player, with the player's name modified to a Ghoulish naming convention (TBD).

Insurance - The What-for

All players will have a minimum level of insurance that will equal the component cost of all starting equipment (ship, goods, etc.). This cost will be reimbursed in credits that can only be spent replacing the goods lost. (More on ship/component replacement below)

In addition, a player will be able to purchase one of four insurance plans to cover 25%, 40%, 55%, or 70% of their ship's components and/or avatar equipment worth. A plan's cost is determined by:

- The player's reputation
- System frequency (whether they spend more time in safe space vs. wild space)
- Current value of all installed components
- Percentage level of coverage
- Number of times a claim has been filed

Terrestrial structures will be subject to damage in the game as well. Although not destructible they can be rendered inoperative. The cost of repairing these installations can be significant. Players with terrestrial holdings (facilities, etc.) can purchase ground-based insurance to cover up to 80% of all repair costs incurred. These terrestrial insurance plans are structured the same way as ship/avatar insurance with the exception of System Frequency. System Frequency is replaced by a basic check to determine if the player's holdings are in wild or safe space.

Replacing Your Ship and other goods

If your ship is destroyed in the game you will be returned to the nearest cloning facility (if you died) or the nearest station/base if you were rescued. At this time you will be able to replace your ship using whatever insurance scheme you settled on.

You will have two options for ship replacement, either take the nearest auto-configured equivalent to your current ship and components or construct your new ship based upon currently available chassis, components, etc. that are available under the percentage rating of your replacement plan. The components you can select from are restricted to ones currently existing on the commodities exchange, and of a percentage credit equivalency to your previous components.

An interface/layout for this is TBD.

Personal goods work the same way. If you are murdered on a planet and lose your armor, weapon, equipment, etc. You will be able to select replacements within each category that are currently A) available on the commodities market and B) at a percentage value (tech/credit) of the original item.

Unique player crafted or very rare items, either ship or avatar, cannot be replaced. But a lower tech, currently available market equivalent will be provided as a replacement option.

Interface design

TBD

Tasks

Required Art/Interface - cloning

- Cloning facility - terrestrial structure
- Cloning facility - star base
- Cloning Plan select/purchase screens
- Clone Update screens

Required Art/Interface - Insurance

- Insurance plan select/purchase screens
- Insurance claim report
- Insurance goods replacement screen
- Insurance ship/ship-component replacement screen
- Terrestrial Structure Repair Screen
- Billing Statement

To be addressed

- How do "certification hits" function after death?
- How is Reputation impacted by death?
- What are the levels of certification/reputation protection for the Clone types?
- What is the base minimum for Insurance coverage?
- Designed Interface screens for Cloning
- Designed interface screens for Insurance
- Ghoul naming convention (How to mod existing player name to Ghoul name)

Ground Combat

Overview

Ground combat is a necessary system when any type of ground component is present. It is the most logical and fulfilling way to introduce tension and conflict to such a setting. This document is intended to describe a ground combat system that would be both appropriate to vision of the game as stated and that would be compatible with our time constraints.

Vision and goals

GOALS

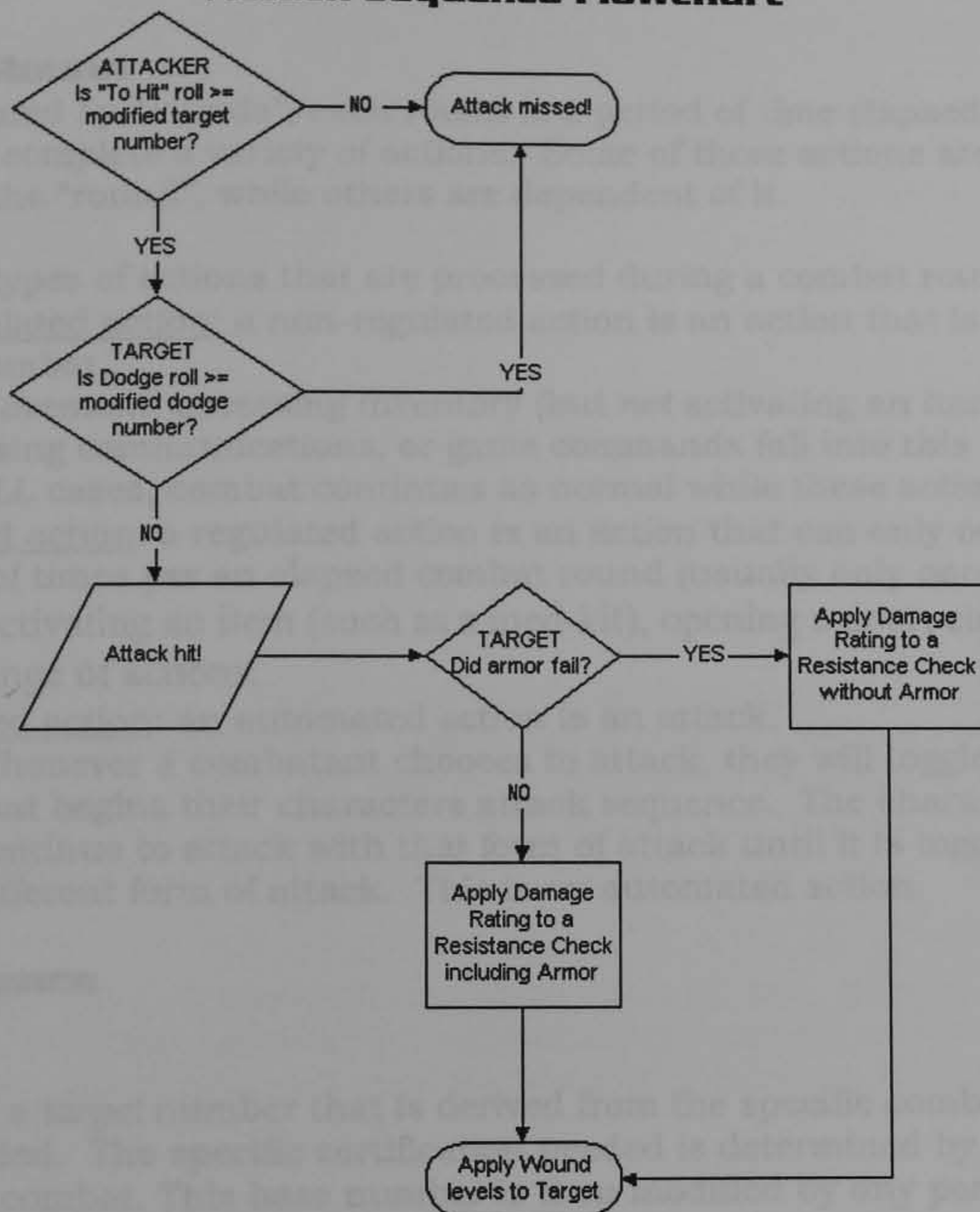
- ☐ The interface should be simple to understand and control
- ☐ The combat system should have depth, not complexity
- ☐ *The combat system should have a simple and clear path to balance; in effect, reduce the number of different variables to be balanced.*

VISION

- ☐ Ground Combat should have an intuitive user interface
- ☐ Ground Combat should expose new strategies throughout the game to the player
- ☐ Ground Combat will make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount

Walkthrough

Attack Sequence Flowchart



Attack Sequence

To Hit

The To Hit roll is a target number that is derived from the specific combat-related calculation needed. The specific roll is determined by whether it is ranged or melee combat. This base number is then modified by the attacker's certification, equipment bonuses, and situational bonuses. Once this target number has been modified, a random number is generated that must equal or exceed the target number to successfully hit the desired target. If successful, the defender has a small chance to avoid the attack. This chance to avoid is called the Dodge.

Dodge

The Dodge number is derived from the target's Coordination statistic and any pertinent certifications. A random number is generated and compared to the Dodge number. If the generated number meets or exceeds the Dodge number, the target has successfully dodged the attack. It is important to note that a target successfully dodging is a much less common occurrence than a target being hit by an attack.

Damage

Once a target is successfully hit from an attack, the damage rating from the weapon will be applied to the target once a resistance check is determined. The resistance check determines how much of the damage actually was absorbed or deflected by the target's armor and natural defenses as opposed to what amount is applied to the target's health. In certain cases, a failure roll will be made before the resistance check is determined due to the condition of the armor. (See Structure Failing below)

Damage Rating

Each weapon is given a damage rating which represents the severity of damage dealt before resistance, armor, and damage traits. The damage rating is measured in steps, each representing a progressive wound level. The first amount is listed in time from how to start to require to long to extend, each being an increasing amount of

Systems design

The Combat Round

Combat is regulated by "rounds", each round is a period of time elapsed wherein the combatants can complete a variety of actions. Some of these actions are tied explicitly to the timing of the "round", while others are dependent of it.

There are three types of actions that are processed during a combat round:

- Non-regulated action: a non-regulated action is an action that is not restricted during combat.
Movement, accessing inventory (but *not* activating an item in inventory), using communications, or game commands fall into this category. In ALL cases, combat continues as normal while these actions are taken.
- Regulated action: a regulated action is an action that can only occur a certain number of times per an elapsed combat round (usually only once).
Activating an item (such as a med-kit), opening a door, etc. fall into this range of actions.
- Automated action: an automated action is an attack.
Whenever a combatant chooses to attack, they will toggle a command that begins their characters attack sequence. The character will continue to attack with that form of attack until it is toggled off or to a different form of attack. This is an automated action.

Attack Sequence

To Hit

The *To Hit* roll is a *target* number that is derived from the specific combat-related certification needed. The specific certification needed is determined by whether it is ranged or melee combat. This base number is then modified by any pertinent certifications, equipment bonuses, and statistical bonuses. Once this *target* number has been modified, a random number is generated that must equal or exceed the *target* number to successfully hit the desired target. If successful, the defender has a small chance to avoid the attack. This chance to avoid is called the *dodge*.

Dodge

The *dodge* number is derived from the target's Coordination statistic and any pertinent certifications. A random number is generated and compared to the *dodge* number. If the generated number meets or exceeds the *dodge* number, the target has successfully avoided the attack. It is important to note that a target successfully dodging is a much less common occurrence than a target being hit by an attack.

Resist

Once a target is successfully hit from an attack, the damage rating from the weapon will be applied to the target once a resistance check is determined. The resistance check determines how much of the damage actually was absorbed or deflected by the target's armor and natural defenses as opposed to what amount is applied to the target's health. In certain cases, a *failure* roll will be made before the resistance check is determined due to the condition of the armor. (See Structure Rating below)

Damage Rating

Each weapon is given a damage rating which represents the severity of damage dealt (before resistance), stun amount, and damage type(s). The damage rating is measured in *steps*, each representing a progressive wound level. The stun amount is rated in time from none to short to medium to long to extended, each being an increasing amount of

time before the target can function (other than a slow movement). (Specific time amounts TBD) The damage type is type of energy used to conduct the damage such as kinetic, heat, and etc. (Specific type list TBD) A weapon may have multiple damage types.

Armor

Armor has a resistance rating, which is measured in *steps*, for each damage type and stun. Armor also has an inherent structure rating, which is measured in *steps* as well.

Resistance Rating

The resistance rating of a specific set of armor is the number of steps of wound level decreased by the armor.

For example, Jimmy has no wounds and is hit with a blast from a laspistol. The laspistol is rated to do 2 *steps* of damage. Jimmy's armor is rated at 1 *step*. After any applicable modifiers, the final tally is that Jimmy's armor resisted 1 *step*, thus reducing the final damage to one *step* on the wound state.

Structure Rating

The structure rating is used to determine the damage the armor takes as it absorbs attacks. Each time a specific set of armor resists damage it takes structure damage related directly to the amount of steps absorbed from the attack. Once a specific piece of armor has taken more than half of its structure rating in damage, it will only resist damage 50% of the time. This is called a *failure roll*.

For example, Jimmy was hit in the above example by a laspistol for 1 *step* of damage. The original damage rating was 2 *steps* and his armor resisted one *step* of damage. Since the armor resisted one *step* of damage, it takes one *step* in structure rating.

Applying the Unresisted Damage

Once a Resistance check has been assessed, the remaining damage rating is applied to the target's current Health level. It is applied by tallying the number of *steps* that are remaining, and incrementing the Health level by these *steps*.

For example, Jimmy, in the above example, resisted the damage from the laspistol down to one step. Since Jimmy was at Full Health before the attack, his Health Level is incremented one step to Lightly Wounded.

Health Levels

- Full Health: This is the full health state.
- Lightly Wounded: This is the first *step* of damaged Health.
- Moderately Wounded: This is the second *step* of damaged Health.
- Seriously Wounded: This is the third *step* of damaged Health.
- Critically Wounded: This is the final *step* of damaged Health.
- Unconscious: When a target takes any *steps* of damage once they have been Critically wounded, they enter into a state of Unconsciousness. They have 30 seconds to receive some form of medical attention. If medical attention is received, they return to Critically wounded Health level.
- Dead: If 30 seconds expire without medical attention, the target dies.

Combat Balance

This concept, while not technically a separate system, merits its own section. The goal of combat balance is to isolate the primary balance variables to one or two specific points. In Ground Combat, the majority of balancing is directly on the array of weapon ratings and armor ratings. To a lesser extent, the effects of the Coordination, Strength,

and combat-related certifications must be balanced as well. Since everything has a fixed set of Health levels (see *Health Levels*), there is no balancing of hit points; as well, since there are few randomly generated numbers (none in damage), testing balancing becomes much more of a bearable process. Basically, the priority and path for balancing is as follows:

- Relationship of Damage Ratings and Armor Resistances. The equipment wielded by the participants is the major modifier to the damage dealt to a target. To a much lesser extent, an *avatar's* Strength may modify the damage as well but only in melee combat; as well, a small collection of combat-related certifications may modify the damage.
- Restriction of Combat Equipment by Certifications. All equipment can only be wielded with the proper certifications. This is one of the primary tools to balancing an *avatar's* ability to deal and withstand damage.
- To Hit Roll governed by Combat Certifications. The ability of an *avatar* to successfully hit a target will be tied to a specific set of combat-related certifications, and it will also be slightly modified by an *avatar's* Coordination statistic.
- Dodge Roll governed by Coordination. The ability of an *avatar* to dodge is directly related to Coordination; and to a lesser extent, certain combat certifications.

Interface design

Combat Controls

The actual control of combat is designed to be simple. The control types needed are a combat toggle and a targeting mechanism (both mouse-driven and keyboard); the targeting mechanism should have keyboard equivalents for not only nearest target, but also nearest friendly and nearest enemy. The standard ground controls such as inventory commands and movement will be used as well. Ground based commands that fall into the Regulated actions should have the use-limiter built in to apply at all times during the ground experience. An example of this would be using a first-aid kit; it should only be usable once per time period (as defined by combat round length) even when not engaged in combat.

Combat Display

The combat display has four key pieces information that need to be shown. The pieces are the player's avatar current Health level, the target's current Health level, the status of the avatar's current weapon, and the status of the avatar's current armor. The Health level displays for both the target and an avatar should follow the Health Level Display description (see below). The current weapon and current armor display will be a pop-up window that can be toggled on/off.

Health Level Display

- Full Health: It is visually represented as a fully filled green indicator.
- Lightly Wounded: It is visually represented as a $\frac{3}{4}$ filled green indicator.
- Moderately Wounded: Visually represented as a $\frac{1}{2}$ filled yellow indicator.
- Seriously Wounded: Visually represented as a $\frac{1}{4}$ filled red indicator.
- Critically Wounded: Visually represented as a flashing unfilled red indicator.
- Unconscious: Visually represented as a flashing unfilled black indicator.
- Dead: Visually represent as an unfilled black indicator.

Tasks

- Creation of the item database and information for all combat-related items
- Code ground combat rules
- Code action use in relation to all actions defined as *Regulated*.
- Artwork for Health Level indicators
- Artwork for the weapon/armor combat display
- Artwork for combat effects (such as smoke, explosions, etc.)
- Code Ground combat sections of the Interface
- Ongoing balancing of the statistics involved with Ground combat

To be addressed

- Prototype Ground combat system
- Design of the specific modifiers from Coordination, Strength, and combat-related certifications
- Generate list and statistics of all combat-related items.

Vision and goals**Goals**

- Recapture and enhance the experience of the P1 jump process
- Keep the function simple, but the fun
- Provide multiple ways to get out of a situation
- Make as fully automated as possible
- What isn't auto should use a minimal number of controls
- Provide other options for players

Visual Points

- ... should have an intuitive user interface
- ... should not require mapping—the player should never feel lost
- ... should expose new strategies through the game to the player
- ... will provide problems to experienced players
- ... should have big visual payoffs

Walkthrough**To be covered in this document:**

- What are jump points?
- What do I need to have to make a system jump?
- The jump drive
- How to Jump
- Other methods of high speed travel between systems/quadrants/universes

Additionally the Primary Parameters of each system for purposes of ship installation and any minimum requirements and the Secondary Range of necessary components (aggravated) for creating the fleet are listed.

Example Example

Jimmy knew he should never have taken the job. Here he was flying a single seat of level 5 software and electronics packages in a newly established planet in WOL Space.

Jump Points, Jump Drive and Jumping

Overview

Jump travel is how players move from system to system. Systems can contain an arbitrary number of jump points to nearby systems. Jump travel was one of the most exciting aspects of Privateer1...What would await you on the other side? Would you make the jump before the bad guys blast you? How WCO - Privateer will capture and enhance this excitement is covered here as well as other modes of high-speed intersystem transit.

Vision and goals

Goals

- ☐ Recapture and enhance the suspense of the P1 jump process
- ☐ Keep the function simple, but the form deep.
- ☐ Provide multiple ways to get out of a situation.
- ☐ Make as fully automated as possible.
- ☐ What isn't auto should use a minimal number of controls.
- ☐ Provide other options for players to travel large distances.

Vision Points

- ☐ ... should have an intuitive user interface
- ☐ ... should not require mapping—the player should never feel lost
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... will provide privileges to experienced players
- ☐ ... should have big visual payoffs

Walkthrough

To be covered in this document:

- What are jump points?
- What do I need to have to make a system jump?
- The jump drive
- How to Jump
- Other methods of high speed travel between systems/quadrants/sectors

Additionally the *Primary Properties* of each system (for purposes of ship installation and any minimum requirements) and the *Schematic Recipe* of necessary components (ingredients) for crafting the item are listed.

Gameplay Example

Jimmy knew he should never have taken the job. Here he was flying a shipment of level 5 software and electronics packages to a newly established planet in Wild Space. The

price was right, but they were a little less than forthcoming with the level of danger. A famed criminal outfit has decided to crush the new planet and has established a blockade in the surrounding systems.

Jimmy safely makes it to within one system hop from his target when the bastards are on him. He activates a chemical booster to rocket him towards the jump buoy. The pirates regain their bearings and then begin full afterburner pursuit.

As Jimmy makes it into the jump buoy volume his jump drive whirrs to life and a pop-up shows him his time to full Jump Lock calculations. The pirates make it to him and begin their attack runs with the Jump Indicator showing half the time has elapsed...leaving him with a 50% chance of failure should he jump early.

He activates his automated defenses and manages to cripple one of the light fighters with a couple of friend or foe missiles. Still, there are too many...and he's only at 30% jump success.

Normally, Jimmy wouldn't risk this, he's always managed to secure a safe Jump Lock before engaging the drive. But this time, it's too dangerous to wait...better the risk of resetting the Jump Lock calculations than a hundred plasma bolt hits.

He hits J, is asked to confirm his pre-jump lock decision, and he confirms with a heavy finger press on the Y. Just as two heavy fighters arrive and launch two torpedoes at his transport the drive kicks in and his ship blasts into jump space.

After a couple of seconds of finger-crossing Jimmy watches his ship enter into his target system, a compliment of defense drones awaiting his arrival to escort him to the planet.

Systems design

What are jump points?

Fiction Note: Although accepted phenomena, argument continues as to whether jump points are naturally occurring phenomena or constructs left behind by a dead alien race. Fingers have been pointed to the Stelteck after past engagements with drones indicated these entities could jump seemingly without a jump drive. This conclusion was made after the lack of a jump engine signature was noted during evaluation of these drone craft.

A common misconception with folks new to the WC universe is that jump points are akin to jump gates. Jump gates are portals the player's ship travels through (ala Bab 5). Jump points are space volumes that contain a subspace pipeline with an exit to another system.

Intensive LR probe scans and careful research discover jump points.

When a jump point is discovered, the center of each of the two volumes tied to the signature (for each system) is marked with beacons. These beacons allow craft to easily navigate the otherwise invisible entities.

Without beacons, finding a jump point would require a costly array of sensors that most spacefarers don't have the space to install, money to buy, or savvy to operate.

Gameplay Note: There are many systems and jump points that are unregistered with Confed. Cloaked and camouflaged beacons only become visible/active when the correct transponder code is detected in system. They can also be detected by other means (cloaked ship detection tactics etc.). These hidden jumps and systems are usually the havens of criminals who desire privacy for their operations.

What do I need to make a system jump?

In order to jump to a system you need:

- Jump Drive
- to be within a Jump Point volume

In the original *Privateer* you had to buy individual quadrant maps in order to know where the hell you were after you came out of an inter-quad jump. Times have changed.

Now, all ships are outfitted with the latest officially registered jump points available from Confed in their nav map. As jump points are discovered the explorer responsible can register them with Confed (to be able to officially register a system name, receive free beacon placement etc.) and all players nav maps are updated with the new system and jump point info. This is covered more in {document name}.

The Jump Drive

All start ships have a basic jump drive installed. Unlike *Privateer 1*, restricting players to the newbie start system until they can afford a jump drive will prove a difficult task to make fun. Possible options include: Force players to fly Corporate vessels (transports, fighters, etc.) to complete missions until they have the cash to afford their own jump drive. This, however, still feels like an inadequate solution for the huge player pile-ups that are likely to occur in newbie systems; should they be restricted by lack of a jump drive.

This is not to say that players can't construct ships without jump drives for intersystem delivery of resources, etc.

Jump drives have a "calculation time" that must be met before a safe jump is possible (covered in How to Jump section). This is a crucial statistic that can be modified by subsequent Navigational Computer software add-ons as well as master crafted Jump Drive properties. These add-ons decrease the amount of calculation time it takes for a jump drive to power up.

Primary Properties:

- **Size:** Capship jump drive vs. standard jump drive
- **Weight:** Determined by housing materials, impacts overall ship speed/handling etc.
- **Jump Lock Calculation Speed:** Average time (TBD) it takes the drive to calculate the jump between two points and sync up with the subspace signature.

Schematic Recipe:

- **Housing-** Jump drive housing, either Capship or standard, Structural material (materials TBD) and used to craft and type of housing determines weight.
- **Software Package:** (packages TBD) Higher package tech levels may decrease calculation time by up to 10%.
- **Electronics:** Electronics package (packages TBD) Higher package tech levels may decrease calculation time by up to 10%.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase calculation speed by up to 10%.
- **Tools/Facilities that are required to construct** (TBD) Tech level of facilities or tools used may decrease calculation time by up to 10%.

How to Jump

Now you have your jump drive installed and have just entered a jump point volume. You then hit **J** to initiate a jump. At this point a pop-up appears that indicates the status of the navigational calculations being made by the jump drive.

You can hit **J** again to jump during any stage, but it is a dangerous prospect until the navigation calculations are complete, if you fail the jump calculations must restart. Once the calculations are complete, and you are within range, the ship automatically makes the jump.

WARNING

This Jump Leads to Wild Space
Continue jump into Wild Space?
Y/N

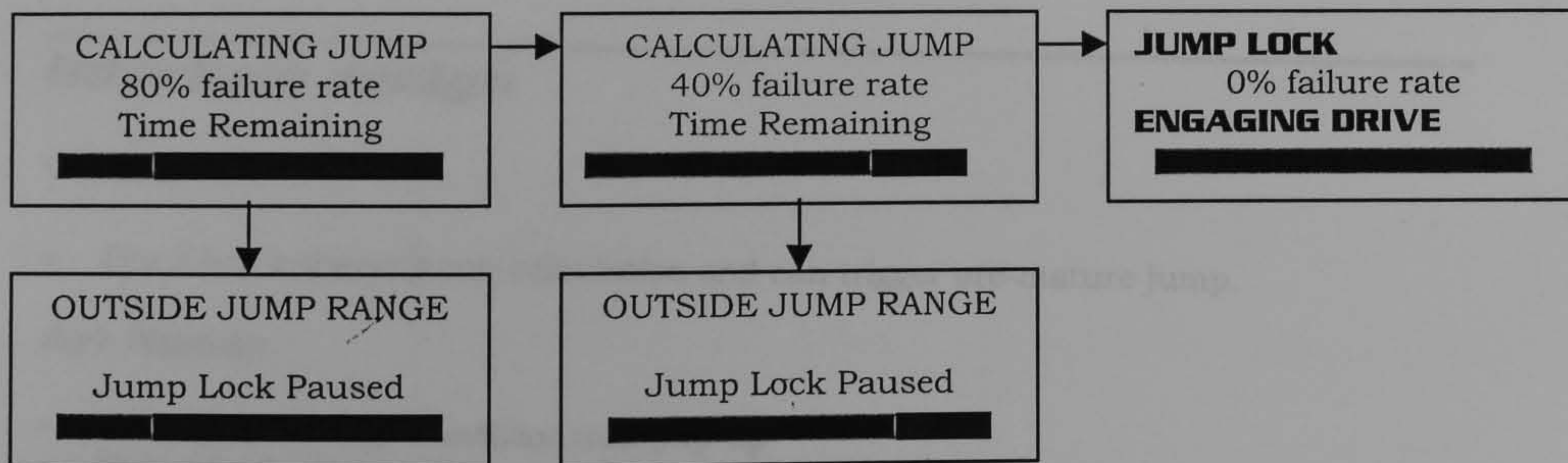
If a jump will take a player into Wild Space they will be warned with a pop-up and jump warning klaxon that asks for confirmation.

You must remain within the jump point volume until the calculations are complete and the drive has established a **Jump Lock**. If you leave the volume, the calculations are **paused, not reset**, until they reestablish range to the jump buoy. However, if the player remains outside of a jump point volume for x period of time the Jump Lock calculation is reset. This is to prevent a potential strategy of charging up the jump drives to near jump, then leaving the jump volume to go attack folks and then being able to jump out faster than normal if you get into trouble.

The total time necessary for a jump lock to be established is divided in half. During the first half the jump is unlikely to succeed with a scaling percentage chance of jump calculation reset (failure) starting at 100% and scaling down to 51%.

The second half is safer, but still dangerous starting at a 50% chance of jump calculation reset (failure) until the Jump Lock is complete.

When **J** is first hit a disconnected series of whirrs and blips indicate the calculations are underway, these sounds begin to slowly sync together until a final bell sounds to indicate Jump lock.



If you decide to jump before the calculations are complete you are asked to confirm by hitting **J** again (pop-up and jump warning klaxon).

WARNING

Jump Lock NOT established
Proceed with partial lock jump?
Y/N

This adds a new wrinkle to the Privateer 1 moment of watching your ship power up its jump drive while bad guys hammer on you. Then you have to make the call of waiting

for the calculations to complete or take the risk of a pre-calc jump. This affords a new level of strategy when attempting to make a hasty exit.

When a jump is engaged we can take one of two visual tactics:

- The in-cockpit millenium falcon star field-blur and rush down pipeline culminating with a flash as the player exits the jump.
- Switch to an external view as a Privateer 1 style jump bubble encapsulates the ship and sends it rocketing down a tight pipeline of rushing stars, switching to the WC3-WC5 external jump effect as the player exits the jump.

When you exit jump you enter the system at a random point within x distance of the jump volume. This is to minimize camping PKs waiting at **exactly** the right spot.

Other Methods of High Speed Travel between Systems/Sectors/Quadrants

Inter-Sector travel methods, to bypass the tedium of jumping 20-30 systems to visit a friend are being discussed.

Here are a couple of ideas:

Newbie start zones have a newly implemented Sector Gate that requires an intense number of resources, and Confederation oversight to run. Entering one of these can take you directly to another sector's newbie start zone (if it happens to be equipped with such a device).

For players outside the legal loop (or advanced players that don't want to trek to a newbie start system), there are wormholes, found only within specific nebula, which are capable of taking the player from one sector to another. This requires specialized jump drive modifications.

These and other methods are TBD.

Interface design

Controls

- The **J** key initiates jump calculation and can trigger pre-mature jump.

Art Needs

- **Wild Space Jump Confirmation pop-up**
- **Jump Lock status pop-ups**
- **Partial Lock Jump confirmation pop-up**
- **Jump effect**

SFX Needs

- **Jump Lock Calculation sound cycle start:** Start sound has disjointed clicks and whirrs as well as the jump drive powering up sound (basically a camera flash power up).
- **Jump Lock Calculation half-way-point:** clicks and whirrs sync up as jump drive sound continues to rise in pitch.

- **Jump Lock Calculation complete:** An affirming beep(s) to indicate jump lock complete
- **Start Jump Sound:** The sound of entering a jump
- **Jump Loop:** The looping sound of traveling through jump space
- **End Jump sound:** The rev down of coming out of a jump.
- **Jump Failure sound and klaxon:** A power down noise (like the ill fated jumps the Millenium Falcon experienced) followed by a pulsing klaxon indicating jump failure.

Tasks

- Art needs (both assets and implementation)
- Sound needs (both assets and implementation)
- Jump Buoy volume event system (detect range, suspend and continue calculations, etc.)
- Pop-up triggers with confirmation hooks (if applicable)
- Jump effect implementation (Art, Code, and Sound needs)

To be addressed

- Final version of jump effect
- Other methods of Inter-Sector Inter-Quadrant travel

Walkthrough

To be covered in this document:

- Customization play experience
- What are the basic ship types?
- What are the Components of a ship?
- How can a ship be further customized?
- What kinds of skills & customization can be performed?
- Interface design

Customization Play Experience

Terry was a fairly new player, but had been advancing in Skills and Pyrotech regularly since becoming an apprentice to Dorian. Now that his certification allowed it, he wanted to whip up his Victory Fighter and see what she could do with some high-end components.

While docked at his home starbase, he entered the Customization Screen and purchased the best components he could afford, including an engine that allowed his ship to better defend itself and jump longer.

After receiving his new components, he decided to check his ship's current configuration and start from scratch. Carefully dragging his new purchases onto the open racks of his ship, he saw the statistics of his ship advance into an entirely new level of performance.

Now that he was standing to run with the big boys, Terry decided to get a little fancy and display a logo. He carefully scanned through the library of the ones he liked, and chose to combine the two logos Dorian used on her ship for his own.

Ship Customization

Overview

In *Wing Commander Online*, ships can be customized with different hulls and components. In addition, the ability to further customize a ship with custom textures, sound effects, and lighting configurations provides a tremendous mechanism for the player to develop a truly unique online identity.

Vision and goals

GOALS

- ☐ Provide a wide range of unique and exciting ship types
- ☐ Provide many different utility modules that provide extra levels of capability to a ship
- ☐ Encourage players to make their ship an extension of their online persona

VISION

- ☐ Make ship customization so flexible and accessible it becomes a sub-game within *Wing Commander Online*.
- ☐ Encourage a sense of community by allowing players to customize their ships to reflect a squadron or other organization.

Walkthrough

To be covered in this document:

- Customization play experience
- What are the basic ship types?
- What are the Components of a ship?
- How can a ship be further customized?
- What kinds of artistic customization can be performed?
- Interface design

Customization Play Experience

Terry was a fairly new player, but had been advancing in Skills and Prestige regularly since becoming an apprentice to Doreen. Now that his certification allowed it, he wanted to soup up his Heavy Fighter and see what she could do with some high-octane components.

While docked at his home starbase, he entered the Customization Screen and purchased the best components he could afford, focusing on ones that allowed his ship to better defend itself and store salvage.

After receiving his new components, he elected to Clear his ship's current configuration and start from scratch. Quickly dragging his new purchases onto the open racks of his ship, he saw the statistics of his ship advance into an entirely new level of performance.

Now that he was starting to run with the Big Boys, Terry decided to get a little fancy and display a logo. He quickly scanned through the library of the ones he liked, and chose to combine the two logos Doreen used on her ship for his own.

Now that he was equipped for salvage and light mining, Terry decided to Save this configuration in case he wanted to go back to it without having to drag everything manually again. With that done, he exited the Configuration Screen, pressed the "Launch" button, and went out to see what his ship could do!

Systems Design

What are the basic ship types?

Multi-position *Capital Ships* are the heart and soul of the *Wing Commander Online* space experience, providing a mobile meeting place for player communities where they can refine their skills and develop their online identities while sharing team-oriented battle experiences.

Fighters and two-seat "*Bombers*" are the most familiar vehicles of the *Wing Commander* and *Privateer* experience. They provide players with fast-paced, accessible action, while allowing a wide range of multiplayer tactics and strategies.

Utility Ships (Transports, mining ships, etc.) are the trucks and trains of *Wing Commander Online*. Though players can pilot them, they'll more likely be used as autonomous server-run drones performing low-risk missions at the bidding of merchant characters. Note that *Utility Ships* can be equipped with a wide range of *Utility Modules*, as well as *Gun Pods* and *Fighter Pods* that can provide considerable punch while enhancing their versatility.

Each of these ship types comes in a variety of sizes and hull designs. These hulls not only look different, but also have varying capabilities and limitations such as:

- 1) Maximum weight
- 2) Number and size of engines
- 3) Area for mounting additional internal components
- 4) Number of "Hard Points" for mounting external components
- 5) Maximum crew

What are the components of a ship?

These hull types can be fitted with various Basic Components to provide minimum functionality:

- *Power Source*: Provides power to the ship's engines and systems
- *Engines*: The primary method of propulsion of the ship
- *Maneuver Package*: The thrusters and systems necessary to maneuver
- *Landing Bays*: (Capital and Utility Ship Only) Where smaller ships attach to or enter the capital ship
- *Shields*: Powerful magnetic field generators that diffuse the power of most weapons
- *Armor*: Plates of various materials designed to protect vulnerable systems
- *Point Defense Turrets*: (Excluding Fighters) Smaller weapons intended to engage fighters and anti-ship missiles
- *Main Guns/Weapons*: These include a wide range of energy weapons, mass-drivers, missiles, mines, and torpedoes.
- *Sensors*: Short and long-range radar, infrared, etc.
- *Backup Power*: Secondary reactor(s) or batteries used to provide power in the event the main reactor(s) are inoperative

- *Electronic Countermeasures:* Various electronic transmitters designed to foil radar detection and weapons guidance.
- *Ammunition Storage:* Special storage area for extra ammunition
- *Cargo Hold(s)*
- *Communications:* Radio equipment

All of these components come in dozens of variants, each representing different levels of quality, energy consumption, size, weight, cost, etc. This level of variation allows tremendous ability to customize the appearance and performance of a ship to suit the player's needs, preferences, and financial situation.

How can a ship be further customized?

Utility Modules:

In addition to the Basic Components, ships can also be fitted with Utility Modules to provide new or capabilities:

- *Navigation Beacons:* Provides the ability for allies to quickly locate an area of interest.
- *Asteroid Mining Module:* Allows a ship to mine and gather resources from asteroids.
- *Water Capture Module:* Allows a ship to gather water from objects such as comets.
- *Deep Scanners:* Can provide very long-range scanner coverage across an entire system.
- *Solar Panels:* Useful for backup power
- *Jammers:* Can be used to block radio communications and Navigation Beacon signals
- *Repair Dock:* Allows a ship to perform emergency repairs on smaller craft.
- *Sick Bay:* Can care for wounded players and NPCs.
- *Research Lab:* Can be used to perform research on artifacts, and advanced technology.
- *Utility Bots:* These small robots come in a variety of different configurations, each aimed at a specific duty:

Repair: Repair Bots improve the efficiency of Damage Control Teams by assisting from outside the ship.

Rescue: Rescue Bots are designed to immediately fly out of the ship and pick up a friendly ejected pilot.

Turret: Turret Bots carry light lasers, and can be deployed in groups of 5 to different Quadrants of the ship by the Tactical Officer. They'll fly to that area and maintain a distance of approximately 50 meters from the ship.

- *Datalinks:* This device allows allied ships to share sensor information with one another.
- *Samaritan:* This device prevents a ship's energy weapons from damaging a designated Ally, as well as preventing missile lock
- *Turtle Turrets:* These advanced Point Defense Turrets can retract nearly instantly into an armored recess in the hull to provide extra protection in the event of an attack against the turret itself.
- *Papoose:* This device can be fitted to Heavy Fighters or Bombers and allows them to attach a single Light Fighter (weight limitations apply).

- *Spotlights:* This system uses multiple high-power spotlights capable of tracking single or multiple enemy targets and highlighting them to assist friendly fighters and defensive turrets in locating them.
- *Missile Warning Radar:* This low-power Doppler radar system detects all missile types approaching it, including passive-guidance versions.
- *Snowblower:* Defensive device that ejects large clouds of carbon dioxide ice crystals into the space surrounding the ship to protect against lasers and other directed-energy weapons.
- *Hull Integrity Field:* (Cap Ship Only) This is an array of force fields that effectively doubles the Durability/Damage points of a ship's hull.
- *Sparklers:* A cloud of pyrotechnic dust used to disrupt IR sensors seekers and confuse pursuers.
- *Image Intensifier:* A device that can be used to "spotlight" a target or targets, providing a higher level of situational awareness during combat in low-light area.
- *C³I Pod:* The *Command, Control, Communication, and Intelligence* Pod contains a full sensor suite, and can be mounted on a Bomber or larger chassis. The communication systems aboard allows a crewmember of that vessel to serve as a Tactical Officer when the pod is activated.
- *Automatic Pilot:* Provides the ability to perform a series of scripted maneuvers while allowing the pilot to focus on another function.
- *Shield Projectors:* These devices can provide a "wall in space," by projecting a shield within their perimeter.

Special Systems:

In addition to the more common Utility Components listed above, ships can be customized even further by equipping them with Special Systems. These devices are intended to be offered primarily through other players who use their crafting skills to fabricate components of unique and intriguing capabilities:

- *Deceiver:* Used to fool scanners as to the contents of a ship's cargo hold.
- *Chameleon:* Provides the ability for a ship to project a different-sized radar return than what the ship actually is.
- *Worm:* Allows a character to attempt taking over the controls of an enemy ship by "hacking" its computer systems and cause it to do something from the following list:
 - a) Lower Shields
 - b) Stop Engines
 - c) Jettison Cargo
 - d) Cause Point Defense batteries to fire erratically
 - e) Shut down specific Crew Stations, or cause them to "reboot" (Display "Fatal Error" message, etc.)
- *Blinders:* EMP (Electromagnetic Pulse) Weapons that can temporarily disable scanners and other sensitive electronic devices
- *Firewall:* Reduces the chances of a *Worm* successfully hacking the ship computers.
- *Flasher:* Bright strobe that can be mounted on the back of ships that can momentarily blind pursuing pilots and Signature missiles.
- *Disguise:* Lightweight radar-absorbent foam that allows a ship to resemble a small asteroid or debris. Using weapons while a *Disguise* is applied will burn off the foam, but slow maneuvering (1/4 normal speed) won't affect it.
- *Fighter Pod:* These devices look like large Cargo Pods normally carried by heavy transports. However, they're capable of housing a Fighter or Light Fighter craft. This craft can be deployed in the same manner as a normal craft and flown by a player if the ship is attacked.

- *Gun Pod:* As with the Fighter Pod, the Gun Pod can house a variety of weapon types. When activated, the panels of the Gun Pod retract and the weapons are free to engage enemies.
- *Sniffer:* A small "bug" that attaches to a ship and transmits its location via a covert frequency.
- *Snuffer:* A device that can neutralize signals from a Sniffer or Snooper.
- *Snooper:* A device that attaches to a ship and transmits the intercom, radio, and navigation information of the ship to a receiving unit.
- *Spider:* A small robot that automatically moves about the ship searching for and destroying any *Sniffers* or *Snoopers*.
- *Spammer:* A transmitter that scrambles the intercom and radio traffic of the ship, and provides false navigation/navigation information to Sniffers and Snoopers.
- *Spoofers:* Device that bypasses the navigation controls of a ship and directs it towards a pre-programmed destination. (Can be counteracted by a Firewall.)
- *Possum:* A device that spoofs scanners into thinking a several ship systems are damaged.
- *Scan-Proofing:* System that prevents the content of cargo pods from being scanned. (This is obviously not used by legitimate transport services.)
- *Heisenberg Neutralizer:* Device that nullifies all quantum phenomena within a certain radius around the ship. This can also disable a variety of Alien Tech devices, Superstring Power Sources, Meson Guns, etc.
- *Big Bang:* System that temporarily channels all existing power into the Main Guns to quickly provide an overloaded salvo.

What kind of artistic customization can be performed?

Players can also take advantage of graphic methods to customize the appearance of their ships:

- *Paint:* A variety of different paint schemes will be provided, as well as the ability to hue the colors of each scheme to suit the player's tastes.
- *Logo:* Many different logos will be provided, allowing players to mix and match them to achieve the desired appearance. Multitexturing can be utilized to allow the players to mix and match logos to achieve a unique symbol.
- *Contrail Colors:* Players will be given the ability to change the colors of their engine contrails.
- *Kill Marks:* These symbols can be applied to a ship for all to view.
- *Light Signals:* Players can choose from several custom schemes for their ship's running lights.

Interface Design

All customization and outfitting can be performed through a fully three-dimensional interface that allows the player to quickly view the following:

- *Current Configuration:* This includes all equipped components and modules, as well as the amount and location of open mounting points. This also includes statistics such as Weight, Power Demand, etc.
- *Available Components:* This is an inventory of the Components and Modules available to mount on the ship.
- *Logo:* A close-up of the existing logo(s) on the ship.

Functionality:

- *Equip Component:* This simply involves clicking and dragging a Component from the inventory into a mounting area. If that area was already filled, the previous Component is

automatically moved to the inventory. All pertinent statistics for the ship will change based on the new component.

- *Clear Components:* Clears everything and starts with an empty hull.
- *Buy/Sell Components:* Opens a dialog box with the dealer of choice.
- *Change Logo:* Allows the player to choose and place the logo(s) of his choice.
- *Save/Load Configuration:* Allows particular Configurations to be quickly recalled and applied to a ship. (If the inventory can't support a configuration, that choice will be grayed out.)

Tasks

- Texture maps for the logos and paint schemes
- Geometry and textures for all hull types and components
- Generate code for alternate lighting schemes
- Generate code for component and module functionality
- Generate code and sound effects to support custom contrail colors
- Generate code for the customization and refitting interface
- Flight model changes to reflect the weight of different configurations

To Be Addressed

- Additional component and concepts
- Interface look and feel

Walkthrough

To be covered in this walkthrough:

- Weapons play an integral role in the game.
- What are the weapon types?
- How are the weapons used?
- How are the weapons targeted?
- How are the weapons fired?
- How does weapon damage work?

Weapons Play Experience

Garrett and Bill were two fairly experienced players who had set up a lucrative bounty-hunting partnership in the past few weeks. By equipping their two Fighters with a different range of weapons, they found they could come out ahead by taking using their weapons in concert with one another against an enemy's weak point.

With the White Sun pirate ring recently hitting the shipping lanes in the Ring Quadrant, there were a lot of players and companies offering big rewards to any player who could run the pirates out of the system. Garrett and Bill were on the job.

Bill had equipped his Fighter with Ceramic armor, two standard Mass Drivers, and two Ion Blasters, as well as a Bulk Pod full of Rockets with strongest "Shovel" warheads, and 4 Signature missiles with High Explosive warheads.

Garret had taken an opposite approach, mounting a single large Plasma Laser, a Heavy Cannon, 2 Radar missiles, and 1 W-guided Torpedo with a "Harsh" warhead onto his Torpedo-plated Heavy Fighter. Together, they hit the Jump Gate to the Ring Quadrant.

Ships - Weapons

Overview

The ship-based weapons in *Wing Commander Online* build upon the existing types in previous *Wing Commander* titles by extending the technology to support and enhance the multiplayer experience.

Vision and goals

GOALS

- ☐ Provide a wide range of unique and exciting weapon types
- ☐ Encourage different strategies and player interaction during combat

VISION

- ☐ Encourage and support a community
- ☐ Encourage a sense of player identity by allowing the player to equip his ship to support his combat skills and strategies

Walkthrough

To be covered in this document:

- Weapons play experience
- What are the weapon types?
- How are the weapons rated?
- How are the weapons targeted?
- How are the weapons fired?
- How does weapon damage work?

Weapons Play Experience:

Garrett and Bill were two fairly experienced players who had set up a lucrative little bounty-hunting partnership in the past few weeks. By equipping their two Fighters with a different range of weapons, they found they could come out ahead by taking using their weapons in concert with one another against an enemy's weak point.

With the White Sun pirate ring recently hitting the shipping lanes in the Bing Quadrant, there were a lot of players and companies offering big rewards to any player who could run the pirates out of the system. Garrett and Bill were on the job.

Bill had equipped his Fighter with Ceramic armor, two standard Mass-Drivers, and two Ion Blasters, as well as a Buzz Pod full of Rockets with cryogenic "Snowball" warheads, and 4 Signature missiles with High Explosive warheads.

Garret had taken an opposite approach, mounting a single large Prismatic Laser, a Meson Cannon, 2 Radar missiles, and 1 IR-guided Torpedo with a "Heater" warhead onto his Tungsten-plated Heavy Fighter. Together, they hit the Jump Gate to the Bing Quadrant.

They quickly encountered a tattered White Sun frigate salvaging the hulk of a freighter. As soon as the target appeared on their screens, they saw the frigate launch two Light Fighters towards them.

"Got 'em," said Bill, launching all his Signature missiles at the enemy fighters. One, then two hit the lead fighter, sending it tumbling through space a flaming hulk. The second fighter managed to dodge one missile, but the next missile impacted from the side. Damaged, but not destroyed, the enemy fighter tried to limp back to the frigate, but was gunned down by a single shot from Garrett's Meson Cannon.

"I'm in," Bill said, dodging defensive fire from the pirate ship as he sped toward it. Garrett dumped all his engine power into his weapons, and began pounding at the frigate with the Prismatic laser to soften the shields ahead of Bill. Before he could punch all the way through, however, two missiles headed his way and forced him to evade.

Though Bill's Mass-Drivers weren't nearly as effective against shields as Garrett's Prismatic Laser, he was able to punch through the soft spot his wingman left and score hits on the frigate's hull. At point-blank range, he skimmed over the larger ship, firing all of his rockets from the Buzz Pod underneath his fighter. The crackling sound he heard was the cryogenic liquid inside each rocket supercooling the frigate's armor.

Garrett had taken one direct hit in the tail from one of the missiles, but had spun around in time to see Bill's rockets hit the big ship. Knowing the supercooled section of the ship was now highly vulnerable to heat, he steamed in towards the target and launched the Torpedo at the last second to ensure it wouldn't be destroyed by defensive fire. His own fighter took several hits from the blazing turrets, but the flash on his canopy told him the Torpedo hit home. Wheeling around, he saw the spewing thermal jelly crack and splinter the ship's frozen armor, as secondary explosions began to spread.

"Cha-ching!" Bill said, as his Mission Objective Display changed to "Accomplished."

What are the Basic Weapon Types?

Mass-Drivers:

Mass Drivers use electromagnetic impulses to accelerate a projectile down a rifled tube at high speeds. Ammunition cost and storage capacities are issues with all Mass-Driver guns.

Mass Drivers can fire four different kind of projectiles/ammunition:

Flechette: Each shell disperses several small dart-like submunitions, creating a shotgun effect.

Armor Piercing: Uses a dense-metal rod at the core of the shell to penetrate armor plate.

High Explosive: Provides good all-around damage.

Flak: High-explosive rounds that detonate at a point in space after firing, peppering nearby targets with shrapnel.

Mass-Driver Sub-Types:

- *Vulcan:* This multi-barreled Mass Driver can achieve a very high rate of fire.

Strength: High rate of fire, well-suited for fighter combat

Weakness: Heavy, reduced reliability, high ammunition consumption, can't be equipped with a MAM system.

- **C-Gun:** This massive rail gun fires a dense-metal projectile at nearly the speed of light.

Strength: Massive hitting power

Weakness: Large size, slow rate of fire, high power consumption, recoil effects, range and damage effects greatly reduced when used in nebulae, gas clouds, comet tails, and Snow Blower clouds

- **G-Gun:** (Alien Tech) This enormous ultra-tech device launches point singularities at its target, ignoring shields and causing massive levels of structural damage.

Strength: Monstrously powerful, immune to all countermeasures and defenses.

Weakness: Singularity "missiles" aren't very maneuverable, huge size and power requirements, require Neutronium for ammunition.

Lasers:

Lasers direct collimated light of various frequencies and power to destroy a target. They're simple to build, require little maintenance, and are relatively inexpensive.

- **Prismatic Laser:** Designed to counteract Adaptive Shields, Prismatic Lasers perform as a regular laser against standard shields.

Strength: Good against Adaptive and Regular shields

Weakness: Costly

- **Very Long Range (VALOR) Pod:** These massive stationary lasers are autonomous, and can be set to fire on specific target types as soon as they enter a system.

Strength: Longest-ranged weapons in the game, very fast time of flight

Weakness: Huge size, costly, can only fire when stationary

- **Standard:** These lasers can be mounted on Fighters, Bombers and Defense Turrets, and come in a variety of Power Ratings.

Strength: Cheap, durable, versatile

Weakness: Can be nullified by advanced shields

- **Vulcan Laser:** An advanced laser with multiple rotating collimators and a very high rate of fire.

Strength: High rate of fire, inexpensive

Weakness: High power consumption

- **Homing Laser (Alien Tech):** This very rare laser weapon uses an unknown mechanism to actually guide the photon stream towards the target.

Strength: Extremely accurate

Weakness: Bolts can target friendlies as well as enemies

Particle Beams:

Particle Beams fire streams of helium nuclei to provide higher destructive power than a laser. Their bolts are significantly slower than lasers.

Strength: Stronger than lasers for the same power requirements.

Weakness: More expensive than lasers or mass drivers, slower bolts

Meson Cannon:

Meson cannons are high-power relatives of particle beams. Once fired, the meson bolt gathers quanta from the surrounding space once they've been fired, increasing their destructive power with no extra power demand. This peculiar effect makes them relatively weak at short ranges, but more powerful at long ranges, as the meson bolt has time to gather more energy. Meson bolts become unstable after they've traveled X meters and detonate in a small explosion. These require a high level of maintenance and are prohibitively expensive.

The Meson bolt quickly changes color from dull red to bright white as it gathers power.

Strength: Good stand-off weapon, explosive damage effects

Weakness: Expensive, weak at short ranges

Ion Blaster:

These weapons fire tiny "bullets" of charged plasma, capable of disrupting the electronic systems of their targets. Note that non-metallic armor types (Ceramic, Composite, Regenerative) prevent this effect, but can still be damaged by the heat of the "bullets" themselves.

(When an Ion Blaster has successfully affected the systems of its target, electrical discharges can be seen arcing around it.)

Strength: Good against Absorptive armor, can short out electronics of its targets

Weakness: Reduced effectiveness against non-metallic armor

Overloading Gun-Type Weapons:

Certain energy weapons and Mass Drivers can be "overloaded," firing with more power than their rated maximum. This is commonly called "Hot Loading," and can increase the damage capability of the weapon with the accompanying risk of causing it to malfunction or explode. Hot Loaded Mass Drivers provide more Kinetic damage and higher muzzle velocity, while energy weapons of course provide higher levels of Energy damage.

The degree to which a weapon can be Hot Loaded will depend upon its Tech Level and Quality Level. Higher Quality Levels allow the weapon to be Hot Loaded to greater degrees, as well as function longer under that condition. If it's determined that a weapon fails while firing under a Hot Loaded condition, it will either be completely ruined, or detonate like a warhead. The damage to the ship this detonation causes

depends on the size and location of the weapon, as well as the degree to which the weapon was being overloaded. Externally-mounted weapons tend to do less damage if they explode, while internal weapons can cause large amounts of damage if not destroying the ship outright.

Missiles and Torpedoes:

Missiles are weapons comprised of three components that dictate the combat performance of the system. The Motor section's Tech Level dictates the speed and range of the missile. The Guidance system generally dictates the agility and accuracy of the missile. The Warhead system dictates how much damage the missile does upon impact.

Guidance Types:

- *IFF*: Identify Friend or Foe guidance automatically locks onto the nearest enemy.

Strength: Very accurate

Weakness: Susceptible to Electronic Countermeasures and devices such as Chameleons

- *Infrared*: Automatically locks onto the strongest infrared source, such as hot engines. The missile can also be slaved onto a target the radar is tracking. Can track other missiles as long as their motors are burning. IR missiles are passive, and can't be detected by Radar Warning sensors. They can be detected by Missile Warning Radar systems, however.

Strength: Accurate, inexpensive, good for use against larger missiles such as Torpedoes and Plasma Warheads.

Weakness: Relatively short-ranged, very susceptible to ECM such as flares

Derivatives:

- *Fastball*: Small very high-speed missile with a dense-metal penetrator that's very effective against armor. These missiles emphasize speed over agility, and are poorly suited for use against Fighters.
- *Curveball*: Small, very agile missile that flies an erratic course to the target in an effort to defeat anti-missile batteries. Used primarily against large, slow-moving targets.
- *Lowball*: Slow-moving IR-guided stealthy missile that uses cold propellant to slowly sneak through a ship's shields by adjusting its relative motion to a few meters per second and then fire a self-forging penetrator into the armor.
- *Radar*: Locks onto the selected radar contact by the pilot, and uses active radar to track.

Strength: Range limited only by the firing craft's radar range. Requires active jamming to reliably spoof.

Weakness: Susceptible to Electronic Countermeasures.

- *Rockets:* Non-guided missiles useful for attacking large, slow-moving targets. Very cheap.

Strength: Cheap, fast, can't be jammed

Weakness: No guidance, significant dispersion between individual rockets leads to marginal accuracy.

Derivatives:

- *Buzz Pods:* These rocket pods are angled downward 45 degrees, allowing craft to fly over the target at high speed while firing.
- *Laser:* These missiles require a Laser Designator pod carried by either the launcher or another friendly ship. They are very difficult to jam.

Strength: Can hit specific sections of a ship, very difficult to jam

Weakness: Requires a laser designator for guidance

- *Signature:* Advanced missile that can be tuned to track only a specific target type.

Strength: Virtually impossible to jam by normal Electronic Countermeasures.

Weakness: Exceptionally expensive. Requires several seconds to "learn" the target before launch. Can be fooled by advanced Electronic Countermeasures.

- *Graviton:* This missile locks onto the strongest source of gravity/gravitons in the vicinity. Very useful against cloaked ships of Bomber or larger size.

Strength: Can track cloaked and stealth ships, difficult to jam in open space.

Weakness: Very short detection range, will break lock and attack the largest gravitational source.

Mines:

- *Captor:* Launches missiles when a target comes within range.

Strength: Can be equipped with a variety of missile types.

Weakness: Expensive, limited ammunition capacity

- *Laser:* Fires lasers when a target comes within range.

Strength: Inexpensive defensive system

Weakness: Relatively short range and low power

- *Proximity:* Detonates when any target comes within range.

Strength: Difficult to detect, and can be equipped with a variety of warheads

Weakness: Target must pass very close to be damaged

- *Signature: Detonates when a specific target or target type comes within range.*

Strength: Difficult to detect, and can be equipped with a variety of warheads

Weakness: Can be deceived by a number of Electronic Countermeasures

- *Cluster Mines (Caltrops):* These weapons are intended primarily for use on Fighters and Bombers. They're ejected in groups of approximately 24 and immediately spread out to cover an area of roughly 50 cubic meters. Once a ship flies through this area, all the mines detonate.

Strength: Inexpensive method to prevent close pursuit

Weakness: Requires considerable storage space, inflicts minimal damage overall

Warheads:

- *Snowball:* Cryogenic liquid containers designed to shatter armor and external weapon systems.

Snowball warheads are very effective against hot externally mounted guns, where they can cause an instant Failure condition if they strike the craft (physically hit the craft once the shields are down.)

Snowball warheads can also be used in tandem with energy weapons. Most Armor types struck by a Snowball warhead become extremely brittle. Hitting that Armor section with an Energy or Mass Driver weapon can cause that entire section of Armor to shatter. (The Armor will gradually warm back up over a period of **X** seconds. Until then, it will display as blue to show the pilot the Armor has been super-cooled.)

Snowball warheads are useless against shields

Strength: Can make armor vulnerable to subsequent fire

Weakness: Causes no damage to shields

- *Spitball:* Acid containers designed to dissolve armor.

Strength: Effective against most armor types

Weakness: Causes no damage to shields

- *Heater:* Thermal-charge warhead designed to quickly burn through armor. These warheads are very effective when used just before or after a Snowball warhead impacts the target.

Strength: Can burn through many types of armor

Weakness: Causes minimal damage to shields

- *Plasma:* Creates a short-lived but very large sphere of superheated plasma capable of devastating even the strongest shields. The Plasma sphere envelops

the target on contact, dividing its damage power equally across all the targets Shield and Armor quadrants.

Strength: Can short out electronics, and cause high levels of damage to shields and armor. Damages all quadrants of the target on contact.

Weakness: The plasma sphere only lasts for X seconds before dissipating. The missile is relatively slow and can be intercepted by other missiles. Poorly suited for use against fighters.

- *Armor-Piercing:* Shaped-charge warhead that detonates on contact.

Strength: Highly effective against armor, and effective against shields

Weakness: Very small damage radius, no fragmentation damage

- *Fragmentation:* Distributes shrapnel capable of damaging lightly armored targets in a radius variable with the size of the warhead. The increased damage radius of these warheads makes them very useful against concentrations of lightly armored targets.

Strength: Large damage radius

Weakness: Only effective against lightly shielded and armored targets unless it scores a direct hit.

- *Virus:* This warhead is designed expressly to attack living organic armor types such as *Regenerative*. Once a Virus warhead has impacted living armor it will "kill" the armor at a rate of X% per second.

Strength: Devastating against living armor

Weakness: Worthless against shields and other armor types

- *Quantum:* Very expensive explosive warhead designed primarily to attack capital ships and large targets.
- *Singularity:* More powerful than the Quantum warhead.
- *Chaos:* More powerful than the Singularity warhead.
- *Antimatter:* These warheads go to "11."

Special Weapon Support Systems:

- *Modular Missile System:* Internal Missile Launcher system that can mix and match different warheads and guidance systems and match them to the motors in a matter of seconds. Adds versatility, but can only be mounted on Heavy Fighters or larger ships.
- *Cold Launcher:* This system uses compressed nitrogen to blow the missiles out of the tubes at a high rate of speed. The missiles won't ignite their motors until they're well away from the launcher, reducing the chance the launching ship can give away its position.
- *Stasis Field Generator:* A variation of the Leech technology, this device can immobilize up to Bomber-sized targets within a certain radius.

Strength:

Weakness:

- *Tractor Beams:* These can be used to attract or repulse a target without harming it. Targets can be moved into harm's way, however.

- *Multi-ammunition Magazine (MAM):* The MAM system allows the player to select one of three different types of ammunition for a Mass Driver weapon. This system provides greater versatility to the guns, but at with considerably higher cost and weight. The system is also somewhat prone to jamming.

The MAM system can't be used on *Vulcan* mass drivers.

Systems Design

How are the Weapons Rated?

Weapons are rated in a wide variety of categories, reinforcing the customizability and "Rock, Scissors, Paper" aspect present in *Wing Commander Online's* combat system.

Tech Level: How technically advanced a weapon is.

Quality Level: How well constructed it is.

Size: How much space on a ship it requires.

Weight: How much it weighs.

Rate of Fire: How many rounds per second it can fire.

System Dependency: What systems it needs to fire.

Minimum Range: The shortest range it can engage a target.

Maximum Range: The farthest range it can engage a target.

Maneuverability: How agile the projectile is.

Damage Rating: How much damage potential the weapon has.

Damage Type: What kind of damage the weapon can inflict.

Speed: The speed of the projectile.

Guidance: The type of guidance the projectile uses.

Damage Radius: How big an explosion the projectile creates.

Radius Modifier: How much damage the explosion can do.

How are the Weapons Targeted?

The target selection, lock, and Computer-Controlled Impact Point symbology from the previous *Wing Commander* titles should remain fundamentally unchanged for these weapons.

In an effort to enhance accessibility further, a "Smart Targeting" system could be provided, automatically selecting the most appropriate weapon for a given target type and range. This would allow the player to focus on the fight itself without having to constantly cycle through weapon types on his own to achieve the best effect.

How are Weapons Fired?

Requirements to Fire a Gun Weapon:

- Sufficient Power
- Ammunition remaining (in the case of Mass-Drivers)

Requirements to Fire a Guided Missile:

- Functional targeting system

- Target lock
- Missiles remaining to fire

Requirements to Fire a Rocket:

- Rockets remaining to fire

Requirements to Deploy a Mine:

- Mines remaining to deploy

Requirements to Use a Tractor Beam:

- Sufficient Power
- Target Lock
- Equal or greater available Engine Thrust than the target

How does Weapon Damage work?

Weapons are rated in a variety of different Damage Types. Once a successful hit is registered, the Damage Type of the Weapon is compared to the Shield and Armor Type of the target. The resulting Damage Modifiers are applied to resolve how much, and what type of damage was inflicted.

Interface Design

The weapon interface is virtually self-explanatory, relying on one button or key to select a weapon type, and another button to actually fire it.

To "Overload" a weapon capable of supporting that function, the player would use a specific key to begin the overcharging process. An indicator in the cockpit would show the charge level for that weapon, and a warning would sound if it's determined the weapon would fail if the process continues.

Tasks

- Art generated for the targeting symbology, weapons effects, and explosions
- Code generated for tracking logic, damage resolution, and power consumption
- Sound effects for all weapons and explosions
- Voice prompts and input sound effects for weapons use

To Be Addressed

- *Weapons damage effects*
- *Graphic representations for different weapon damage types*
- *Play balancing for weapons performance parameters*

Walkthrough

Here's what's covered:

- *Shields - Function and Properties*
- *Shield power management*
- *Shield Primary Properties and Enhancement Recipes*
- *Additional Shield Types*
- *Shields art notes*
- *Shields UI notes*

Ships - Defenses - Shields

Overview

Shields are the mystical sci-fi convention that protects ships from harm with an energy field capable of diffusing energy and absorbing kinetic damage. Every space game worth its salt has approached the task of making shield effects, but until recently graphics limitations have undermined any game's ability to realize this properly. The gameplay, crafting, visual, and aural properties of shields in *WCO - Privateer* is covered in this document.

Vision and goals

Goals

- ☐ Keep the function simple and the form deep
- ☐ Encourage player crafting
- ☐ Encourage multiple shield strategies
- ☐ Simplify the control scheme
- ☐ Redefine and set a new watermark for shield effects

Vision Points

- ☐ ... should have an intuitive user interface
- ☐ ... should have a sense of discovery
- ☐ ... must have a presentation that blows people away at first glance
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... will have a technology tree that unfolds over time
- ☐ ... will make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ ... will offer tools for player-driven content
- ☐ ... will provide privileges to experienced players
- ☐ ... must support many different styles of play
- ☐ ... should have big visual payoffs
- ☐ ... should support character building and advancement
- ☐ ... should have little touches that add depth
- ☐ ... must encourage and support a community

Walkthrough

Here's what's covered:

- Shields - Function and Properties
- Shield power management
- Shield Primary Properties and Schematic Recipe
- Additional Shield Types
- Shields art notes
- Shields sfx notes

Additionally the Primary Properties of each system (for purposes of ship installation and any minimum requirements) and the Schematic Recipe of necessary components (ingredients) for crafting the item are listed.

Gameplay Example

Jimmy knew that sound. A crackling thud and hiss echo through his ship as a Vulcan cannon hurtles hundreds of rounds into the aft shield. Jimmy checks the shield indicator and notices a distinct drop in aft power.

He doesn't have time to worry about that right now.

Locked in his ITTS gauge is a heavy pirate fighter. He fires a twin stream of plasma as he empties his energy battery into his dual plasma beams. They strike the pirate's shields hard, sending a wash of yellow across a swirling blue surface of compensating energy patterns.

The beams are too much for the shields, melting them away like a blowtorch on plastic then scorching the hull. The hull heats to white-hot and breaches, cracking the enemy ship into arcing debris wrapped in red embers and electrical tendrils.

Jimmy's missile lock warning gauge pops up. An IR is rapidly closing from an initial range of 400k. He reroutes shield energy to compensate for the Vulcan induced loss and turns his concentration on his troublesome second enemy.

Energy banks only half recharged, Jimmy locks his enemy in his sights as the IR hits home. The missile's approach and impact sounds like a drunk running through a waterfall then slamming into a brick wall. He briefly loses the target, but quickly reacquires.

Deciding to save energy for a later burst, he unloads his own Vulcan cannon into the side of the desperately maneuvering enemy. The pirate's shields distort as it absorbs the kinetic power. Curling blue and white spikes radiate from the point of impact and dance off the edges of the shield like slowly retracting claws scratching the surface of a translucent pearl...

Systems design

Shields - Function and Properties

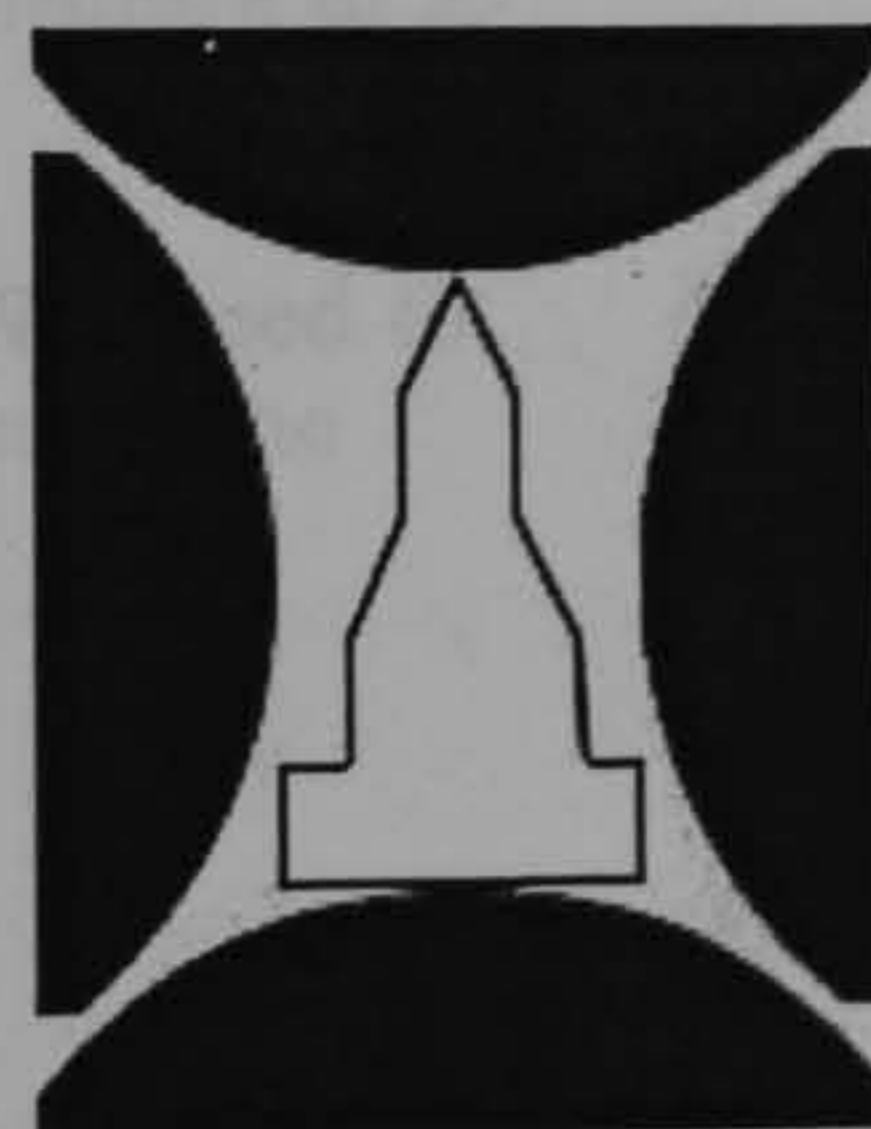
Shields are what keep Wing Commander from being a very short engagement game.

Shields are indicated on the HUD along with armor/hull integrity.

My crappy illustration shows the typical layout.

Shields are the frontline of defense. Each of the four quadrants around a ship can absorb a number of damage points before failing. They can then regenerate.

All the average player needs to know is that the better the shield package available for their ship configuration, the more damage the shields will take before their ship goes kerplooeey. Other than power management, they need never dig any deeper. For the gear heads there are the following tidbits.



Shield strength (hit points) is determined by the power capacity of the shield battery. Power units in WC are in Gigajoules (Gj)

This can be a ratio of Gj to shield HP or a one to one Gj/HP (TBD).

Once the shields are penetrated any subsequent hits go straight to the armor, once through the armor the ship is destroyed. You must disable the shields to a given side of the ship before you can impact the armor.

Shields regenerate in relation to the recharge rate of the shield battery. Recharge rate is improved by better electronics packages.

Shields diffuse and disperse energy damage (energy/beam weapons) across its surface. They absorb kinetic damage (warhead, mass driver, bullets). A given shields efficiency in absorbing or diffusing these types of damage results in a percentage decrease to the damage points the shields suffer. Some shields are better at absorbing kinetic damage than energy damage and vice versa. All standard shield packages (not player crafted) have kinetic and energy damage resistance percentages of zero.

Higher tech level player crafted software packages, emitters, and electronics packages potentially modify these resistance percentages cumulatively. This is a random egg for a crafter. Dependent on the crafter level of the components and tools/facilities used to build, a resistance bonus against one or both of the damage types may be added. This is and other crafter bonuses are covered in the Crafting doc.

Ex: Jimmy just bought a new shield package with a cumulative tech level of 4. He knows it will recharge a little more slowly than his last package due to its increased power capacity. However it has a more efficient recharge rating, which means it's better than just his current system with a larger battery slapped in. He couldn't care less about that. What really have him excited are the bonuses! It has a 20% resistance to energy damage and a 26% resistance against kinetic damage! The player selling this doesn't know how much its worth and just accepted a bid from Jimmy that is ridiculously low. Jimmy grins into his screen and prepares to give that bastard ScoobyNoodle a rematch.

Capital Ships and other large craft with more than three slots for primary batteries (shield, energy, thruster) can potentially have more than one shield battery active with the aid of a Shield Battery Coupling. This takes one additional systems slot, properties TBD.

Shield Power Management

Shield power management has not been an active component of WC. As an interface concern it has always been problematic. Although we want to explore the potential of power management in WCO - Privateer we want to avoid the key press nightmare of X-Wing.

This is a tricky, tricky, tricky task for development ("Speed Racer" tricky). We need to develop a very simple interface to allow the player to possibly manage power in the following ways:

- Redirect % shield power from three sections to one section.
- Redirect % shield power from two sections to one.
- Redirect % shield power from one section to two.
- Redirect % shield power from one section to one.
- Equalize shield power
- Toggle Auto-Equalize on/off

Shield Primary Properties and Schematic Recipe

Primary Properties

- **Size:** There are two sizes, small and large or standard fighter and cap ship.
- **Weight:** Factors into overall ship speed and handling
- **Power Capacity:** In Gj, determines shield hit points
- **Recharge Rate:** up to x Gj per second bottleneck from reactor
- **Energy Damage Resistance:** up to 40%
- **Kinetic Damage Resistance:** up to 40%.

Schematic Recipe

- **Housing** – There are large and small shield emitter housings. Large or small housing and structural material (materials TBD) used to craft combined with battery determines weight.
- **Software Package:** (packages TBD) Package level can increase energy and/or kinetic damage resistance percentage to a maximum of 10%.
- **Electronics Package:** (packages TBD) Package level can increase the base recharge rate by up to 25%.
- **Emitter:** tech (crafter) level can increase energy and/or kinetic damage resistance percentage by up to 10%. Can also increase the base recharge rate by up to 25%
- **Battery:** determines power capacity and adds to weight.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase energy and/or kinetic damage resistance percentage by up to 15%.
- **Tools/Facilities that are required to construct** (TBD) Tech level of facilities or tools used can increase energy and/or kinetic damage resistance percentage by up to 5%.

Additional Shield Types

TBD we are exploring the option of having different special shield types including smart shields, adaptive shields, etc. This doc will be revisited as soon as the design team can meet on the topic.

Art Notes

A good space combat game puts emphasis on the quality of its shield effect. Here are some visualization suggestions for ours.

Think of shields as having the consistency of a latex balloon with magnetic powder worked into its composition.

Now, think of energy weapons as high-powered streams of water diluted with iron filings. Bolt weapons are squirts and beam weapons are streams.

As the stream hits the balloon, the water disperses around the balloon with a majority spraying off the impact point and the filings forming temporary patterns against the magnetic powder that are washed away.

Now consider that the water is heated and colored (yellow, red, green, etc.) and the filings are colored in a deeper hue. The balloon skin acts like those heat sensitive fabrics that distort with a swimming oil slick look as heat is applied to them. So what you get is the primary color of the water swimming over the balloon surface as it

shimmers with an iridescent pattern of electrical hues and lingering patterns of the deeper colored filings.

As the stream or bolts wear down the balloon the surface gradually retracts until it bursts. This gives a localized flash of electricity and energy (energy colored the same as the weapon used against the shield. To see shields melt and fizzle would be great. Another way of thinking of this effect is in the gameplay example that reads: Like a blowtorch on plastic.

Okay...sticking with my kooky balloon metaphor.

Kinetic weapons are either rocks/pebbles or sticks (guns vs. missiles). The dream effect would be to actually see the shield surface deform inwards from the force of the kinetic impact.

The pellets and/or toothpicks have to breach the shield to do any damage to the armor. As they hammer at one section causing the shield to bend electrical tendrils leap out in a circle from the point of impact and curls back over to reconnect to the shield. As over the top as the gameplay example was, I think it makes sense... Curling blue and white spikes radiate from the point of impact and dance of the edges of the shield like slowly retracting claws scratching the surface of a translucent pearl...

It's understandable (yet incredibly depressing) if we can't accomplish the deforming shield mesh...But the electrical claws should be acceptable on their own...

Just some idears...

Sound Notes

Typical shield hits have usually come in three flavors. Shield strong, shield medium, shield weak gets hit sounds. Let's try and come up with something more.

Think of each component involved in shield generation and how they would react to stress.

Of course we wish to create in-cockpit and outside cockpit versions of these sounds. The acoustics of several tones of steel chassis will be neat to consider.

First, the interior mechanics...The shields themselves are generated by an emitter that draws power from a battery. As shields are hit you hear a variable tone that can range from 1khz all the way down to 100hz. As the shields attempt to absorb or diffuse the damage hitting them this ramps up to its max frequency (dictated by max available shield power) and descends to its lowest frequency before breach.

Then there is the shield itself. Sweeping pulses and crackling energy combined with the shield hit sound of the specific weapon should give players an aural clue as to what type of weapon is hitting them.

These energy sounds should feel liquid. Like an electrical soap bubble that drips electric goo as its stability is compromised. When breached there is a rushing static whoosh.

The target of damage will hear from within cockpit both the mechanical sounds and the shield sounds muffled and echoed by the hollow chassis they are reverberated through.

Exterior camera or enemy ships will not hear the mechanical portion of this sound, only the shield specific noises.

Anywho...I have some other ideas about this I'm willing to discuss.

Interface design

Controls

- Shield Power Management – TBD

Art

- Shield energy hit art
- Shield energy hit breach
- Shield kinetic hit art
- Shield kinetic hit breach
- Shield got hit, high energy remaining, swirl
- Shield got hit, medium energy remaining, swirl
- Shield got hit, low energy remaining, swirl
- Shield HUD/Cockpit display
- Power management pop-up

SFX

- Shield kinetic hit interior – high energy remaining
- Shield kinetic hit interior – med energy remaining
- Shield kinetic hit interior – low energy remaining
- Shield kinetic hit breach interior
- Shield energy hit exterior – high energy remaining
- Shield energy hit exterior – med energy remaining
- Shield energy hit exterior – low energy remaining
- Shield energy hit breach exterior
- Klaxon – Shield(s) at 25%
- Klaxon – Shield(s) at 0

Tasks

TBD

To be addressed

- Power Management Control scheme
- Additional Shield Types

Ships - Defenses - Armor

Overview

Armor consists of sections of various materials applied to a ship's vital areas to provide protection in the event the shields are breached. The ship-based armor in *Wing Commander Online* build on the previous *Wing Commander* approach by extending the technology to support and enhance the multiplayer experience. By offering many different armor types, each with its own balance of cost, weight, and types of protection, we provide the player another layer of depth for combat, customization, and identity.

Vision and goals

GOALS

- ☐ Provide a wide range of unique and exciting armor types
- ☐ Encourage different combat strategies based on these armor types

VISION

- ☐ Encourage and support a community surrounding the space combat play mechanic
- ☐ Encourage a sense of player identity by allowing the player to equip his ship to support his combat skills and strategies

Walkthrough

To be covered in this document:

- Armor play experience
- What are the armor types?
- How are the different armor types rated?
- How does armor damage work?

Weapons Play Experience

Most folks on the server had tried to take on the mercenary Derrick at one time or another, and most folks lost. He always chose his fights well, and was never to be found when a Confed patrol went looking for him.

Barry knew that taking out this Top Gun would give his Prestige a serious shot in the arm, and bring in the high-paying missions he was after. Barry had paid players of all types for tips on where Derrick was hiding, and almost fell out of his chair when he got a Message from a friend pinpointing Derrick's location one jump away.

"Be careful, though," his friend said. "He's got some kind of funky, shiny armor. It kept thickening itself before I could punch through!"

Quicksilver Armor? Barry thought to himself. *That's why Derrick is so tough!*

Only a few days before Barry had come across another player who'd been crafting just that kind of armor, but had a tough time finding the exotic raw material. The one word of caution the craftsman had for Barry was that the Quicksilver armor was amazing, but vulnerable to extreme cold.

Barry immediately bought several cryogenic "Snowball" warheads for his Signature missiles and headed out to find Derrick.

The fight was almost anticlimactic. Overconfident in his armor, Derrick made a head-on charge towards Barry. Unconcerned as a rocket salvo and several Meson Cannon blasts depleted his shields, he didn't know that Barry had immediately followed up with several Snowballs. Within seconds, the mercenary's ship had lost all of its armor, its sensors, and one engine as Barry crowed with delight. With a growl of disgust Derrick surrendered the fight, and Barry streaked home with a handful of salvage and an entirely new reputation to bask within.

What are the Basic Armor Types?

Armor can be salvaged from hulks, or Crafted from different raw materials. (See *Crafting.doc* for more information on the fabrication process.) Each different type of armor has advantages and disadvantages, allowing the player to further refine the capabilities of his ship to suit his finances and combat strategies.

- **Aluminum:** Light, inexpensive armor that represents the lowest level of protection.

Strength: Lightweight

Weakness: Marginally effective

- **Steel:** Heavier and stronger than Aluminum, and the least expensive armor type.

Strength: Cheap

Weakness: Heavier than Aluminum

- **Titanium:** Nearly as light as Aluminum, and as strong as Steel. Much more expensive than either.

Strength: Light and strong

Weakness: Expensive

- **Composite:** Lightweight, very strong armor made from glass, carbon, and metal fibers suspended in layers of polymer matrices.

Strength: Lightweight

Weakness: None

- **Ceramic:** Moderately heavy and expensive armor designed for high levels of protection from energy weapons.

Strength: Excellent protection against Energy weapons

Weakness: Vulnerable to Kinetic damage

- **Tungsten:** Very heavy and exceptionally tough metallic armor.

Strength: Very good all-around protection, particularly against Energy and Explosive damage

Weakness: Heavy, expensive

- **Silksteel:** Exotic, lightweight alloy composed of interwoven microscopic strands of metal and polymer materials.

Strength: Light and durable

Weakness: None

- **Tantalum:** Very heavy, and one of the hardest non-radioactive metals known.

Strength: Very good all-around protection

Weakness: Very heavy, expensive

- **Reflective:** Mirrored armor designed to reflect beams from lasers and other energy weapons.

Strength: Excellent protection against energy weapons

Weakness: Vulnerable to Kinetic energy

- **Regenerative:** Living organic armor that can "heal" itself. This armor can't be reproduced by human technology. (This armor can be a resource gathered from a certain type of Space Monster.)

Strength: Self-repair

Weakness: Vulnerable to acids and viruses

- **Hydro-ferrite:** Alien-forged metallic/crystalline hydrogen. This armor can't be reproduced by human technology.

Strength: Virtually weightless, stronger than Titanium

Weakness: Only found in some alien derelicts, can't be repaired

- **Neutronium:** Ultra-dense material from dead stars.

Strength: Unrivaled protection against Kinetic damage

Weakness: Incredibly heavy, very rare, requires weeks to forge

- **Absorptive:** High-tech armor that can shunt a percentage of the energy that strikes it back into the Power Supply of the target. This armor can't be reproduced by human technology.

Strength: Absorbs power from energy weapons.

Weakness: Very rare, can't be repaired.

- **Quicksilver:** Ultra-tech liquid metal armor system of unknown origin that automatically thickens itself in the quadrant where the shields are weakest. This armor is very light but exceedingly effective against virtually all weapons except

Snowball warheads. It acts as *Reflective* armor against Lasers. This armor can't be reproduced by human technology.

Strength: Allocates extra protection automatically where the shields are weakest. Excellent protection against Lasers via reflection.

Weakness: Very expensive, can't be repaired by any human facility.

- **Fullerene:** This substance is constructed of interwoven threads of C₆₀ molecules with a single gold atom at the core of each. This combines the exceptional strength and heat-absorbing capacity of the carbon structures with the conduction and shielding properties of gold.

Strength: Excellent against most weapon types, and provides protection against the system-disabling effects of *Ion Blasters* and *EMP* weapons such as *Blinders*. Completely impervious to *Spitball* and *Heater* warheads.

Weaknesses: The most expensive human-constructed armor in the game. Requires the highest tech level facilities and engineer skills to construct or repair. Very heavy.

Though Armor requires no power in order to offer protection to the ship, it does add a significant amount of unsprung weight that will reduce maneuverability and acceleration performance. Armor also costs money to install and repair, making it an item some players may elect against using.

Systems Design

How are the Armor Types Rated?

Armor is rated in a wide variety of categories, reinforcing the customizability and "Rock, Scissors, Paper" aspect present in *Wing Commander Online's* combat system:

Cost: The base cost range

Laser: The estimated vulnerability to Laser weapons

Particle: The estimated vulnerability to Particle weapons

Kinetic: The estimated vulnerability to Kinetic weapons

Heat: The estimated vulnerability to Heat and Shaped-Charge weapons

Cold: The estimated vulnerability to Snowball-armed/cryogenic weapons

Explosive: The estimated vulnerability to Explosions

Acid: The estimated vulnerability to Acid-armed weapons

Virus: The estimated vulnerability to Virus-armed weapons

Weight: The base weight range

The following table is a compilation of the current armor types for the game, and their characteristics.

	<u>Cost</u>	<u>Laser</u>	<u>Particle</u>	<u>Kinetic</u>	<u>Heat</u>	<u>Cold</u>	<u>Explosive</u>	<u>Acid</u>	<u>Virus</u>	<u>Weight</u>
Aluminum	Low	Poor	Poor	Poor	Poor	Good	Poor	Good	Excellent	Low
Steel	Lowest	Fair	Fair	Fair	Fair	Good	Fair	Poor	Excellent	Medium
Titanium	High	Good	Fair	Fair	Good	Fair	Fair	Good	Excellent	Low
Composite	Medium	Fair	Fair	Good	Good	Good	Fair	Poor	Excellent	High
Ceramic	Medium	Excellent	Fair	Fair	Excellent	Fair	Good	Good	Excellent	Medium
Tungsten	High	Good	Good	Good	Good	Fair	Good	Fair	Excellent	Medium
Silksteel	High	Good	Good	Very Good	Fair	Fair	Good	Poor	Excellent	Low
Tantalum	High	Very Good	Very Good	Very Good	Very Good	Fair	Very Good	Fair	Excellent	High
Reactive	High	Poor	Poor	Poor	Poor	Good	Excellent	Poor	Excellent	Medium
Reflective	Very High	Best	Good	Poor	Fair	Poor	Poor	Excellent	Excellent	Medium
Regenerative	(Alien Tech)	Good	Excellent	Excellent	Good	Good	Fair	Poor	Poor	Medium
Hydrite	(Alien Tech)	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Fair	Excellent	Lowest
Neutronium	(Alien Tech)	Best	Best	Best	Best	Best	Best	Best	Excellent	Best
Absorptive	(Alien Tech)	Good*	Good*	Fair	Fair	Poor	Fair	Poor	Excellent	Medium
Quicksilver	Extravagant	Excellent	Very Good	Very Good	Very Good	Poor	Very Good	Very Good	Excellent	Low
Fullerene	Extravagant	Excellent	Very Good	Good	Excellent	Good	Good	Excellent	Excellent	Medium

How is Armor Equipped?

Armor is equipped in the exact same manner as any other ship's component--through the Ship Customization Screen. Armor is equipped on a whole-ship basis, with the cost directly dependent on the size of the ship to be outfitted.

How does Armor Damage Work?

Armor absorbs *Damage Points* from a weapon impact. Each impact decrements the remaining armor material in the appropriate quadrant, using the *Damage Point* rating of the Weapon multiplied by the *Damage Modifier* of the Armor material:

$$\text{Incurred Damage} = \text{Weapon Damage Points} \times \text{Armor Damage Modifier}$$

Hit location and resultant damage is resolved in Quadrants: Forward, Aft, Left, and Right. Shields and Armor are rated against the various types of weapons damage mechanisms with the following scale based on the Shield and Armor *Damage Modifier*:

- Poor*: 0-20% of the damage potential is absorbed.
- Fair*: 20-40% of the damage potential is absorbed.
- Good*: 40-60% of the damage potential is absorbed.
- Very Good*: 60-80% of the damage potential is absorbed.
- Excellent*: 80-100% of the damage potential is absorbed.
- Best*: 90-100% of the damage potential is absorbed.

Damage that's not absorbed by Shields will go through to Armor. Damage that's not absorbed by Armor will go through to the ship's ship components and systems.

Leak-Through Damage:

As Armor becomes worn down, there's an increasing chance that systems in the affected quadrant can be damaged with each successive hit, reflecting the fact that there may be holes blown in the Armor plate directly in front of that system. The following table compares the Armor strength remaining to the chance for Leak-Through Damage.

- 90-100%: No chance for Leak-Through damage.
- 70-90%: 10% chance of Leak-Through damage.
- 50-70%: 25% chance of Leak-Through damage.
- 30-50%: 40% chance of Leak-Through damage.

10-30%: 60% chance of Leak-Through damage.
1-10%: 90% chance of Leak-Through damage.

Note: As armor sections become damaged, there's an increasing chance that a specific system located behind that armor section will be damaged with each weapons hit.

Interface Design

Once armor is equipped, it requires no player input to operate or maintain, aside from *Repairing* it at a ship repair or construction facility.

Tasks

- Texture maps generated to represent the different armor types
- Code generated for armor damage resolution
- Sound effects for all different armor types when hit by a weapon
- Flight model changes to reflect the weight of different armor types

To Be Addressed

- Armor damage effects/pyrotechnics
- Play balancing to finalize armor performance parameters
- Developing more armor types

Ships - Defenses - Missile Defenses

Overview

Missile combat is a popular aspect of Wing Commander. Defending against missile attacks in Wing Commander is a key gameplay component and really separates the chaff from the wheat. The systems depth needs to be in place for the hard core crafter or pilot, but the defenses themselves (in actual use) must be as automated as possible for the average user. This document covers the varied defense systems and tactics a pilot can utilize to evade a primed warhead speeding at them at 600-1000kps.

Vision and goals

Goals:

- ☐ Keep the function simple, but the form deep.
- ☐ Provide multiple ways to get out of a situation
- ☐ Make as fully automated as possible
- ☐ What isn't auto should use a minimal number of controls.

Vision Points:

- ☐ ... must be easy to get into, as a new player and as an advanced one
- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... will have a technology tree that unfolds over time
- ☐ ... will make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ ... must support many different styles of play
- ☐ ... should have little touches that add depth
- ☐ ... should have secrets and lore to it

Walkthrough

This document covers the varied non-pilot methods of missile defense including:

- Missile Decoys
- ECM unit
- Anti-Missile Cannon
- ECCM unit
- Other Missile Defense tactics

Additionally the Primary Properties of each system (for purposes of ship installation and any minimum requirements) and the Schematic Recipe of necessary components (ingredients) for crafting the item are listed.

Gameplay Example

Jimmy has a pretty damn good heavy transport. He's been building it up for six months, why shouldn't he? Outfitted with anti-missile cannons on each of his four turrets, a level 2 ECM, and dual Decoy Pod Launchers, he's prepared for action.

He enters a system on his way to his client, delivering a load of structural materials needed for housing and chassis construction. A band of raiders take the opportunity to ambush him as he heads for the next jump point.

Nervous at first, he smiles after targeting each of the three craft. Light fighters with weak weapon load outs. He comms them, "You dumb bastards have a chance to get out of my way. Attack me and I'll fry you." At this point each of the fighters launches a mixture of Heat Seeker, Image Rec, and Friend and Foe missiles. He configured his ECM to automatically control his decoy deployment and it hums to life, showing the existing ECM charge and the missile lock showing his decoy status and current missiles locked onto him.

His ECM is able to throw off the Friend or Foe missiles, and he manages to fool the sensors on the Image Recs with his Distortion Projector decoys...but the heat seekers make it through. His anti-missile cannons whine into being and spray a barrage of projectiles taking out the missiles before they impact on his shields.

He then turns his attention towards frying these jerks. He manages to ice one of them with his automated turret guns, and a second with a couple of image recs. The third escapes his wrath and circles around to launch another salvo of missiles.

As the missile lock indicator lights up showing two friend or foe missiles, his ECM clicks into action. Then a warning klaxon sounds and he notices his ECM is jammed! The pirate as an ECCM unit! The two friend or foes blast away his rear shields and do light armor damage.

He sends a comm to the player "I'm gonna make you pay for that. And thanks for letting me know I have to upgrade my ECM software package. Happy Cloning!"

Systems design

Standard Disclaimer: This and other ship defense and offense docs will contain a high level of detail about how the system is crafted, the components needed, and the relationships to efficiency. The casual player will never have to worry about these things. Just as I don't care about the specific guts of my car, it just works and that's enough for me. But for those gear heads that want the best hot rod, these details are essential.

Warning Signs

All craft equipped with even the most basic systems has a **missile lock** indicator. This software can be upgraded to provide more information and functionality. Crafting Note: Upgrade availability could be tied to either Radar or Targeting system tech levels.

- **Missile Alert Level 1 (all ships):** A simple pop-up box or cockpit light that says MISSILE LOCK with warning klaxon.

MISSILE LOCK

Art Note: This pop up need not be language dependent. The words are only to illustrate the needed info in each consecutive instance of missile alert requires.

- **Missile Alert Level 2:** Warning Klaxon and Pop-up box that contains MISSILE LOCK, Ids the Missile Type and number of decoys or ECM charge remaining (or flashing status if jammed/depleted with associated warning klaxon).

MISSILE LOCK
Image Rec
12 IR Decov

MISSILE LOCK
Friend or Foe
ECM: JAMMED

MISSILE LOCK
Friend or Foe
ECM: DEPLETED

MISSILE LOCK
Friend or Foe
ECM: ██████████

- **Missile Alert Level 3:** Warning Klaxon and Pop-up box that contains MISSILE LOCK, Ids the Missile Type and number of specific decoys for type or ECM charge remaining (or status if jammed), SFX for proximity range counter and option to immediately target last detected incoming missile.

MISSILE LOCK
Image Rec
12 IR Decoy
Range: 600k

MISSILE LOCK
Heat Seeker
12 HS Decoy
Range: 600k

Primary Properties:

- **Required System(s):** Targeting Computer of x level and/or Radar System of x level. Levels for both TBD

Schematic Recipe

- **Software Package:** (packages TBD) One for each warning level.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build (TBD).
- **Tools/Facilities that are required to construct** (TBD) In this case a simple development workstation would suffice.

Decoys

Decoys are handled by a Decoy Pod Launcher (TBD) and are manually launched by the player, unless they have an actively configured ECM unit (see ECM Systems below). Dependent on the last missile type locked on, the appropriate decoy pod is deployed or none are launched if the player does not have a counter for the missile type (this is independent of missile lock indicator level). After a decoy is launched it lasts x seconds (dependent on battery efficiency) before burning out and being deleted from the world. During this time it pulses every ½ second and any missile (locked onto the player only) within its range volume can has a base 30% chance of being decoyed. With the exception of dumbfire missiles or rocket pods that just fly a straight line, there is a decoy type for each missile:

- **Distortion Projector:** Image Recognition Decoy that falls x meter behind the craft before powering its thruster to pace the player's speed and direction. It pulses a blue/green light from a series of channels ringing its surface to represent the scattered band laser

emission that attempts to scramble the missile's recognition ability every ½ second until its battery dies.

- **Thermal Pulse:** Heat Seeker Decoy that spins after release shooting dual thrusters and pulsing from red hot to cold steel every ½ second until its battery expires.
- **Burst Shadow:** Friend or Foe Decoy that pulses signal ghosts of the player's signature every ½ second until its internal battery runs out.

Crafting Note: Decoys can be built from schematics and the appropriate materials. The schematic determines the type it will be and will have a recipe of necessary resources/components/systems/code in order to build. Decoys are initially rated at being a low percentage (30%) Crafters with appropriate skills should be able to eventually raise that as high as 75% through successive software and hardware upgrades.

Primary Properties:

- **Required System(s):** Requires an installed Decoy Pod launcher.
- **Size:** Either a 1 or 2. Determines the number of slots it occupies in decoy pod. Burst Shadow takes two slots per decoy all others are 1.
- **Weight:** Factors into overall ship speed and handling
- **Percent Chance of successful decoy on missile type:** Starts at a base of 30% every ½ second against standard build missile types.
- **Life span:** Determined by battery, up to x seconds (TBD between 3-5 seconds).

Schematic Recipe:

- **Housing** - Each decoy has its own housing. Structural material (materials TBD) used to craft housing combined with battery determines weight.
- **Software Package:** (packages TBD) One for each type of decoy. Package level can increase the base percentage chance for success up to 10%.
- **Electronics:** Electronics package (packages TBD) Package level can increase the base percentage chance for success up to 10%.
- **Motor** (for DP and BS): Motor package (packages TBD)
- **Battery:** (batteries TBD) Determines decoy lifespan and weight.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase disruption chance by up to 15%.
- **Tools/Facilities that are required to construct** (TBD) Tech levels of facilities or tools used can increase the base percentage chance for success by up to 10%.

Idea Note: Software packages can have versions that contain additional features, such as ECM resistance. The notion of modifiable software packages being engineered or hacked is an interesting potential feature for crafting.

ECM, Anti-Missile Cannon, ECCM

An **ECM** system automatically activates when a friend or foe missile locks on. These are expensive units with different efficiency ratings. They free up valuable pod launcher

Wing Commander Online: Privateer Ships - Defenses - Missile Defenses

slots for anti-IR and anti-HS decoys and run automatically. They can also be configured to automatically deploy decoys for non-FF missile types.

ECM uses a quick charge internal battery with x duration based on battery level with a base level of 30% per second chance of disrupting an incoming missile's lock.

ECM availability for ship installation is dependent on the type of communications system installed. Successively higher-level communications systems increase the base disruption chance of 30%, by x%.

ECM with x level of software package can control an anti-missile cannon(s) to take down any missile that gets within x range (See Anti-Missile Cannon below).

Primary Properties

- **Required System(s):** Requires Communication System of x level (TBD). Successively higher-level communications systems increase the base disruption chance of 40% by up to 10%.
- **Power drain** (on main reactor): while active in Gj (gigajoules)
- **ECM Pulse duration:** x seconds, determined by battery
- **Percentage Chance of Disruption:** Starts at a base of 30% as used against a standard Friend or Foe device.
- **Weight:** Factors into overall ship speed and handling

Schematic Recipe:

- **Software Package:** (packages TBD) Package level can raise the base percentage chance for success a maximum of 10%.
- **Electronics Package:** (packages TBD) Package level can raise the base percentage chance for success a maximum of 10%.
- **Battery:** (batteries TBD) Determines power drain, pulse duration, and weight.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase disruption chance up to 10%.
- **Tools/Facilities that are required to construct** (TBD) Tech levels of facilities or tools used can raise the base percentage chance for success a maximum of 10%.

Anti-Missile cannons are modified articulated Vulcan cannons that attach to any turrets the player's ship has. These have limited Vulcan ammunition rounds. They do not actually shoot the missiles out of the sky. When a missile or torpedo comes within x range of a given cannon's firing arc, the cannon in question begins to "fire". This is a brilliant muzzle flash with random tracers and the sound of a high-pitched rotor whine. There is x% chance every ¼ second a missile is within the arc that the cannon will destroy it.

Although infrequent on fighters, well-equipped heavy transports or capital ships will appear to light up with these bad boys on turrets.

This percentage is determined at a base level of 25% and increased in relation to software package levels of the ECM unit controlling it, the onboard targeting computer and the Cannon's properties.

To get that cool auto-turret sequence in Aliens: when the Anti-Missile Cannons begin firing a pop-up counts down the available ammunition.

Wing Commander Online: Privateer Ships - Defenses - Missile Defenses

Anti-Missile Cannon
200 round(s) left

or

Anti-Missile Cannon
[REDACTED]

Primary Properties

- **Required System(s):** Requires a mounted turret and ECM package of x level (TBD). Successively higher-level ECM systems can increase the base hit chance of 25% by up to 10%.
- **Power drain:** (on main reactor) while active in Gj (gigajoules)
- **Size:** Requires one standard turret hardpoint.
- **Weight:** Factors into overall ship speed and handling
- **ECM Pulse duration:** x seconds
- **Percentage Chance of Disruption:** Starts at a base of 25% as used against a standard missile.

Schematic Recipe

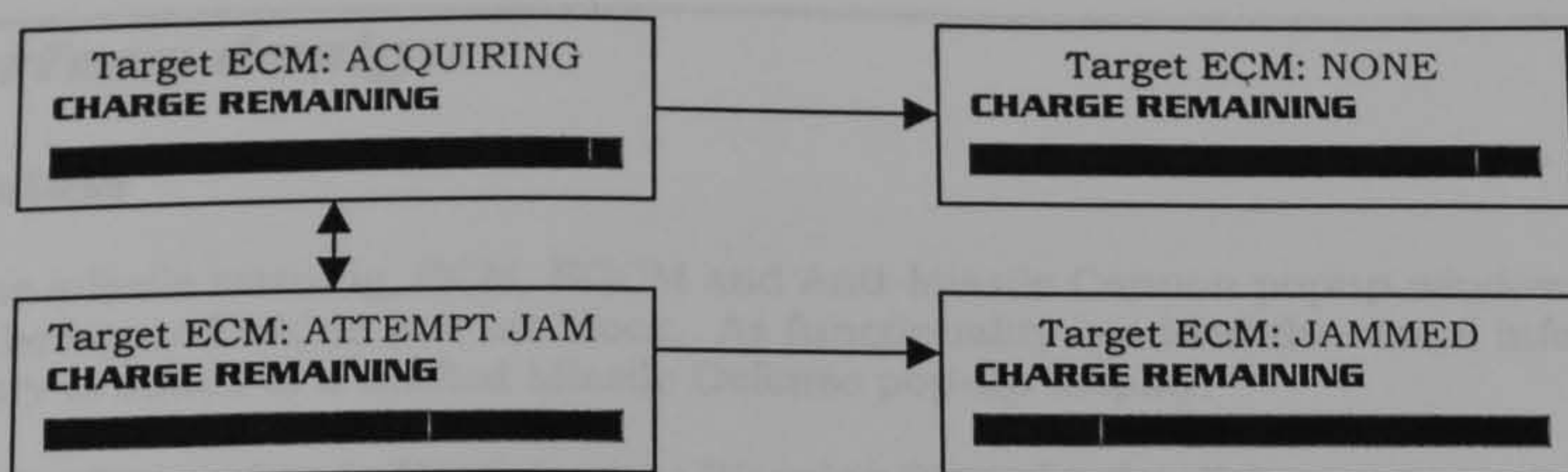
- **Housing** - Structural material (materials TBD) used to craft housing determines weight.
- **Software Package:** a (packages TBD) Package level can increase the base percentage chance for success by up to 10%.
- **Electronics Package:** (packages TBD) Package level can increase the base percentage chance for success by up to 10%.
- **Motor Type:** (types TBD)
- **Ammunition:** Vulcan rounds
- **Ammo Housing**
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase disruption chance up to 10%.
- **Tools/Facilities that are required to construct** (TBD) Tech level of facilities or tools used can increase the base percentage chance for success by up to 5%.

An **ECCM** jams the target's **ECM** unit for x duration (determined by battery) at a base 40% chance to jam. This percentage is modified by the comparative tech levels of the competing systems software (i.e. an ECCM with a level 7 software package will have a significant % chance bonus to jam an ECM with a level 2 software package). The construction levels of the unit's schematic ingredients further modify the percentage.

When activated the ECCM attempts to establish a jam by pulsing the target every ½ second. When and if a jam is established it is then fixed until the ECCM unit's energy runs out. The ECCM is either manually activated by the player, or can be set to automatically engage when an enemy target is locked. The ECCM first **ACQUIRES** the enemies ECM for ½ second (basically checks to see if the target has one). If the target has no ECM unit that fact is displayed and the ECCM automatically disengaged. If the target has an ECM the next state is **ATTEMPT JAM**. This remains until either the ECCM successfully jams the target ECM or it runs out of juice and the cycle begins anew after the unit recharges (based on Power Drain property and Reactor Output).

When the player activates the ECCM unit a pop-up indicates the status of the target's ECM unit and the amount of charge remaining:

Wing Commander Online: Privateer Ships - Defenses - Missile Defenses



Primary Properties

- **Required System(s):** Requires Communication System of x level **and** Targeting System of x level (TBD). Successively higher-level communications & targeting systems increase the base jam chance of 35%, by up to 5% each.
- **Power drain** (on main reactor): while active in Gj (gigajoules)
- **ECCM duration:** x seconds
- **Percentage Chance of Disruption:** Starts at a base of 30% as used against a standard Friend or Foe device.
- **Weight:** Factors into overall ship speed and handling

Schematic Recipe:

- **Software Package:** (packages TBD) Package level can increase the base percentage chance for success by up to 5%.
- **Electronics Package:** (packages TBD) Package level can increase the base percentage chance for success by up to 5%.
- **Battery:** (batteries TBD) Determines power drain and weight.
- **Crafter Skill/Certification Required:** Crafter type and minimum level of certification to successfully build. Higher levels of certification may increase base chance up to 10%.
- **Tools/Facilities that are required to construct:** (TBD) Tech level of facilities or tools used can increase the base percentage chance for success by up to 5%.

Other Missile Defense Tactics

Of course players should still be able to potentially out maneuver missiles/torpedoes or target them and blow them out of the sky (dependent on radar/targeting computer level).

Heat Sink

Luring HS missiles into another ship's exhaust has always been a really cool tactic, but to avoid PK tactics we must invisibly restrain it.

If a player maneuvers such as to position a heat seeker within x range of an **enemy** exhaust/engines there is x percentage chance the missile will disengage from the player and acquire the enemy ship as its new target. It will not recognize a neutral or friendly ship.

A player can also attempt to maintain a safe distance from a missile and outlast its booster battery until it's depleted. At which time the missile explodes.

Other tactics are TBD.

Interface design

Overview

All of the missile warning, ECM, ECCM and Anti-Missile Cannon popup windows should be housed within one art block. As functionality is added the visual info necessary is added to a unified Missile Defense pop-up display.

Missile Lock warning indicator – (see Warning Signs for details)

Controls

- None

Art Needs

- Pop-up or cockpit light for lvl one Missile Alert (see Warning Signs)
- Pop-up for lvl two missile alert (see Warning Signs)
- Pop-up for lvl three missile alert (see Warning Signs)

Sound Needs

- Warning Klaxon for Missile Alert (all levels)
- Warning sound for ECM JAM (level 2 and 3)
- Warning sound for ECM DEPLETED (level 2 and 3)
- Missile Proximity alert 3 stages based on range (level 3 only)

Decoys - (see Decoys for details)

Controls

- Single key/button press to deploy decoys

Art Needs

- Distortion Projector IR decoy object
- Thermal Pulse HS decoy object
- Burst Shadow FF decoy object
- Decoy Pod launch exhaust particles
- Decoy pod launcher object for ships (TBD)

SFX Needs

- Decoy pod launch (click, whir, thunk, pshhhhh, **in other words** key press, rev, disengage, exhaust sound)
- IR ambient (1 second loop)
- HS ambient (1 second loop)
- FF ambient (1 second loop)
- "Out of Decoys" warning bell
- Out of Decoys failed launch sound for Pod Launcher (preferably the start click & whir of the standard launch sound without the thunk and exhaust sound of the launched decoy firing away).

ECM, Anti-Missile Cannon, ECCM – (see ECM, Anti-Missile Cannon, ECCM for details)

ECM Controls

Wing Commander Online: Privateer Ships - Defenses - Missile Defenses

- none

ECM Art Needs

- Covered under Missile Lock

ECM SFX Needs

- Covered under Missile Lock

Anti-Missile Cannon Controls Needs

- None, fully automated

Anti-Missile Cannon Art Needs

- Cannon object for turret hard point. Fully articulates along turret within 180° or available safe firing arc.
- Cyclical Muzzle flash while firing
- White Hot tracers with a hint of yellow that **fade out** in ½ second
- Pop-up to indicate active status and remaining ammunition count

Anti-Missile Cannon SFX Needs

- Rev up mechanical whine start sound (½ second)
- higher pitch vulcan cannon firing sound (loop)
- Rev down mechanical whine stop sound (½ second)
- Pop-up "Out of Ammo" bell
- Out of ammo clicking sound from cannon (½ second)

ECCM Controls

- Single key/button to activate ECCM

ECCM Art Needs

- Pop-up indicating target ECM status and remaining ECCM charge.

ECCM SFX Needs

- ECCM active pulse
- Target ECM jammed bell
- ECCM charge depleted klaxon
- ECCM "No Target ECM" tone

Tasks

- Art needs (both assets and implementation)
- Sound needs (both assets and implementation)
- Distortion Projector IR decoy object follow ship routine
- Thermal Pulse HS decoy object spin routine
- Missile vs. Decoy/ECM/Anti-Missile Cannon die roll conflict resolution system
- ECCM vs. ECM conflict resolution system
- Pop-up triggers

Wing Commander Online: Privateer Ships - Defenses - Missile Defenses

- ECM and ECCM charge counter
- Anti-Missile Ammo counter
- Heat Seeker reacquire AI (if player gets x distance from enemy ship exhaust)

To be addressed

- Decoy Pod Launcher characteristics, crafting needs, etc.
- Varied percentages, and crafting specifics for each system
- Other Missile defense/evasion tactics

Overview

Wing Commander has always been about fast paced combat with intuitive controls. For Wing Commander Online we want to streamline the system and change certain elements to address bandwidth concerns. We want to make WCO a game for both action and strategy junkies, including twitch skills with tactical thinking.

Vision and goals

Goals

- Keep the fun factor high
- Defeat the Wing Commander Online problems
- Make installing occupancies and power a fun tactical addition instead of an annoying mechanical
- Maintain the simplicity of Wing Commander flight while adding the beautiful effects real physics afford us

Vision Points

- ... must be easily accessible to a mass market audience and a casual player
- ... must have an intuitive user interface
- ... must have a presentation that blows people away at first glance (physics effects)
- ... will offer a high degree of user customization for player characters and ships
- ... should expose new concepts throughout the game to the player
- ... will make use of "rock-paper-scissors" relationships so that no one specialty, gun, or skill is perfect
- ... must support many different styles of play
- ... must have 10+ tactics that add depth
- ... must make the player feel smart

Walkthrough

In this doc we will briefly examine:

- Wing Physics and how they affect the player
- The new maneuvering method, including the Afterburner Fix, Kicks, Combat maneuvers, etc.
- Power management system

Systems Design

Physics, Maneuvering, Power - Fighters

Overview

Wing Commander has always been about fast paced combat with intuitive controls. For Wing Commander Online we must streamline the system and change certain elements to address bandwidth concerns. We want to make WCO a game for both action and strategy junkies, balancing twitch skills with tactical thinking.

Vision and goals

Goals

- ☐ Keep the function simple but the form deep.
- ☐ Defeat the Wing Commander eternal afterburner problem
- ☐ Make installing components and managing reactor power a fun tactical addition instead of an annoying headache.
- ☐ Maintain the simplicity of Wing Commander flight while adding the beautiful effects real physics afford us.

Vision Points

- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... must have an intuitive user interface
- ☐ ... must have a presentation that blows people away at first glance (physics effects)
- ☐ ... will offer a high degree of easy customization for player characters and ships
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... will make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ ... must support many different styles of play
- ☐ ... must have little touches that add depth
- ☐ ... must make the player feel smart

Walkthrough

In this doc we will briefly examine:

- Wing Physics and how they impact the player.
- The new maneuvering method, including the Afterburner Fix, Slide, Combat maneuvers, etc.
- Power management system

Systems design

WCO Physics

Wing Commander will utilize Newtonian physics, with a fly-by-wire system for the player.

Essentially all of the natural motion, arcing debris, maneuvering capabilities, explosive properties, will be represented **without** the player having to learn how to fly a real plane or spaceship. This is very important. The benefits of a real physics system are not to complicate the play experience, but to enhance it. Never punish the player for not understanding a college/High School course that they are most likely skipping to play your game.

All of the tedium of reality should be automated. Wing Commander has traditionally been a game that any individual could get a hang of. You push your joystick left the ship turns left. How you manage your shields, armor, gun power, maneuver/afterburner energy, and the other toys you equip your ship with is what we want players to worry about.

"But what's the fun in having a physics system, when the player doesn't have to learn to use it?"

Well, for special effects, visual rewards, and a more visceral sense that you are truly in space. For example when travelling through an asteroid field, the rocks float, spin, drift and react to impact naturally. Other than steering, the player doesn't have to worry about thrust, inertia, etc. (unless they accelerate into one) as the ship systems in the 29th century are built to take the guesswork out of navigating dangerous space. They just have to sit back, enjoy the game, and give us bags of money.

Maneuvering, or How to Fix the WC Constant Afterburner problem

With the exception of Wing Commander: Privateer players have had to traditionally keep a finger planted on the TAB key to effectively pilot their ship in combat. You were able to hold down the afterburner key and maneuver in any direction, quickly changing your vector at high speed. This led to a number of issues with Single Player, but for multiplayer it adds a practical bandwidth problem that requires change.

To address this we are introducing the notion of maneuvering thrusters that require energy to change your ship's direction. You turn your joystick left or right, up or down, and these jets fire, giving your opponent a good idea of where you're about to be going, as well as a visual reward we've yet to see in a AAA space sim.

These thrusters require energy to fire. The more severe your turns are (against speed, weight, etc.) the more of a drain on your thruster energy battery. If you constantly make high-speed maneuvers this battery will begin to drain until you are rendered a very slow turning tub.

Afterburner energy is also tied to this battery, and drains it faster than standard maneuvers alone. Attempt to maneuver while afterburning and you'll find that energy reserve drain quite quick indeed.

So afterburner usage is balanced against your maneuverability. Quite a nice model and one that reduces the bandwidth concerns of high-speed directional changes since you'll be flying an almost straight line if you utilize your afterburner.

Players will be able to upgrade their maneuvering thrusters to consume less power at higher tech levels, however afterburner drain will only increase with higher tech levels

(providing a more powerful burst at the expense of battery power) to balance out the gains.

Articulated main engines give the player the opportunity to drastically increase the yaw or pitch capability of a craft while reducing the maneuver drain for the axis in question.

Sliding

One of the cooler features found in WC was sliding. This is similar to the Bab5 visual of a craft riding their current vector while spinning around on their axis. This is a component we should preserve for the fun-factor/coolness element it provides. Nothing beats Sliding when doing an attack run on a Capital Ship, or spinning around to launch missiles at an enemy.

The dead reckoning issues involved are significant, but not overwhelming. However, because of this, we must provide limitations and risks to sliding.

When a person engages a slide (a toggle), their directional input fires the maneuvering thrusters and rotates the ship around the axis while locking the current speed and vector the craft is travelling.

As soon as the slide is engaged a ray is cast that the player must move back to before disengaging the slide. The ray cast from the front of the ship is green, from the rear of the ship is red. The player must center the green ray in their reticule before disengaging slide to avoid a compensating thruster hit.

There is a massive thruster drain as the ship attempts to compensate for the new vector the pilot wishes to travel against their current. If the pilot is facing directly opposite his vector then his/her afterburners fire to compensate. This will render the ship very slow (if not stopped) and unresponsive for several seconds.

The massive thruster drain to compensate for coming out of a slide and into a new vector will be a distinct handicap the player must weigh against the benefits of sliding.

All in all, this adds a new wrinkle to typical WC/Freespace/X-Wing combat that will look as fun as it is to play.

This leads us into power management.

Power Management

There is no fuel per se in WCO. A reactor will work forever, but the amount of energy it delivers to the system batteries is variable based upon drains in the form of additional systems, complete system battery depletion etc. This section will hopefully describe the power structure of WCO ships.

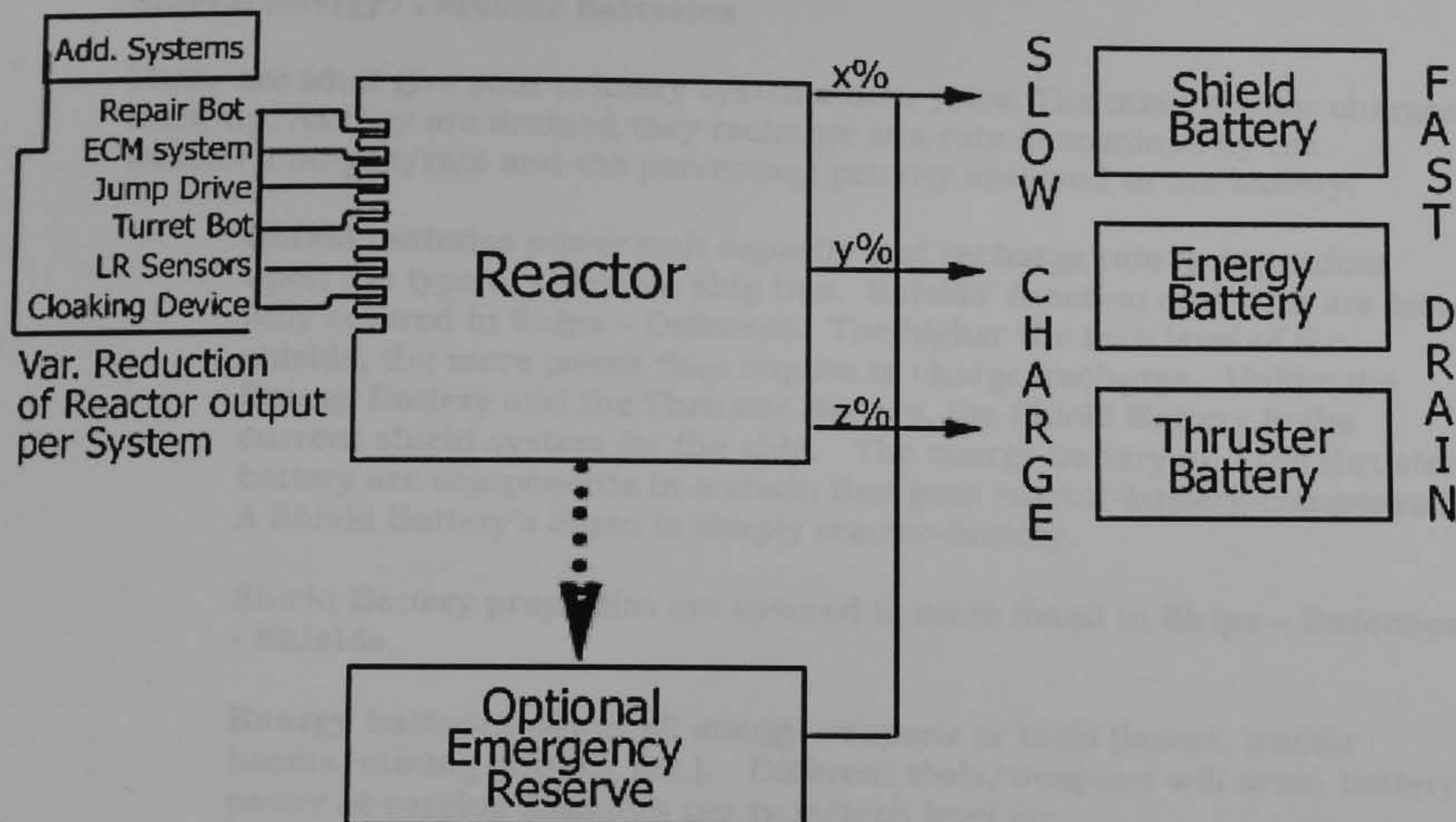
An underutilized feature of previous Wing Commanders was Power Management. The player had a triangle with the extremes of ENGINE, SHIELDS, and GUNS at each corner. By moving a small dot they could set the level of power going to each of these energy pools.

Because the ships you flew (with the exception of Privateer) were all fixed in their specifications, this was sufficient. However, in WCO we are allowing players to upgrade and modify nearly every aspect of their secondary avatar: their ship. This means that

reactor power, system batteries, and additional component drains become strategic considerations when building your dream ship...and engaging in combat.

Players will be able to configure the amount of power delivered to each battery from the main reactor, route power from one battery to another, turn on and off additional systems, and even turn off all ship power (except life support/emergency lights). This allows for really power on and off tactics that include hiding from radar, asteroid hide and seek.

First up, here's a basic diagram of how power is routed:



Let's unpack, shall we?

Reactor

The key to a ship's power is its **reactor**. The reactor, in tandem with the overall ship weight (chassis type, all components), dictates a ship's top speed. It has a total amount of energy, in units that it delivers to the ship's three main batteries at a given rate. This total amount is decreased by any additional ship systems the player installs.

Different reactors can provide increased total output, as well as increased rates of transfer.

So the main properties of a reactor are its:

- Size (limits Reactor/Chassis combinations)
- Weight (impacts overall ship handling and speed)
- Total Power Output (in Gigajoules or Gj)
- Power Transfer Rate (Gj per second)

Different Strokes...

The above is the most basic arrangement. Larger ships (capital ships, etc.) will be able to house more than just three batteries, but they must have at least one of each type. However multiple like types are considered one battery for purposes of power management.

What this means is that one player with a heavy transport may have 4 batteries; One for energy, one for thruster, and two for shields. Another may have one thruster, one shield, and two energy batteries. However when they manage their power they will only have the gross categories of SHIELD, ENERGY, and THRUSTER to route power to.

- In order to link multiple batteries of the same type you have to use an "additional system"-crafting slot on your ship for a **COUPLER** to connect the two.

Damage can effect power output and transfer rates. However, a reactor will never be completely/permanently disabled in order to prevent players getting stuck. The exception to the rule is EMP, or Leech weapons that temporarily drain/disable ship power.

The reactor feeds its power to the main batteries. These can be configured to consume varying percentile amounts (75% shields, 15% energy, 10% thrusters etc.).

Shield/Energy/Thruster Batteries

These are what give your primary systems their juice. The main reactor charges them up. As they are drained they recharge at a rate determined by the Reactor's output/rate and the percentage priority assigned to the battery.

Shield batteries power unit capacity and recharge rate is dependent upon the type of shields a ship has. Shields' function and form are more fully covered in Ships – Defenses. The higher the tech level of the shields, the more power they require to charge/recharge. Unlike the Energy Battery and the Thruster Battery, the Shield Battery **is** the current shield system for the ship. The energy battery and the thruster battery are components in a chain that goes reactor-battery-component. A Shield Battery's chain is simply reactor-battery.

Shield Battery properties are covered in more detail in **Ships – Defenses - Shields**.

Energy batteries power all energy weapons or tools (lasers, tractor beams, mining cutters, etc.). Different tools/weapons will drain battery power at varying amounts per type/tech level etc.

Energy Battery properties include:

- Size
- Weight
- Power Capacity (Gj)

Thruster Batteries provide you with Afterburner and Maneuvering power.

Thruster Battery properties include:

- Size
- Weight
- Power Capacity (Gj)

Additional Systems

Additional systems include everything that isn't a tool/weapon that is "fired". These are items that have constant drains on the Reactor, reducing its total output by the amount of power they use.

These systems power options should be toggled in the following ways:

- Individual system on/off
- Customizable Grouped systems on/off

- All systems on/off

Additional systems will have the following power related properties in addition to their individual template/blueprint properties:

- Size
- Weight
- Power Drain

The interface for this is TBD (see Power Options).

Power Options

As stated in the Additional Systems section the player will have very fine control over which systems are and are not drawing power from the reactor.

They should also have the ability to route the main batteries in the following ways:

- Dump Three battery power to One
- Dump from Two to One
- Dump from One to One
- Change % reactor drain between battery types.

The interface for power management must be very powerful and very simple. This, like shield management (see Ships – Defenses – Shields), is a very tricky task indeed.

Interface design

Controls

- Main battery Power Management – TBD
- Additional Systems Power Management – TBD
- Activate Slide – Single key press, toggle ON/OFF TBD
- Afterburner – Single key press, HOLD to sustain. TBD

Art

- Power Management Interface – Reactor/ Battery power TBD
- Power Management Interface – Additional System power options TBD
- Afterburner cone - TBD
- Maneuvering jets – TBD
- Slide vector art (ray that indicates current direction of thrust) – See Maneuvering - Sliding

Audio

- Afterburner 3 stage sound – Engage, Loop, Disengage
- Slide activate sound – Short Tone
- Maneuvering jets – Jets fire (very short looping “can of air” type white noise)
- Power Management – Sound of reactor energy % being routed to the batteries
- Power Management – Sound of battery energy being dumped to another.

Tasks

To be addressed

- Power Management interface
- Additional Systems Interface

Generated Missions

Ground and Space

Overview

In Privateer Online players will be posting requests for services on a mission board. Some will have good intentions, some bad intentions, but all come with missions that demand payment. There will also be computer-generated missions to fill in the gaps of difficulty. The fictional corporations of Privateer Online (Phantom Dynamics, Bio-Tek Industries, Paradise Kinetics, etc.) will frame these missions as requested services. This will provide novice and experienced players a supply of tasks to undertake, it needs to learn, offer to advance, and others to recruit. What those tasks are and how they come to be is covered later.

Goals and vision**GOALS**

1. The mission system should have the diversity of difficulty levels within the same play session.
2. The mission system should be able to home in on player-made tasks, but also offer a variety of computer-generated tasks to provide a variety of experiences and variety.
3. The mission system should be able to randomly generate a series of related missions, instead of just a single mission, to create a chain of related experiences.
4. Introduce a series of Levels/Complications that combine multiple missions into a contiguous experience.
5. The mission system should evolve the player's experience beyond the initial mission.
6. This mission system should feel as unique as possible.
7. The mission system must not only give the player something to do, but also provide them with a template from which to view the future of their own future mission progress.
8. Make non-graphic mission advances that could include and have that can't be readily quantified on a bar.
9. Missions should encourage players to work on each other for help. (e.g. A mission may be completed by only one person, but the more advanced may include a player accepting a mission and then posting a help request to the other to make up the experienced player's/ship's/surprise/surprise necessary to complete the job).

VISION

1. ... must be easy to get into, as a new player and as an advanced one.
2. ... must offer clear goals to achieve.
3. ... should have gameplay that can be a short play session (< 10 minutes).
4. ... should have a human element to the mission.

Generated Missions

Ground and Space

Overview

In Privateer Online players will be posting requests for services on a mission board. Some will have good intentions, some bad intentions, but all come with motives that demand payment. There will also be computer-generated missions to fill in the gaps of difficulty. The fictional corporations of Privateer Online (Hurstons Dynamics, Bartok Industries, Paulsen Kinetics, etc.) will frame these missions as requested services. This will provide newbies and experienced players a supply of tasks to undertake, lessons to learn, skills to advance, and allies to recruit. What those tasks are and how they come to be is covered here.

Goals and vision

GOALS

- ❑ The missions generated have to reflect the diversity of abilities possible within the game play experience.
- ❑ The "mission board" the players refer to will be home to player made tasks, but must always offer a selection of AI generated newbie experiences and scalable advanced objectives.
- ❑ The missions being offered should be randomly generated from a series of linked matrices, instead of just cycling through a chain of fixed handmade experiences (ala Everquest).
- ❑ Introduce notion of Links/Complications that combine multiple missions into a contiguous experience.
- ❑ The mission system must evolve the player's experiences beyond the initial tutorials.
- ❑ This mission system should feel as natural as possible.
- ❑ The mission system must not only give the player something to do, but also provide them with a template from which to base the format of their own future mission postings.
- ❑ Make non-newbie missions adventures that contain twists and turns that can't be readily quantified on a fan site.
- ❑ Missions should encourage players to reach out to each other for help. (*i.e. A newbie mission may be accomplished by only one person, but the more advanced may involve a player accepting a mission and then posting a help wanted ad of their own to rustle up the experienced pilots/engineers/surveyors necessary to complete the job*)

VISION

- ❑ ... must be easy to get into, as a new player and as an advanced one
- ❑ ... must offer clear goals to newbies
- ❑ ... should have gameplay that can fit a short play session (< 30 minutes)
- ❑ ... must minimize human administrative intervention
- ❑ ... will be easily extensible by the Live team
- ❑ ... should expose new strategies throughout the game to the player

- ☐ ... will offer motives and rewards for short session play
- ☐ ... should encourage specialization
- ☐ ... must support many different styles of play
- ☐ ... should support character building and advancement
- ☐ ... should have little touches that add depth
- ☐ ... should have secrets and lore to it
- ☐ ... will offer players lots of rewards

Walkthrough

The major components required for this system are:

- *Basic mission categories*
- *The Mission Generator*
- *Ground and Space Mission/Skill Objectives*
- *Generating Links/Complications between missions (TBD)*

An example of a newbie mission:

Jimmy has been in the game for a very short time. He wants his avatar Irin to eventually become a merc for hire, capable of cleaning the galaxy of any scum for a price. But Irin has to crawl first, so Jimmy heads over to the mission board to see what formative tasks he can tackle. He checks out the list and prioritizes based on Pilot skill. The first mission he sees is for Hurston Dynamics, who needs a scout to recon a signal coming from an automated mining facility in a Proxima asteroid belt. It's a guaranteed amount of credits, not huge, but it's not a huge deal. It also states his ship's computer will have all security transponder signals for the station uploaded for rendezvous.

He hops in his ship and takes the jump.

Arriving in Proxima he pulls up the Nav map and clicks on the nav marker placed after he agreed to the mission. Kicking in a short burn pod he had added to his craft he flies towards that heading until he gets the blip of the station amidst a tumbling field of rock.

A small chime indicates a warning message from his Nav computer advising him to slow his speed before he enters the field. Obeying, he lets off the throttle and begins to weave in and out of the spinning asteroids, heading ever closer to the mining station's blip. Finally, it's in site, and a wreck. Arcing sparks and scorched metal indicate that the facility was attacked, and recently. When he gets within 150klicks his communications channel clicks in with a secure text message. "Hurston Dynamics: Thanks Irin, we need you to target and hail the station. It will then upload the latest data...should only take a few seconds."

Jimmy targets the station, communicates with it and a small hum with the message "Status Upload Initiated" rolls across his ComScreen. A chime indicates completion and a new communication from Hurston comes in: "Good Work Irin. We are crediting your given account with the promised payment. We look forward to working with you in the future."

Jimmy feels good, and wealthier. He's learned how to navigate an asteroid field, and seen that danger lurks even in supposedly safe spaces. He looks forward to taking down whomever it was that hit the facility in the future. Now back to Sol to buy that Gauss cannon he saw for sale.

Systems design

Fictional Framework

The fictional corporations that populate Privateer Online have dominion over certain sectors of Confed space. As with any corporation, temporary services are required to flesh out the portfolio and the players are the talent pool.

The more missions you run for a particular corporation, the higher your rep with that company when it comes to buying newbie goods. Of course, there does come a point of diminishing returns the further away from the initial Newbie skill set you get.

Core Rules To Mission Generation

There must be x number of newbie missions available no more than x hops from any starting system, at any given time. These will represent an opportunity to expand and explore each of the different primary skills (Skills TBD).

Newbie missions should always take place within "Safe" space, or a comfortable distance from safe space.

Newbie bounty, escort or an equivalent ground offensive/defensive mission will only pit the new player against enemies of equal or lesser strength balanced against the status of the player's ship or weapons.

Newbie missions will always have a min number available, no matter what start system you are from. The advanced and scalable missions serve to flesh out a roster not necessarily exploited by the player. So, if the mission board drops below x non-newbie missions, the mission generator should pick up the slack.

When a player accepts a mission there are two potential timers activated. One is an actual Mission Timer that may or may not be present (*i.e. You must deliver these goods to System x before they decay*) the other is a maximum "no action" time limit. This means the player can accept and not actually complete a mission for x amount of time before that mission decays and returns to the board for another player to accept.

A newbie objective will only be accessible by the person who accepts the mission. In most cases this will manifest in the form of objectives spawning only when the player gets within x distance of the target, and in the case of deliverables they will only be attachable to the player's transport. This is covered more in *Mission Types*.

Skill Development

Newbie missions are about exploring and advancing the skills the player is most interested in. They are also about training newbies on the core elements of gameplay they will need to become very comfortable with over the course of the game.

The primary skills a player pursues (ex: surveying, piloting, gunnery, farming) need to drive the newbie mission structure. Overall objectives for missions are also determined by the primary skill required to accomplish it.

Mission Types

Missions will take place both in space and on the ground. The objectives for these missions revolve around a core group of types that will be fleshed out to represent specific objectives for certain skills:

- 1) Patrol/Recon

This can be either a ground or space based mission involving the player fly or travel through a given volume and "report" his/her findings to the client. This "report" is sent automatically when the player hits all of the given points or has completed the instructions offered when they reach their Patrol/Recon location.

2) Recover

A recover mission can either be ground or space based and involves the player recovering a widget/artifact/goods for the client. In all cases of AI generated missions this artifact will spawn as soon as the player approaches its location. The notion of keyed transponder freqs can be used to secure the item from looters. (ex: *A cargo ship can only connect to a missing cargo pod if it has the proper transponder signal and can only find it with the freq for its beacon signal. If another player loots this item it will "self-destruct" rendering any "goods" inside worthless.*)

3) Survey

A survey mission is similar to a Patrol/Recon mission, but involves the skills of a Surveyor to complete. This involves detailed mineral readings from a planet, atmospheric readings, nebulae samples, and core samples from Asteroids. This is for all of the primary explorer skill sets.

4) Establish

Establish mission involves the placement or installation of equipment by the player (Ex: *A datastream beacon in a newly discovered system, a defensive grid on a planet, an automated mining drone, atmospheric processor etc*).

5) Deliver

Deliver from point A to point B. 'Nuff said. Same rules for cargo ownership as described in *Recover* applies.

6) Assault

Can be either ground or space. Assault includes hunting enemy AI ships, enemy installations (i.e. *Pirate bases, Retro installations*) AI ground critters etc.

7) Defend

Defend installations, bases, etc. from attack.

{Note: that players should be rewarded/encouraged to post missions that utilize new players in roles on capital ships as gunners, navigation, tactical, etc. This also applies to players hiring newbies as transport drivers and "taxi service" providers. This will be covered in detail in both the "mission board system" doc, and "player generated missions" document. }

Mission Generation

When a newbie mission is accepted and completed, that mission is replaced with a new one. The method of generating the new mission is as follows: (a simple flow of this is at the end of the document)

1) What is the primary skill for the mission?

This is the most important question. This will determine the resources that need to be present for the skill to be utilized (*nebulae, asteroid field, enemies, ore, goods, planets, trading post etc.*)

2) What are the secondary skills (if any) for this mission?

This will determine the nature of any secondary objectives present (*Asteroid field for Navigation, targets for gunnery, communication skill for uplink, etc.*)

3) What system(s) does this take place in?

The first step here is to check which newbie start locations have the least amount of available newbie missions and/or don't have the current skill exploited.

Next it scans the surrounding volume of systems (within *x* distance) for ones that possesses the resource(s) necessary for the skill(s) objective to be generated. In some cases this might be a single system (*go to Planet x in the Mclean system and survey the mineral density of the third planet*) in the case of a delivery it will need to link two systems with a trading post etc.

4) Is the mission Ground or Space based?

This is based upon the skill to be exploited and the dimensions of the system selected. If the skill to be exploited is Ground Weapon tactics then it would have to be on a planet, whereas a Survey skill mission may be space or ground based dependent upon the resources present.

5) Then we head down branches based upon whether the mission is Space or Ground based. Both have similar paths to follow with the exception of the first step: "Which Planet is it on?" Or "Where in the System does it take place?"

6) What is the primary objective?

Once the skill(s), system(s), and ground or planet states have been determined we can then analyze the resources present and draw out an objective. (i.e.: *Survey planet y in system x for its current Iron Ore density*).

7) What is the Guardian(s) of this objective?

Some missions may have guardians that "protect" the objective from the player. These guardians may be pirates, indigenous ground life forms, or even space features such as dense asteroid fields or volatile nebulae.

8) What are the secondary objectives?

Some missions may have secondary objectives that are tied to the secondary skills to be exploited – "After delivering the unobtainium scan the planet for its agricultural potential", or "While Patrolling the Asteroid field eliminate any hostiles present."

9) What is the Guardian(s) of this objective?

Same as seven.

10) Who is the client?

Based upon the mission's stellar location, and the objectives involved, the client (Either a corporation or Confed) will be selected.

11) Is there a time limit?

Based upon objective, some missions may have inherent time limits,

12) What is the value of the mission?

Based upon all of the above, a monetary value for the mission will be determined.

13) Is there a Link or Complication?

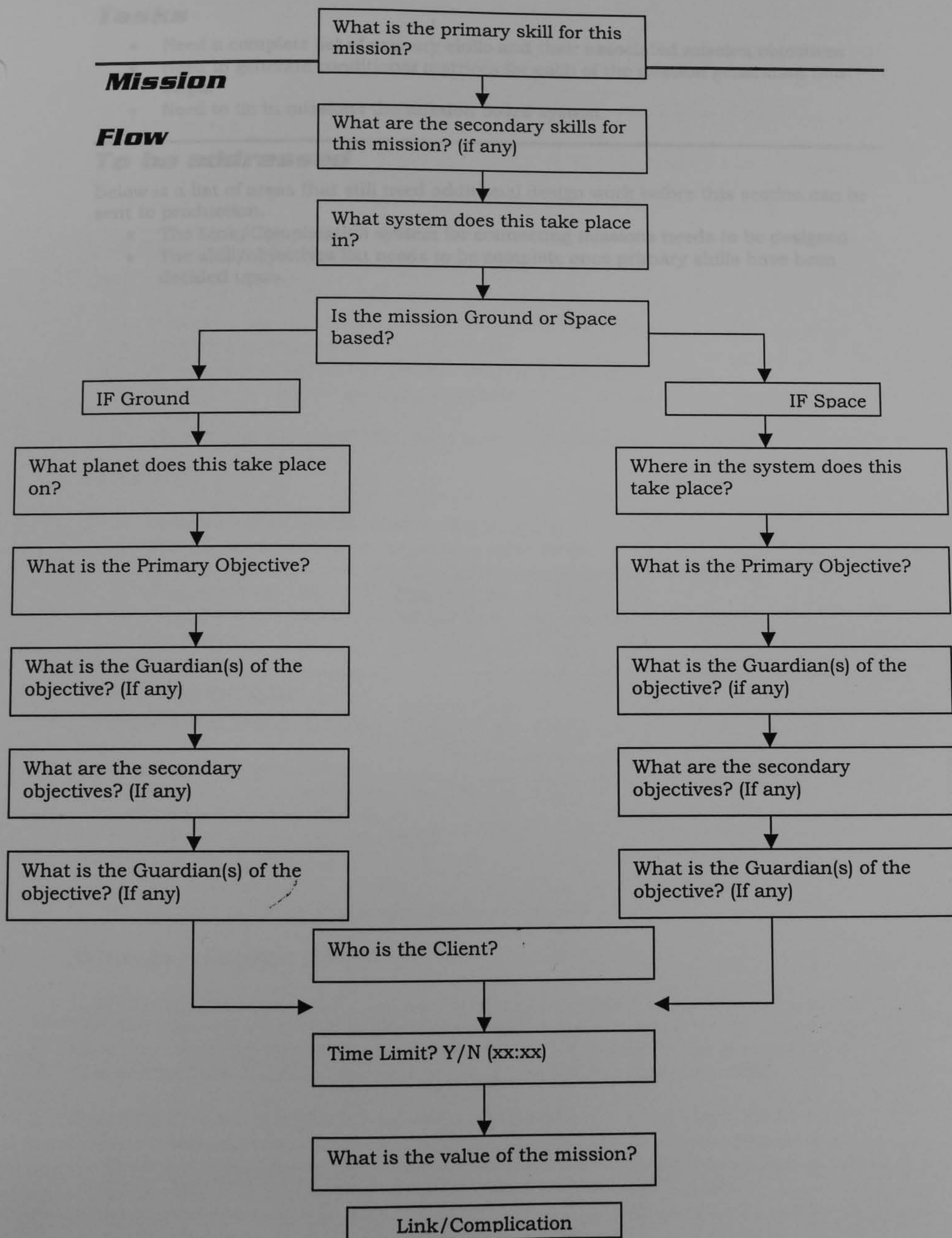
This will gauge the player's ability and generate a potential "complication" or "link" to another mission that utilizes the common objective's current resource (TBD).

The mission example given at the beginning of the document can be generated using this system.

All of the above steps rely on matrices delineating the relationships between Skill/System/Resource/Location/Objective/Guardian/Client/Time Limit/Complication.

Mission

Flow



Tasks

- Need a complete list of primary skills and their associated mission objectives
- Need to generate conditional matrices for each of the mission generation flow steps.
- Need to tie in missions the mission board system.

To be addressed

Below is a list of areas that still need additional design work before this section can be sent to production.

- The Link/Complication system for connecting missions needs to be designed.
- The skill/objectives list needs to be complete once primary skills have been decided upon.

- Provide a logical extension to the Wing Commander space combat experience.
- Provide a strong teamwork experience.
- Provide a familiar environment for players in cooperative worlds.
- Ensure new players can enjoy tactical and combat without having to fly a ship.
- Provide another methodology for character specialization and identity.

Vision Points:

- Should have an intuitive and exciting interface.
- Should use the mouse as the primary input device.
- Should allow players to specialize and learn to master in various roles.
- Should provide innovative ways to interact with the game world.
- Should provide additional challenges and rewards to the space combat experience.

Walkthroughs

To be covered in this document:

- What is a Capital Ship?
- What are the Crew Stations on a Capital Ship?
- Capital Ship Play Experience
- How do I Obtain a Capital Ship?
- Capital Ship Combat
- Systems Design
- Capital Ship Interface Design and Requirements

What is a Capital Ship?

Capital Ships are large vessels designed primarily for military use. They range in size from the relatively small Corvette to the massive Supercarriers. Though possessing more firepower and resources than the tactical spacecraft of the previous Wing Commander titles, Capital Ships have much less speed and maneuverability.

Departing from the previous Wing Commander paradigm, Capital Ships will be very difficult to cripple or destroy, requiring the coordinated efforts of many players to successfully engage. Since a Capital Ship will rarely be reported in an attack, it also represents excellent salvage potential for those capable of defeating one.

Capital Ships

Overview:

Capital Ships are large, heavily armed spacecraft. In *Wing Commander Online*, these ships can provide a fundamental community-building mechanism, supporting cooperative activity as several players perform distinct and challenging tasks.

Vision and Goals:

Goals:

- Provide a logical extension to the Wing Commander space combat experience.
- Provide a strong teamwork mechanism.
- Provide a familiar environment for players to cooperate within.
- Ensure new players can enjoy camaraderie and combat without having to fly a ship.
- Provide another mechanism for character specialization and identity.

Vision Points:

- *Should have an intuitive and exciting interface.*
- *Should use the mouse as the primary input device.*
- *Should allow players to specialize and gain prestige in certain duties.*
- *Should provide immersive sights and sounds at all times.*
- *Should provide additional levels of depth and retention to the space combat experience.*

Walkthrough:

To be covered in this document:

- What is a Capital Ship?
- What are the Crew Stations on a Capital Ship?
- Capital Ship Play Experience
- How do I Obtain a Capital Ship?
- Capital Ship Combat
- Systems Design
- Capital Ship Interface Design and Requirements

What is a Capital Ship?

Capital Ships are large vessels designed primarily for military use. They range in size from the relatively small Corvettes to the massive Supercarriers. Though possessing more far more range and firepower than the tactical spacecraft of the previous *Wing Commander* titles, Capital Ships have much less speed and maneuverability.

Departing from the previous *Wing Commander* paradigm, Capital Ships will be very difficult to cripple or destroy, requiring the coordinated efforts of many players to successfully engage. Since a Capital Ship will rarely be vaporized in an attack, it also represents excellent salvage potential for those capable of defeating one.

What are the Crew Stations on a Capital Ship?

Capital Ships provide 4 unique Crew Stations, each providing unique contributions to the overall operation of the ship:

Captain: Responsible for coordinating the other bridge stations and issuing simple commands to the other Stations and Escort ships. *Captains* are responsible for assigning other players to Crew Stations and organic support spacecraft, and can view the current operational status of each crew station via the Sensor Display screen. Captains can override a specific station in the event of an emergency.

Captains are also responsible for the ship's navigation and primary weaponry. By using the 3D Sensor Display interface, they can enter multiple waypoints, targets, speeds, and actions via a simple point and click interface.

Engineer: Allocates energy via interlinked sliding scales to different ship systems including *Engines, Shields, Main Guns, Point Defense, Electronic Countermeasures, and Sensors.*

(The settings the *Engineer* provides for these systems reflect immediately on the interfaces of the other Crew Stations.)

The *Engineer* is also responsible for assigning *Damage Control Teams* and *Repair Bots* to specific systems. This is done via a simple pop-out interface screen by clicking and dragging the *Wrench* and *Bot* icons to the affected system.

Finally, the *Engineer* can control the *Quadrant Power* applied to the Shields and Point Defense through a basic graphic interface.

Tactical Officer: The Tactical Officer position is one of the most important during battles, providing an involving and challenging real-time strategic aspect to the Capital Ship environment. This position could be considered similar to that of a real-world AWACS aircraft, providing the "Big Picture" to the Cap's tactical spacecraft.

The "Tac" is responsible for directing the Cap's associated fighters, bombers and supplemental weapon systems. Using a simple point-and-click interface, he can *Assign* targets, routes, and other actions to friendly individual tactical spacecraft, as well as issue general orders to groups. He can also provide assistance to the supported tactical craft by *Warning* them of incoming threats, towing them in with a *Tractor Beam*, or dispatching *Turret Bots* or *Repair Bots* protect and assist allied spacecraft.

Turret Gunner: This is the simplest of the active Bridge Crew stations, using a basic "gun sight" interface controllable through a mouse or joystick. The Turret Gunner position is ideally suited to new players who want to enjoy the camaraderie and fun of being part of a Cap Ship crew, but may not have enough experience to operate one of the other stations.

The Turret Gunner control interface will be identical to that of the previous Wing Commander titles, with an exception possibly being the requirement for two controller buttons, one to fire the guns, and the second to activate a secondary Turret function or weapon.

(Note that Turrets not currently occupied by a player will resort to automatic control, responding based upon their quality level, damage state, and available power.)

Observer: This position allows a player to view all of the Bridge activities, and communicate with other Bridge Crew, but can't provide any input to the Crew Stations, or make any comments that appear on the Sensor Display.

Play Experience:

The *Pegasus* was a Light Cruiser, owned by Macrosoft-Boarland Industries. Commanded by Captain L.T. "Whisper" Slocum, her crew currently consisted of 7 other players.

Pirates had been hitting Mac-Boar's shipping routes hard during the past few days. As a result, the COO ordered the *Pegasus* to the Nimoy system to ensure the day's convoy of transports made it through the Jump Gate.

Though the *Pegasus* wasn't a Carrier per se', she had two light fighters crammed in her docking bay, awaiting any sign of trouble. The crew's two best pilots stood by in the Ready Room as Observers, watching the scanner displays and talking about the last port call they barely remembered.

Engineering was currently manned by a real comedian of a player who went by the callsign of Spotty. He'd spent most of the patrol telling jokes involving flatulence until the Captain threatened to *Relieve* him of duty. The Tactical Officer station was manned by a veteran player known as Walt. Two other brand-new players were currently in Observer positions, but would be the first to take over any Point Defense Turrets if need be.

All of the crew viewed common displays, but only the ones assigned to the appropriate stations could make adjustments and issue commands via the appropriate pop-out interface screens. Of course, everyone interjected opinions when they felt they were needed. With a group of heavily loaded transports steaming their way, everyone's eyes were on the sensors screens.

Suddenly, a contact showed up well inside the radar's range, surprising the entire crew.

"Where did that come from?" the Captain demanded.

The Engineer confirmed the Sensor Power Allocation had been set to maximum, while the Tactical Officer clicked on the contact only to have it analyzed as "Unknown."

"Stealth," the Tactical Officer noted. Pirates were known to use low-observable spacecraft a great deal.

"Launch Fighters," the Captain ordered, now using the iconic battle language to save time and confusion. Seconds later, Alpha 1 and 2 roared away from their moorings toward the contact, their icons appearing on the Bridge's radar display for all to see.

The two players in the Observer stations suddenly found themselves manning the controls of the dorsal and belly Point Defense turrets. Experienced players would have manned the stations themselves out of reflex, but the Captain didn't mind

assigning them himself. He was hoping these two newbies would show him something if a fight took place.

The Captain laid in an intercept course for the Unknown, the engines thrumming as the *Pegasus* increased speed.

"Negative contact," Alpha 1 reported, the words appearing over the icon representing his ship in the radar display. That statement was redundant, since a dashed line would have been drawn between his icon and the Unknown if he'd gotten a lock. The light fighters were designed for speed and agility, but had short-ranged sensors. Still, they should be able to pick up something...

Another warning siren blared as two more contacts appeared, these moving towards the convoy. Again, Type Unknown.

"Focus Fighter Here," the Captain input, but the Tactical Officer had read his mind and given the fighters the location of the new contacts, with orders to "Attack this target."

Alpha flight went into Emergency Power and sped towards the new contacts, while the first Unknown began to move towards the *Pegasus*.

"Increase Shields—Increase Guns," the Captain said to the Engineer. The *Pegasus* slowed as Spotty reduced the Engine power in exchange.

"What is that??" the Tactical Officer exclaimed as the Unknown came into range of the Optical sensors. The target camera showed a strange object, larger than the *Pegasus*, but with a translucent, scintillating superstructure and pulsating blue cylinder at the center.

The Missile Warning Radar system now blared, showing four incoming projectiles coming from the Unknown.

"Evasive," the Captain ordered, while most of the crew now peered out the "Window," the view of real space, to see what these projectiles looked like. The Engineer allocated maximum power to the forward shields.

The Point Defense turrets started barking as the newbies tried to pull off a lucky shot and bring down one of the incoming missiles. The Captain nodded with approval as the dorsal turret gunner actually tagged two of the inbounds, sending them careening off into space out of control.

The Tactical Officer now had his hands full as yet another Unknown appeared behind the speeding Alpha flight. Right-clicking on the new contact, the two pilots of Alpha flight saw the warning automatically in their cockpits. These guys were good, with Alpha 2 breaking off to come around and face the new threat.

The two inbound missiles hit the shields of the *Pegasus* in the frontal quadrant. The Bridge shuddered with the impact, as the shield power was completely depleted. Though the new Silksteel armor helped keep the ship in one piece, the radar was knocked offline.

"Crap!" screamed the Tactical Officer. Alpha flight was now officially hung out to dry.

The Captain was entering the order to repair the damaged sensors, but he could see the Engineer already allocating Damage Control to the job.

"Two minutes!" Spotty estimated.

Though the radar was out, the Optical sensors were enough to bring the weapons to bear. The Pegasus returned fire with her Main Guns, two large Prismatic Lasers mounted on separate turrets. The rainbow beams lanced out, one of them striking the target dead center. The Unknown flexed and shimmered, veering off in a flanking maneuver.

Alpha flight was officially starting to hate life. Without the radar coverage from the Pegasus, the two tiny fighters faced three bizarre-looking craft with phenomenal maneuverability and acceleration. It was all the pilots could do to keep the unknowns from zocking onto their six, never mind keeping them away from the convoy of transports.

Alpha 2 was having the toughest time. Not only was he the least experienced of the pilots, he was closest to the third unknown that had appeared behind them while they were after the other two. Things got really depressing when the first volley of fire from his pursuer went completely through his aft shields and took a big chunk out of his armor. Twisting and turning as hard as he could, the unknown leisurely stuck with his every move.

"Need help?" came a call over the common frequency. Alpha 2's communications screen showed the face of a human pilot wearing tacky sunglasses.

"Clear my six!" Alpha 2 said with a single key press, while Alpha 1, having been automatically warned of 2's desperate situation, tried to wheel around and help his wingman.

"Inbound," replied the stranger. Alpha 2 noted by his radar display that the new arrival was off to the right, a slick-looking customized bomber.

Alpha 1 had his wingman's pursuer in sight now, and squeezed off a couple of shots from his Ion Blaster while dodging fire from his pursuers. At least they were pulling them away from the convoy.

"Duck," came a call from the bomber, as three missiles streaked towards the unknowns.

"Radar's back," the Engineer said, as the Tactical Officer viewed the situation with the fighters. *Where'd in the hell did that bomber come from??*

The Main Guns kept pounding at the big Unknown, but the Captain had to turn the Pegasus away when another volley of strange missiles blew gaping holes in the frontal armor sections, bouncing the crew around.

"Damn it, spilled my coffee," Spotty lamented.

"Turret Bots Front" the Captain ordered, seeing the Engineer instantly comply. Small robots carrying light lasers zipped away from their docking racks and took positions on Pegasus' nose. The Navigator laid in an Intercept course and increased speed to maximum.

"Get as close as possible, and use the C-Gun," the Captain typed in the text box. The massive rail gun required the Pegasus be aimed straight at its target, but packed a monstrous punch.

More strange missiles came, but the turret gunners and bots made short work of them. The Prismatic Lasers were now scoring regularly as the Unknown slowed.

Seconds later, the strange object was straight off the bow, staring down the glowing barrel of the C-Gun. One projectile, no larger than a tennis ball, tore space itself as it smashed into the Unknown at nearly the speed of light. The recoil of firing slowed the Pegasus to a crawl, but the Unknown now tumbled through space, spewing a strange clear liquid. The sparkling blue cylinder was extinguished.

The Tactical Officer hadn't seen any of this, but was instead trying to help Alpha flight and the new bomber deal with their opponents.

"Move Here," the Captain ordered, indicating the dogfight as the destination, but the Pegasus was taking too long to accelerate. The fight would be over before she could provide any fire support.

Fortunately, the bomber's missiles had been more than helpful enough. Each unknown had taken a direct hit. But instead of inflicting damage, the EMP warheads of the missiles had shorted the unknowns' power grid, leaving them momentarily paralyzed and cartwheeling out of control through space.

Alpha flight didn't need any prodding. Within seconds, they were pounding on the target of their choice at close range. One, then two unknowns dimmed and spewed liquid, while the bomber's guns made short work out of the third just as it seemed to be recovering.

"Thanks!" Alpha 1 beamed over to the bomber as he flew up beside it. His heart jumped into his throat as he saw the White Sun logo on the bomber's fuselage. This was a pirate ship!

"Truce," the bomber pilot said. "I just want the salvage."

"Truce??" the Captain asked incredulously, monitoring the conversation. "We're supposed to shoot them on sight."

"If it weren't for that guy, Alpha flight would be toast. Probably the convoy, too."

Seconds passed before the Captain issued the "Recall Fighters" order.

"The loot's yours. Don't let me see you again," the Captain said to the bomber, but added a wink.

"You're welcome," the bomber pilot replied, already opening his salvage module and scooping up chunks of the viscous debris from the unknowns.

"Why do I get the feeling White Sun will have some new technology soon?" the Navigator asked, knowing the answer. Though the Captain was torn between scooping up some salvage for himself and making sure the convoy made it through, he knew that accomplishing this mission was his only objective. One day, they'd see those Unknowns again.

How Do I Obtain a Capital Ship?

Even the smallest Cap Ships represent an enormous investment. This makes obtaining them an undertaking only groups of players or companies can afford. (Obtaining Capital Ships should be considered the equivalent level of achievement as owning a castle in UO.) In addition to the monetary requirement, certain reputation and certification requirements are also considerations before the purchase request is approved by the shipyard.

New Cap Ships must be ordered from a shipyard capable of laying a keel of the specified size and technical level. Cap Ships are made to order, and can't be purchased through normal ship-sales venues. The cost and construction time for the ship depends on the capacity of the shipyard, the availability of the associated resources, and the size/complexity of the desired ship. Payment in advance is required for all orders.

Once the ship is finished, the purchasing party is immediately notified. Legitimate shipyards automatically register all ships constructed at their facility, so the payment of taxes may also be required prior to taking possession of the ship. (The provision of at least one player qualified to serve as Captain may also be required.)

In the unlikely event a shipyard is destroyed while a ship is under construction, that ship is also destroyed. (In this event, the purchase money is automatically refunded to the buying party.) All shipyards capable of constructing Capital Ships are located in Confed-patrolled Safe Space, and are always heavily defended.

Capital Ship Combat:

Combat between Capital Ships will initially take place at longer ranges, as they possess more powerful scanners and longer-ranged, more accurate weapons than smaller vessels. The goal is to develop an environment where the ships have to gradually close with one another as their sensors and fire control systems are damaged and engage at close ranges in spectacular broadside volleys.

Capital Ship Gunnery:

Main Guns:

The Main Guns are the Cap's primary armament for engaging large targets such as other capital ships. These consist of large turreted energy and mass-driver weapons, torpedoes, or enormous keel-mounted "Planet Killers." Main Guns possess greater range than their fighter-mounted counterparts, due not only to the larger size of the weapons, but also the greater available power and more stable firing platform. The Main Guns are controlled by the Captain through the *Weapons Status Display* interface.

Main Gun Turrets traditionally offer two identical alternately firing weapons in a single housing. Turreted weapons offer the benefits of flexibility, but are heavy, expensive, and vulnerable. As with the smaller Point-Defense Turrets, Main Gun Turrets come in a variety of types, each offering various slew rates, elevation limits, and protection.

Keel mounting is reserved for very large weapons that would be unfeasible to mount on a turret. (Note that some smaller ships must use keel mounting for weapons that a large ship could mount in a turret.) Just like fighter-mounted weapons, keel-mounted weapons on Cap Ships require the firing ship to be pointed directly at the target to score a hit. (Ensuring the target must be virtually immobile to successfully engage.)

Torpedoes are larger versions of ship-ship missiles designed for use against large targets. Slower and less agile than their smaller counterparts, they're far more durable and carry a much larger warhead. (Advanced torpedoes can possess their own shields, ECM, and armor.) Torpedoes are launched vertically from

fixed launch tubes inside the ship. (They can deploy from the top or bottom of the ship, depending on the location of the target.)

Much like Point Defense Turrets, Main Guns can be fired via two methods: Manual (Player Controlled), and Automatic. Turrets have Firing Arcs, azimuth and elevation values relative to the Cap Ship's beam line that dictate either the limits of articulation, or a software safety that prevents the weapon from shooting the ship itself. These Firing Arcs vary based on the Hull Size, Turret Location, and Turret Type.

Mass-Driver Main Guns can also be loaded with Flak and Flechette rounds to provide improved anti-missile and anti-fighter capability in a defensive posture.

Barrage Settings:

To improve their short-range defensive capability, Main Gun Turrets can be ordered to fire in Barrage Mode. Mass-Driver Turrets will automatically be loaded with either Flak or Flechette rounds if available, and Energy Weapon Turrets will be set to ¼ power in order to increase their rate of fire. The turrets will begin to fire in coordinated sectors around the ship to provide an additional layer of defensive fire against enemy fighters and weapons.

Point Defense Turrets:

Point Defense consists of small turret-mounted weapons capable of engaging close-range targets with high rates of fire. These weapons can be guns, or short-ranged missile launchers. Most Cap Ships have several, with the largest having dozens.

Players can also directly control Point Defense turrets, although the character avatar is considered to be doing this through a remote control terminal from the Bridge, NOT sitting inside the turret itself.

Also see the *Turret Gunner* station of the Bridge Crew section above.

Fighter/Bomber Support:

Most Capital Ships are able to launch and recover tactical spacecraft such as Fighters and Bombers. The "pilots" for these craft can be chosen from any player on the ship, and can be Assigned in the same manner as a normal Crew Station. (The fighter/bomber is launched as soon as a pilot is Assigned.) These craft are then supported by the Capital Ship through the Tactical Officer, and are capable of providing long-range coordinated offensive and defensive firepower to the battle.

Once these fighter/bombers return, they can be rearmed, refueled, and repaired inside the Capital Ship's docking bays. (The time required for this depends on the configuration of the Capital Ship, as well as the extent of the work required on the fighter/bomber.)

The recovery mechanisms for fighters are automatic, only requiring a fighter/bomber get within a specific range before relinquishing control to the computer.

Capital Ships can also receive Support from allied but external units (spacecraft not normally carried within the Capital Ship). This allows an external craft to offer Support, or the Capital Ship to request it. Once both ships have agreed,

the Supporting craft would be handled in exactly the same manner as the Cap's organic roster of fighter/bombers by the Tactical Officer.

Also see the *Tactical Officer* station of the Bridge Crew section above.

Systems Design

The primary systems and interfaces required to support Capital Ships will be:

- a) *Sensor Display*
- b) *Power Allocation Display*
- c) *Damage Control Display*
- d) *Quadrant Control Display*
- e) *Roster Display*
- f) *Fighter/Bomber Roster Display*
- g) *Battle Camera*
- h) *Iconic Battle Language*
- i) *Text Window*
- j) *Weapons Status Display*

Sensor Display: This will be the primary interface used by the Captain and Tactical Officer, providing the "Big Picture" of the area around the ship. This will be a represented by a spherical "holographic" display that allows players to pan, rotate, and zoom the viewing area.

The Sensor Display represents the extent of the Cap Ship's sensor range. (Note that this range will vary depending upon the type of sensors equipped, their condition, and the amount of power the Engineer has allocated to them.)

Detectable objects, navigation waypoints, and other pertinent information will be shown inside the Sensor Display. Filters will be applicable to allow players to clarify the Display to suit their preferences and situation. Bandwidth permitting, players will be able to *Highlight* a specified area by pointing to it with their mouse cursor and transmitting that cursor location to other players.

The presentation paradigm for this display will be that of a holographic projection on top of a table, with other chairs around the table representing the other Bridge Crew stations.

Power Allocation Display: This is one of the pop-out Displays used by the Engineer. This display is a series of interlinked sliders that allow the player to adjust the amount of power from the Cap's reactor to be applied to the various systems.

Damage Control Display: This is another pop-out Display used by the Engineer, allowing him to assign Damage Control Teams and Repair Bots to sections of the ship needing attention via a simple click-and-drag interface.

Quadrant Control Display: This is the final Engineering-specific display, allowing the player to allocate power to the Shield Generators and Point Defense Turrets via a quadrant-based interface. Turret Bots, mobile point-defense devices, can also be allocated to specific quadrants through this display.

Roster Display: This display shows the names and locations of all players on the Cap Ship. Clicking on the player name will either take the viewer directly to the interface the listed player is manning, or bring up a Battle Camera view of that player's craft.

Fighter/Bomber Roster Display: A list of the active Fighter and Bomber craft associated with the Capital Ship, including their status and current pilot, if applicable. Left-

clicking on a ship listing in this Roster will show the craft in the Battle Camera, provided it's in action, as well as highlight that ship's icon in the Sensor Display.

Battle Camera: This pop-out view can be used to provide a close-up real-world view of a selected target.

Iconic Battle Language: Players serving on the Bridge crew can utilize a symbolic language-independent method of communication. This allows players of any nationality to communicate battle-specific issues to one another via the Bridge Crew interface in a quick, bandwidth-friendly method. Voice fonts, resident on the client, will be played during these iconic messages to reinforce the intent. These icons will be used primarily by the player serving as Captain to coordinate the other crewmembers' efforts, but can be used by other crewmembers with certain limitations consistent with their position.

The icons will be context-sensitive, dimming themselves (becoming unavailable for selection) when a situation not appropriate to their use exists. (i.e., "Move Shields")

The iconic language focuses on Actions, Objects, Modifiers, Systems, and Locations as per the following table:

<u>Action</u>	<u>Object</u>	<u>Modifier</u>	<u>System</u>	<u>Location</u>
Abandon Ship (Mouse Target)		All	ECM	(Cursor Location)
Attack	Speed	Back/Rear	Engines	
Capture		Forward	Hull	
Decrease		Left	Main Guns	
Defend		Right	Navigation	
Deploy		Up	Point Defense	
Dock		Down	Sensors	
Focus			Shields	
Ignore			Turret #	
Increase			Landing Bay	
Intercept			Mines	
Look			Cargo	
Move			Tractor Beam	
Recall			Repair Bot	
Repair			Rescue Bot	
Rescue			Turret Bot	
Retreat				
Roll				
Steady				
Jettison				
Tractor				

Text Window: This display is where players can communicate with Rich Text Format messages. Players communicating from tactical craft will have their messages color-coded to match their status. Clicking on the particular message will perform the same function as clicking on the corresponding player's name in the Roster Display. A preferences menu will allow players to customize their fonts, text color, etc.

Weapons Status Display: This display is used by the Captain for controlling the Main Guns of the Capital Ship.

Window View: In addition to the Sensor Display, players can also utilize the Window View to look outside the ship into "real" space. This viewpoint will be similar to that of

the bridge in *Star Trek*, where each player's vantage point will be slightly different depending upon the Crew Station he's occupying. For instance, if the player is serving as Captain, then he'll be able to see the Window View in the background, with the backs of the player avatars "manning" the appropriate stations in front of him. These avatars will have physical properties to them in that if the ship takes a hard hit, they'll sway and bob to reflect the impact.

Interface Design

Sensor Display View Operation and Functions:

- **Pan and Rotate View:** Use the mouse cursor and left click to "grab" the edge of the sphere and then move the mouse to slew the view to the desired point.
- **Zoom:** The mouse scroll wheel or the +/- icon will zoom the view in or out.
- **Filters:** A drop-down menu based on Windows-style check boxes will toggle these view features:
 - a) View Only Friendlies
 - b) View Only Enemies
 - c) View All
 - d) Damage/Shield Indicator Bars
 - e) Missiles
 - f) Routes/Waypoints
 - g) Friendly Targeting Status
 - h) Enemy Targeting Status*
 - i) Word Balloons: (These text boxes are displayed over Friendly icons on the Sensor View when they transmit a message. These are truncated versions of the full text message that appears in the dialog box.)

* This view toggle requires an additional Ship Component (Signal Intelligence Module) to display.

Art Requirements:

- a) Rendered background scene
- b) Icons for the display
- c) Control icons and cursors
- d) Crew-specific mouse cursors

Sound Requirements:

- a) Ambient and environmental sounds
- b) Control input sound effects

Text Window Operation and Functions:

The Text Window interface is a simple dialog box with an entry field at the bottom and a scroll mechanism. Players can utilize this interface to enter simple text messages.

Messages from players that are flying external support aircraft such as Fighters and Bombers are color-coded to reflect the status of the particular ship.

Additional messages will be displayed by the system:

- Destruction or departure of a supporting ship.

- Arrival of a supporting ship.
- Launch of a supporting ship.
- Recovery of a supporting ship.
- Communications from an enemy or neutral ship.
- Changes in the Cap's system status.

Art Requirements: Fonts

Sound Requirements:

- a) Input sound effects
- b) Voice fonts for system messages

Station Selection:

Crew Stations are selected by placing the mouse cursor over an unoccupied station and left-clicking. Provided the player meets the requirements for that Station, he's zoomed into the appropriate console interface.

The Captain can place players into specific stations or docked spacecraft by clicking on the appropriate name in the Roster interface, and dragging that name onto the desired Station or spacecraft icon. If that Station was already occupied, the Captain is asked to confirm the assignment. (The player who had occupied that Station previously is automatically moved to an Observer role in that event.)

Art Requirements: Special mouse cursor

Sound Requirements: Voice font to reinforce the Assignment function

Captain's Station Sub-Screens and Functions:

- *Iconic Interface:* The Command Icons are always visible from the Captain's Station, requiring only mouse clicks on the appropriate icon and on the relevant location or target in the Sensor Display, if required. The last five Commands the Captain issue are automatically stored at the top of the Icon panel, allowing even more rapid communication if needed.
- *Navigation:* Navigation commands are input directly onto the Sensor Display using a two-step process. Once the Navigation Icon has been selected, the Navigation Panel will enlarge. The Captain will input the "horizontal" route component of the waypoint by moving the mouse to the desired location on the Sensor Display and left-clicking. After that, the Captain enters the vertical component in the exact same manner.

The default Route Speed is always the Ship's Best Speed, unless the Captain elects to Modify the Waypoint Speed by right-clicking on the waypoint, or simply moving the speed slider on the Navigation Panel.

Additional navigation commands can be entered via a supplemental right-click menu:

- a) *Move to Intercept: (Designated Target)*
- b) *Match Speed: (Designated Target)*
- c) *Match Heading: (Designated Target)*
- d) *Match All: (Designated Target)*
- e) *Evasive Maneuvers*
- f) *Z-up*

- g) Z-down
- h) Fast Roll

- **Weapons Status:** This interface allows the Captain to view the status and control the firing of the Main Guns. The Captain need only to select the weapon(s) of choice by left-clicking as per the Windows method, and then left-click on the target in the Sensor Display. Weapons can also be set to Overcharge and Barrage Modes via the Weapons Status sub-screen.
- **Override:** The Captain can Override another's player control of a Station by double-left-clicking on the appropriate station. A dialog box then appears asking the Captain to confirm the Override. At this confirmation, the Captain assumes direct control of that station in addition to the Captain's Station, which is then minimized. The player who had previously occupied that Station becomes an Observer.
- **Lock View:** To ensure that the entire Bridge Crew can see where he's looking, the Captain can elect to Lock the View of the other Crew Stations. This will immediately set all the Stations to the Sensor Display zoom and orientation the Captain is using, highlight any Target(s) the Captain has selected, and display his mouse cursor position within the Sensor Display.

Art Requirements:

- a) Mouse cursors and icons for each function
- b) Symbols for the iconic language

Sound Requirements:

- a) Effects for iconic language input
- b) Weapon sound effects
- c) Engine sound effects
- d) Control input sound effects

Tactical Officer's Sub-Screens and Functions:

Issuing Orders for Supporting Fighters/Bombers:

- **Assign Target:** By clicking on the icon for a particular friendly Fighter/Bomber in the Sensor display, the Tac can issue basic commands to the player inside. By left-clicking on the friendly Fighter and then left-clicking on an enemy target, the player in the fighter hears a voice prompt and has his sensor screen automatically update to prioritize the new target.
- **Warn:** To warn a friendly Fighter about an enemy craft or object, the Tac would left-click on the Friendly and then right-click on the enemy/object.
- **Assign Route:** To give a friendly Fighter a new route, the Tac would LC on the Friendly's icon and then click on any area of space not occupied by an enemy/object in the same manner as the Navigator inputs waypoints for the Cap. Multiple waypoints can be entered in the same manner as the Captain would use to lay in a course for the Cap Ship.
- **Defend:** To order a Fighter to Defend a friendly target, the Tac would LC on the Fighter and then RC on the object to be defended.

Menu Orders:

More generalized orders can be issued by right-clicking on a specific Friendly or Group. An iconic menu will then be displayed with the following orders:

- a) Attack Fighters
- b) Attack Bombers
- c) Attack Capital Ships
- d) Defend Mother Ship
- e) Dock Immediately
- f) Retreat
- g) Fire at Will
- h) Pursue
- i) Don't Pursue

The Tac can also form *Groups* via the familiar interface of holding the left-click down, "drawing" a box around the Friendlies he wishes to Group together, then clicking the "Group #" button or using the CTRL-# interface from games such as *Command and Conquer*.

As a default, all Fighters belonging to the Mother Ship (the Cap occupied by this particular Tac) will automatically be assigned to Group 1, and Bombers belonging to the Mother Ship will be assigned to Group 2.

Art Requirements: Cursors and icons for each different order

Sound Requirements:

- a) Control input sound effects
- b) Order acknowledgement voice fonts
- c) Warning sound effects

Engineer's Sub-Screens and Functions:

- *Energy Allocation Screen:* This screen allows the Engineer to adjust the percentage of the Cap's total reactor power applied to each major system:

- a) Engines
- b) Shields
- c) Main Guns
- d) Point Defense
- e) Electronic Countermeasures
- f) Sensors

Each system is represented by a sliding scale. The scales are completely interlinked. Adjusting one slider has an equal effect on all the other non-selected sliders. The far right of these sliders represents an "Emergency Power" range. (Note that the player's Engineering skill can vary the amount of Emergency Power that can be reliably applied to a system.)

A "Limit Bar" is applied to all sliders, showing the maximum power a specific system can accommodate. This slider can change position as a system is damaged and repaired. (Note that a player's Engineering skill also has an effect on the "Limit Bar" position.)

- *Damage Control:* This interface is a small top-down silhouette of the Cap ship, with color-coded regions reflecting the condition of that area's systems. A color-coded text listing of the Cap's system is also displayed. Icons representing the Cap's Damage Control Teams and Repair Bots are also present on this interface, allowing

the Engineer to Assign these repair assets to the appropriate system by simply left-clicking on the asset and dragging it onto the silhouette or text listing.

- **Quadrant Control Screen:** This interface consists of concentric rings divided into Front, Left, Right, and Rear quadrants. The Engineer can increase Shield and Point Defense power to these quadrants by left-clicking the Shield or Turret icon, and then left-clicking on the quadrant he wishes to increase power to. The power from the other three quadrants is then reduced equally to support this demand. Shield power is represented by the number of concentric blue rings in each quadrant. Point Defense power is represented by the brightness of the green background in each quadrant. The outermost concentric ring is color-coded to show the amount of armor protection in each quadrant, ranging from white, to yellow, orange, red, and finally black.

The Engineer can also deploy Turret Bots to specific quadrants by left-clicking the Bot icon, and left-clicking the desired quadrant.

Art Requirements:

- a) Rendered background
- b) Ship silhouette art
- c) Function-specific cursors and icons

Sound Requirements:

- a) Order acknowledgement voice fonts
- b) Engine sound effects
- c) Shield sound effects
- d) Warning sound effects
- e) Bot sound effects

Turret Gunner Station: This station is essentially a gun sight. The symbology on this screen includes a Field of Regard and Field of View indicator comprised a small indicator consisting of one rectangle inside a larger one. The smaller rectangle shows the current field of view's position, and the larger one represents the azimuth and elevation limits of the turret. Additional indicators show the weapon power and/or ammunition levels. A small 2D Sensor Display (consistent with previous *Wing Commander* titles) and a Text Message box are also components in this station. Different CCIP (Computer-Controlled Impact Point) symbology may also be displayed depending on how the Turret is configured.

Art Requirements:

- a) *Field of Regard/View symbology*
- b) *Gun sights for each different weapon*
- c) *CCIP symbology*

Sound Requirements:

- a) *Turret slewing sound effect*
- b) *Sound effects for each weapon*
- c) *Weapons impact sound effects*

Observer Station: This station is identical to the Captain's Station, but has no functionality beyond the ability to manipulate the viewpoint of the Sensor Display, and converse via the Text Message box.

Art Requirements: No unique requirement

Sound Requirements: *No unique requirement*

Tasks

Below is a top-level list of tasks necessary to complete the production of this section.

- Artwork, cursors, and icons for all interfaces must be generated.
- Sound effects
- A set of interfaces must be generated for each Crew Station.
- Code to support the functionality of each Crew Station must be generated.

To be Addressed:

- Engine Thrust Arcs
- Escort Ships
- Capture
- Retreat and Victory
- Surrender
- Support
- Honor and Prestige
- Self-Destruction/Scuttling
- Escape Pods
- Boarding Parties
- Relieving Bridge Members of Duty
- Mutiny
- Battle Damage Effects for Cap Ships
- Player Crew Station Skill Bonuses

Systems design

Aggregate Group Kill Credits

When dealing with combat and who gets the credit for the kill, the current philosophy is that the individual who does the most overall damage gains credit for the kill. This credit comes in the form of salvage privileges, accomplishment credit for certification, and any mission rewards.

Whenever individuals from a group are cooperating in combat, their damage counts are combined into a group damage count. If the group damage count is the most overall damage to a target, then the whole group gains the credit for the kill.

Kill Credit Distribution

When a grouping gains the kill credit, it is distributed to the group. The group leader determines the exact method of distribution. The following are example methods: (The complete list of distribution methods is "to-be-designed".)

Equal Share: Equal share denotes that each group member gains an exactly equal share of the kill credit, whether this is credits, accomplishment values, or other

Grouping

Overview

A collection of individuals can form a squadron (for space) or a party (for ground). This collection of individuals is defined as grouping. Additional functionality is an important method used to encourage player cooperation and grouping. Offering extended functionality is crucial to this.

Vision and goals

GOALS

- Encourage player grouping by offering additional functionality

VISION

- Encourage and support a community
- Support a wide range of in-game communications systems
- Offer robust support for players to form guilds and organizations

Walkthrough

Added grouping features allow players to form into parties or squadrons and coordinate all actions as a cohesive unit. As well, offering group aggregate kill credit and distribution encourages grouping, and makes them a viable alternative to the solo play-style.

For example, Jimmy and Philo decide to form a two-man squadron to take on pirates in a nearby tame-space sector. They decided when they initially grouped that Jimmy was the leader, and that they would split all kill credit equally. Once they arrive, they begin the systematic removal of the pirate threat. Jimmy and Philo use group chat to coordinate their attacks, and always have information on the other member's status without navigating through another interface. After each kill, all kill credit is split equally; which includes the bounty rewards on the pirates.

Systems design

Aggregate Group Kill Credit

When dealing with combat and who gets the credit for the kill, the current philosophy is that the individual who does the most overall damage gains credit for the kill. This credit comes in the form of salvage privileges, accomplishment credit for certification, and any mission rewards.

Whenever individuals from a group are cooperating in combat, their damage counts are combined into a group damage count. If the group damage count is the most overall damage to a target, then the whole group gains the credit for the kill.

Kill Credit Distribution

When a grouping gains the kill credit, it is distributed to the group. The group leader determines the exact method of distribution. The following are example methods: (The complete list of distribution methods is "to-be-designed".)

Equal Share: Equal share denotes that each group member gains an exactly equal share of the kill credit, whether this is credits, accomplishment tallies, or other

rewards. If any portion cannot be divided evenly, the remainder is awarded to the group leader to distribute at his discretion.

Percentage Share: The percentage share method distributes the kill credit based on percent of effort. If there is any excess remaining after the distribution it is awarded to the group leader to distribute at his discretion.

Group Chat

Whenever a group is formed, a separate chat channel is created specifically for this group. It will be as easily accessible and can be set as the default mode of communication when grouped. It is present to allow group members to quickly and easily communicate with each other even when separated by vast distances.

Group Status

When a group is formed, a small display will be constantly present that shows a quick summary of each group members status. In space, it will display their current overall ship status. On the ground, it will display that member's current individual status.

Squadron Flight Formations

A group leader of a squadron can give formation commands to specific group members. When the squadron leader gives a formation command, a visual guide is displayed in the specified member's Heads-Up-Display that will direct them into a specific point in a formation. Once it is formed, the group members can synchronize speeds to allow for the maintaining of the formation.

All Interface is "To-be-designed"

Tasks

- ☐ Artwork needs to be generated for the grouping interfaces.
- ☐ Code needs to be generated to handle group vs. individual kill credit.
- ☐ Code needs to be generated to handle the distribution methods.
- ☐ Artwork needs to be generated for the Squadron Formation Guide
- ☐ Code needs to be generated to implement the Squadron Formation Guide.

To be addressed

- ☐ Design the distribution methods.
- ☐ Mockup all grouping interfaces

Crafting

Overview

Crafting is intended to be a simple and visual process that has depth, and is not intended to be complex or tedious system. Players are encouraged to experiment with different materials and components using the *build templates* as recipes to constructing items.

Vision and goals

GOALS

- ☐ Simple to learn and use the interface for *build templates*.
- ☐ Offer a lot of depth without unnecessary complexity
- ☐ Create a sense of ownership

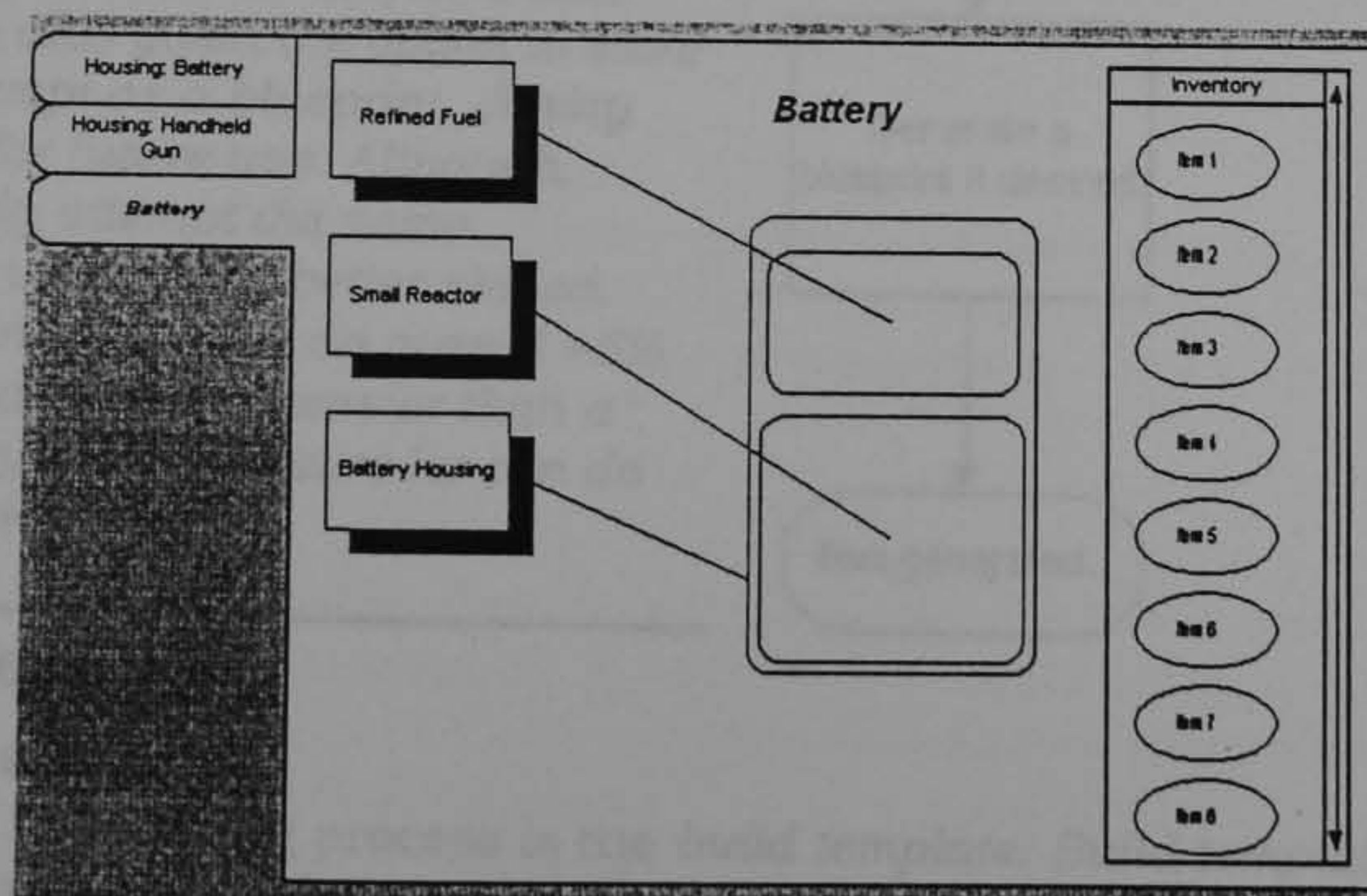
VISION

- ☐ Easily accessible to a mass market audience and a casual player
- ☐ Have an intuitive user interface
- ☐ Make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ Expose new strategies throughout the game to the player

Walkthrough

To construct an item, a player uses *build templates*. *Build templates* are acquired through gaining levels in appropriate certifications, and cannot be deleted or lost. The *build templates* are stored in the player's PDA, and are accessed through that interface. The player can only construct items based on *build templates* that they have access to, determined by their certifications.

For example, Jimmy decides he would like to build a fusion battery. So, Jimmy accesses his PDA and finds the build template for the fusion battery. If he did not have the proper certifications, the fusion battery build template would not appear in his PDA.



Once a player chooses the desired *build template*, they place the required types of components or resources into the slots that are present on the *build template*.

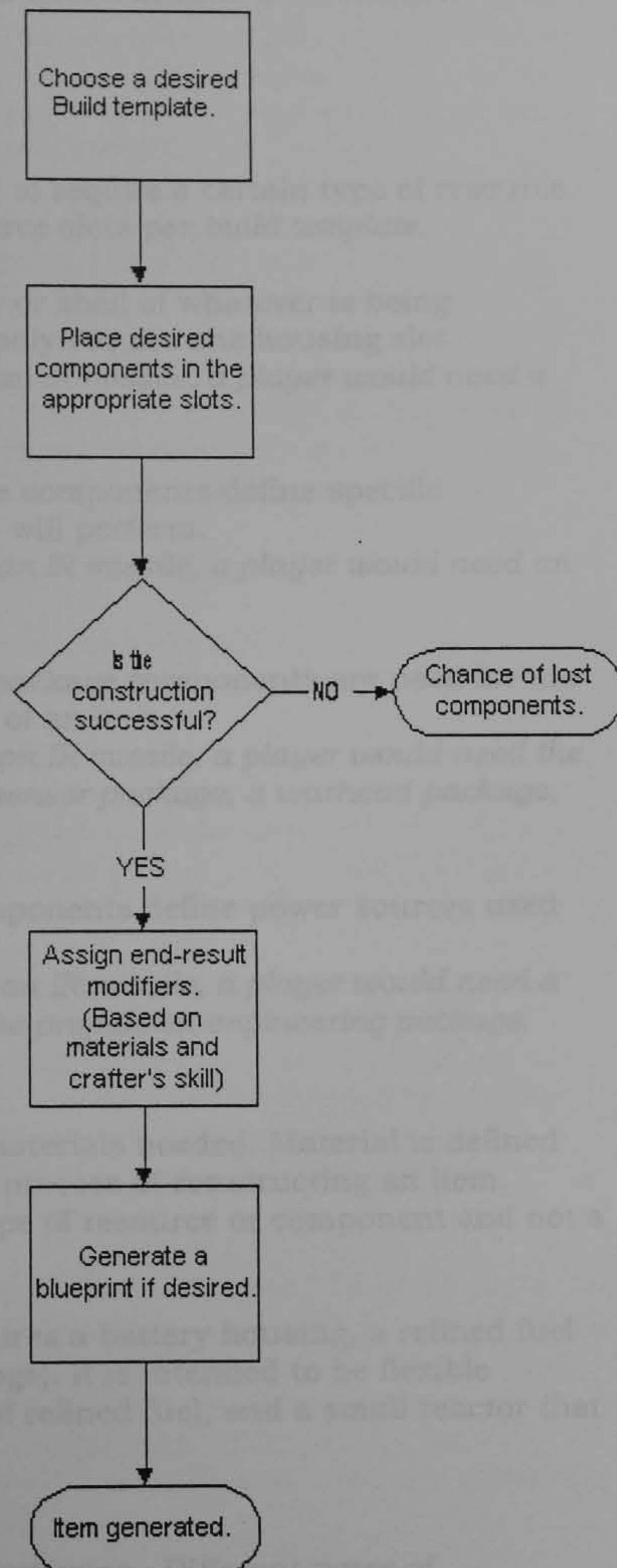
From the previous example, Jimmy selects the fusion battery build template. On this screen, Jimmy sees that there are three slots on the template. One slot is marked "battery housing", another is marked "refined fuel", and the final slot is marked "small reactor".

Each time a player successfully constructs an item using a *build template*, a set of positive and negative modifiers are assigned to that item. These modifiers are based on the key factors of materials used and the crafter's skill.

From the previous example, Jimmy decides to use a basic battery housing which has no bonuses to weight or structure. He also decides to use plutonium as the refined fuel which gives a +5% to power output. Finally, he uses a fusion reactor Mk2 which grants a +5% to power output but adds an additional 40 units of weight.

When a player successfully constructs an item using a *build template*, they are also given the choice to save this specific instance as a *blueprint*. Blueprints are used to store the modifiers received from that specific construction. This *blueprint* can be used by any player to create an item with identical modifiers, but only if they have access to the *build template* that corresponds to the *blueprint*. It can also be uploaded to an automated factory to mass-produce that item (with identical modifiers).

From the previous example, Jimmy has successfully made his first fusion battery. Unfortunately, Jimmy's certifications are at the bare minimum needed, so he receives a -5% power output modifier with his crafter's skill modifier. Jimmy is also given the option to store his successful attempt as a blueprint. Jimmy decides to save it for future use. Although, Jimmy will probably attempt the same construction again when he is better skilled. His current blueprint gives him an overall +5% to power output and is a little heavier than a store-bought battery, but he is sure he can do better in a few weeks.



Systems design

Build Templates

The cornerstone of the crafting process is the *build template*. *Build templates* are **not** items, thus they cannot be traded or sold. *Build templates* are used to experiment with configurations and to construct single instances of items.

The *build templates* are stored in the player's PDA, and are accessed through that interface. The player can only construct items based on *build templates* that they have access to, determined by their certifications.

Build templates are represented visually as a line diagram with different slots located around the diagram. These slots are where the player places appropriate materials. Every build template may use a different amount of slots but each slot belongs to a list of generic types.

Slot Types

The five different slot types are:

- Resource: A resource slot is present to require a certain type of resource. There will usually be multiple resource slots per *build template*.
- Housing: A housing is the container or shell of whatever is being constructed. An item will normally only require one housing slot.
For example, when constructing an IR missile, a player would need a missile housing.
- Software Package: Software package components define specific functions that the constructed item will perform.
For example, when constructing an IR missile, a player would need an IR software package.
- Engineering Package: Engineering package components are used for the internal electronics and mechanics of an item.
For example, when constructing an IR missile, a player would need the following engineering packages: IR sensor package, a warhead package, and a propulsion package.
- Power Package: Power package components define power sources used by the constructed item.
For example, when constructing an IR missile, a player would need a power plant package for powering the propulsion-engineering package.

Materials

Each *build template* has slots that represent the materials needed. Material is defined as any component or resource that is used in the process of constructing an item. Generally when a material is required, it is of a type of resource or component and not a specific one.

For example, a fusion battery build template requires a battery housing, a refined fuel (resource), and a small reactor (engineering package). It is intended to be flexible enough to handle any battery housing, any type of refined fuel, and a small reactor that matches the fuel type.

Raw Resources

Raw resources are the result of resource gathering. Different types of installations gather raw resources, both terrestrial and celestial. Some raw resources are used directly as materials and others must be processed into refined resources before being useful.

For example, iron ore is mined from deposits planet-side. Normally, iron ore is useless unless processed into iron by a smelting installation.

Refined Resources

Refined resources are the result of installations that take raw resources and process those resources into useful materials. This includes creating alloys and other exotic substances.

Component

Components are either store-bought or crafted materials that are used to construct items. The distinction between a component and a resource is that a component is constructed from resources or other basic components. Only the most basic components are readily available for instant purchase, a player must craft the more beneficial or more complex components.

Attribute Inheritance

Ultimately, all materials are derived from raw or refined resources. The resources have attributes which are used to compute the *materials used* modifier (discussed below).

For example, iron ore is refined into iron. Iron has a material modifier of +20% weight when used in construction. If a player refined iron ore and other raw resources into a steel alloy, the material modifier may change to a -10% weight when used in construction.

Constructed Item Modifiers

Each time a player successfully constructs an item using a *build template*, a set of positive and negative modifiers are assigned to that item. These modifiers are based on the key factors of materials used and the crafter's skill.

The *materials used* modifier represents what base material types the components were constructed from and any crafter's skill bonuses gained from the components. If only raw or refined resources are used in the construction of an item, the *materials used* modifier is based on the base materials.

The *crafter's skill* modifier is based on the certification levels of the crafting skills used in constructing the item. Depending on the level of certification, the modifier is determined from a random range of numbers that is relative to the difficulty of the item being constructed. *For example, Jimmy has a level 5 certification in Widget construction. The widget that he is attempting to build requires at least a level 3 certification to even attempt building it. His crafter's skill modifier would range from a low negative to a relatively high positive bonus. (ie. -5% to +20% modifier)*

It is important to note that a player that constructs an item using the *build template* will not receive the same end-result modifiers during consecutive, successful attempts. This is due to the crafter's skill modifier being a random range and not a fixed bonus. This allows a player to "tinker" with the same process using different components of the same type to find the ideal combination, but also allows some variation so that two different players with identical skill-sets using identical components can have different finished products (based on the modifiers assigned).

Blueprints

Once a player discovers a configuration of a crafted item that they desire, they can save that instance of a *build template* as a *blueprint*. They are used to store the modifiers received from that specific instance of crafting. To receive the modifiers from the original instance, the player must use the exact same type of materials.

For example, Jimmy crafted a fusion battery with the modifiers of +5% power output and +20% weight modifier. If he saves this successful attempt as a blueprint, those are

the modifiers he will always received when using that blueprint. The materials he used were a basic battery housing (newbie), plutonium (a refined resource), and a fusion reactor Mk2 (a crafted component). To gain the modifiers by using the blueprint, he must use the exact same type of materials in further crafting of the item.

A player who has the appropriate certifications can use a blueprint derived from the related build template. They may also upload the blueprint to a factory to mass-produce the item defined by it. In either case, the items produced will have the same constructed item modifiers as the original instance of the crafted item.

Itemization

Unlike *build templates*, *blueprints* **are** items. They can be traded on the commodities markets, given to other players, or uploaded to factories for mass production of an item.

For example, Jimmy decides to mass-produce his fusion battery. He travels to his factory and uploads the fusion battery blueprint. Once he places the appropriate materials in the storage portion of his factory, it will begin to churn out fusion batteries with the same modifiers as his original success at a fusion battery. Jimmy could also take the blueprint and sell it on the commodities market or give it to a friend to use in their factory.

Substitution

There is a process to changing the materials specified in a *blueprint*. If a player has the appropriate certifications, he can choose to substitute one material of a certain type with another. If this is done, some or all of the modifiers gained from the *blueprint* will be lost depending on several factors. These factors are "to be designed".

For example, Jimmy's blueprint requires the basic battery housing. He decides to craft a titanium battery housing and use these for his blueprint. Since it is of the same type of material, it can be substituted. But, the final product will have different modifiers than the ones produced with the basic battery housing.

Interface design

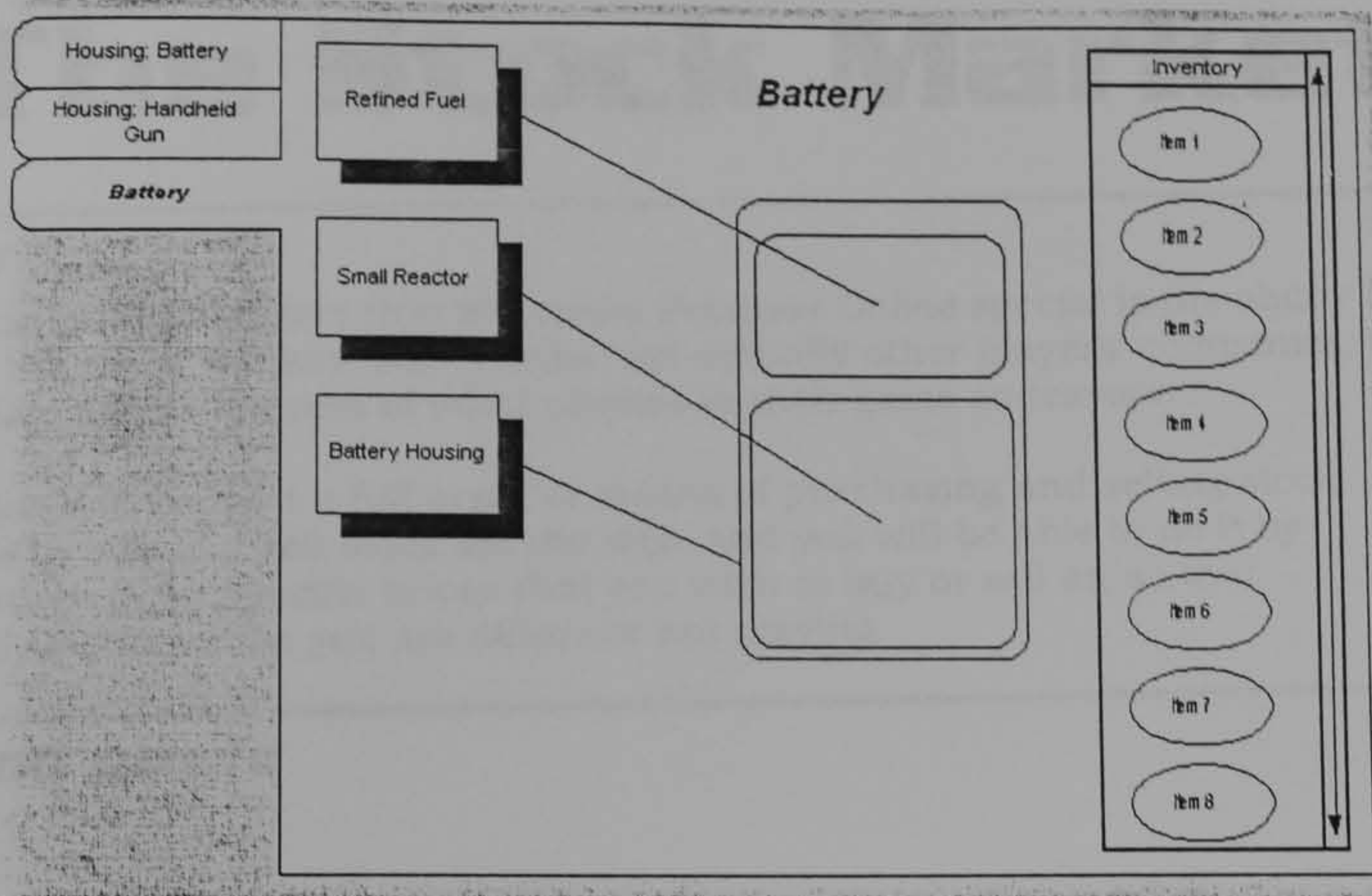
Build Template Screen

This screen is the primary interface for most crafters. It is an interactive screen used to build and/or tinker with different combinations of materials.

Along the left-hand side will be tabs that list either categories or specific build templates. On the right-hand side, a scrolling inventory bar is used to drag-and-drop into slots that represent the different materials. In the center, a line diagram represents the item to be constructed with the corresponding material slots needed.

Blueprint Screen

The *blueprint* screen is similar to the *build template* screen, except that the *blueprint* screen is non-interactive and does not show the inventory window to the right. The exception would be when a player attempts a *substitution*. In a case of *substitution*, the *blueprint* reverts to a *build template* screen to allow the player to test his substitutions. It does not actually change the *blueprint* in any way.



Tasks

- ☐ A specific list of *build templates* for all item types must be designed.
- ☐ Data must be generated on the specific *build templates*.
- ☐ A specific list of basic (newbie) components and materials must be generated.
- ☐ Data must be generated on all the material attribute modifiers.
- ☐ Data must be generated for the modifier effects of substitution.
- ☐ Data must be generated on all certification modifiers and modifier ranges.
- ☐ Code must be generated to handle the crafting system and interface.
- ☐ Art must be generated for each type of *build template*.
- ☐ Art must be generated for the *build template* interface and *blueprint* interface.
- ☐ Several balance passes must be made to get the modifiers in-line with gameplay.

To be addressed

- ☐ Substitution effects must be designed.
- ☐ More in-depth mockup of the interface screens needs to be generated.
- ☐ Data must be generated on how to handle the item management with *blueprints*.

The Stock Market

Overview

One of the special, unique twists that will make *Privateer Online* special is the ability for players to play the stock market. The "stocks" are actually other players' companies, so you can speculate on the success of other players in their game endeavors.

The stock market will support a full array of means of purchasing and selling stock. You will be able to buy and sell stock via the web, and you will be able to do it by putting limit orders in on specific prices that you wish to buy or sell at, so that transactions can occur whilst you are offline or not playing.

Vision and goals

Goals

- ☐ Provide opportunity for players to speculate in a fictional stock market.
- ☐ Provide opportunity for players to trade in commodities.
- ☐ Provide opportunity for players to create and manage fictional corporations.
- ☐ Create a simple but powerful system for stock portfolio management.
- ☐ Must be easy to research, sort, scan, buy, and sell stocks.
- ☐ Must be easy to sort, scan, buy, and sell commodities.
- ☐ Provide integrated display of news and stock ticker.
- ☐ Must have strong obscenity filtering routines for stock symbols.
- ☐ Add value to Premium Accounts

Vision Doc

- ☐ Game play can fit a short play session (< 30 minutes)
- ☐ More easily accessible to a mass market audience and a casual player
- ☐ Should provide an intuitive user interface
- ☐ Should have a login sequence that is fast and easy
- ☐ Will be playable via the Web in some fashion
- ☐ Provides a means of "playing" while offline (out of the game)
- ☐ Has a low target bandwidth
- ☐ Support as low end a machine as possible, to maximize the audience
- ☐ Should keep backend costs low
- ☐ Should have low support costs
- ☐ Should blur real life and the online character
- ☐ Should offer motives and rewards for short session play
- ☐ Should satisfy Achiever-oriented players
- ☐ Should support character building and advancement
- ☐ Should encourage a sense of player ownership in the game world
- ☐ Should have ladders and leader boards for everything in the game (, including social activities*** Can we provide a social atmosphere, a la the Exchange?)

Walkthrough

- Creating Stock
- Going Public: The IPO
- Buying and Selling Stock
- Interface Elements
- Scenarios

- Leader Boards
- Other

Creating Stock

When a Company is formed, the number of Authorized Shares is determined. Once created, the number of Authorized Shares cannot be changed in the initial version of WCOP. Future updates to the live version may make it possible to amend (private), or split/repurchase (public) the number of shares. The value of the Authorized Shares for a private company is \$0.

Helpful recommendations, such as default amounts, should be presented during the Company Formation process. Ten million shares may be a good recommended stock pool size for Authorized Shares.

The Company initially owns all of the Authorized Shares. The Guildmaster (Chairman?) can issue shares to other Guild Members (employees?) as incentives. If a Guild Member leaves a private Company (or a player deletes their character), the shares issued to them revert to Company ownership.

Authorized Shares in private company can never be traded publicly, nor can they be traded among guild members. All shares, private or public, are considered **non-voting**.

Going Public: The IPO

A Company may decide to raise capital for expansion or other purposes. One method for raising capital is through the public sale of its stock in an Initial Public Offering (IPO).

Unlike the real world, WCOP does not require an investment firm to underwrite the IPO. Instead, the company itself must underwrite the value of the issued stock and may set the IPO price to be competitive or outlandishly high, letting the market determine the price they are willing to pay.

During the initial phase of the IPO period (for the sake of argument, one game week) the stock ticker will display the IPO symbol and price at regular intervals. The IPO price will remain until the IPO period has elapsed OR until all of the IPO shares have been sold at the IPO price. A sell order at the IPO price will remain until all of the IPO shares are sold, or in the unfortunate event of a failed IPO, re-priced.

Ask transactions will need to retain the owner, ask price, and actual price of the shares. That mechanic will also allow the re-purchase of shares by the seller.

QUESTION: Without an underwriter, what happens in a failed IPO? Can the company lower the IPO price or dump their stock onto the market?

SUGGESTION: The Guildmaster, or person(s) given the authority to handle stock transactions for the company, can retract the unsold shares, re-price them, and sell them again in the open market. This may incur some direct negative affect on the status of the IPO or the company, possibly through a news event or buyback fees. We should talk with our contacts at The Motley Fool and see if they have any suggestions on how to handle failed IPOs without an underwriter, including such issues as stock buybacks and repricing.

Buying & Selling Stock

Players can access the online Stock Market through their PDA, where they may actively participate in market transactions. Players can also passively monitor trade volume by watching the Stock Ticker (see Interface, below.)

Interfaces must be provided for the following

- ☐ Buy Shares
- ☐ Sell Shares
- ☐ Symbol Research
 - Symbol Lookup
 - Stock Charting
 - News
 - Link to Web Site
 - Stock Watch List (a personalized stock ticker)

Future Interfaces may include the following:

- ☐ Ask/Bid Tracking (by symbol)
- ☐ Multi-symbol Lookup
- ☐ Portfolio Management
- ☐ Real-time Quotes / Tracking (for Premium Accounts)
- ☐ Puts and Calls
- ☐ Market Watch Tracking (i.e. DJIA, S&P 500)

Share lot sizes for Ask and Bid orders could be offered in round lots (100) or lots of 10, 25, 50, etc., however, an unrestricted lot size is preferable. In either case, an algorithm will need to be written to match and execute Ask and Bid orders in a timely manner.

Interface Elements

The Stock Ticker

A Stock Ticker is displayed on any PDA "screen" where a Stock Market interface is being presented. A typical position for the Stock Ticker would be at the bottom of the interface screen. The Stock Ticker should display the Stock Symbol and Last Trade for any stock that has been traded in volume during the day. (Display interval rules TBD.) In addition, the symbol and price for any current IPO will be shown at regular intervals until either the IPO period expires or all shares from the IPO have been sold (wherein normal trading rules resume for the stock.)

Like the real world, it is assumed that the stock ticker information will be delayed by several minutes. This provides an opportunity to upgrade Premium Accounts with real-time, in-game stock quotes.

Stock Quote

The minimal Stock Quote interface might present the Stock Symbol (as a link), the Last Sale, Net Change, % Change, and Share Volume.

Last Sale	Net Change	%Change	Share Volume
ERTS \$ 73.0625	0.125▼	0.17%	196,100

The simplest interface should provide a button to allow the player to immediately buy the stock by opening a "Buy Stock" interface. While viewing *any* interface for a specific Stock, buttons must be provided to access the Buy, Chart, News, and Web functions.

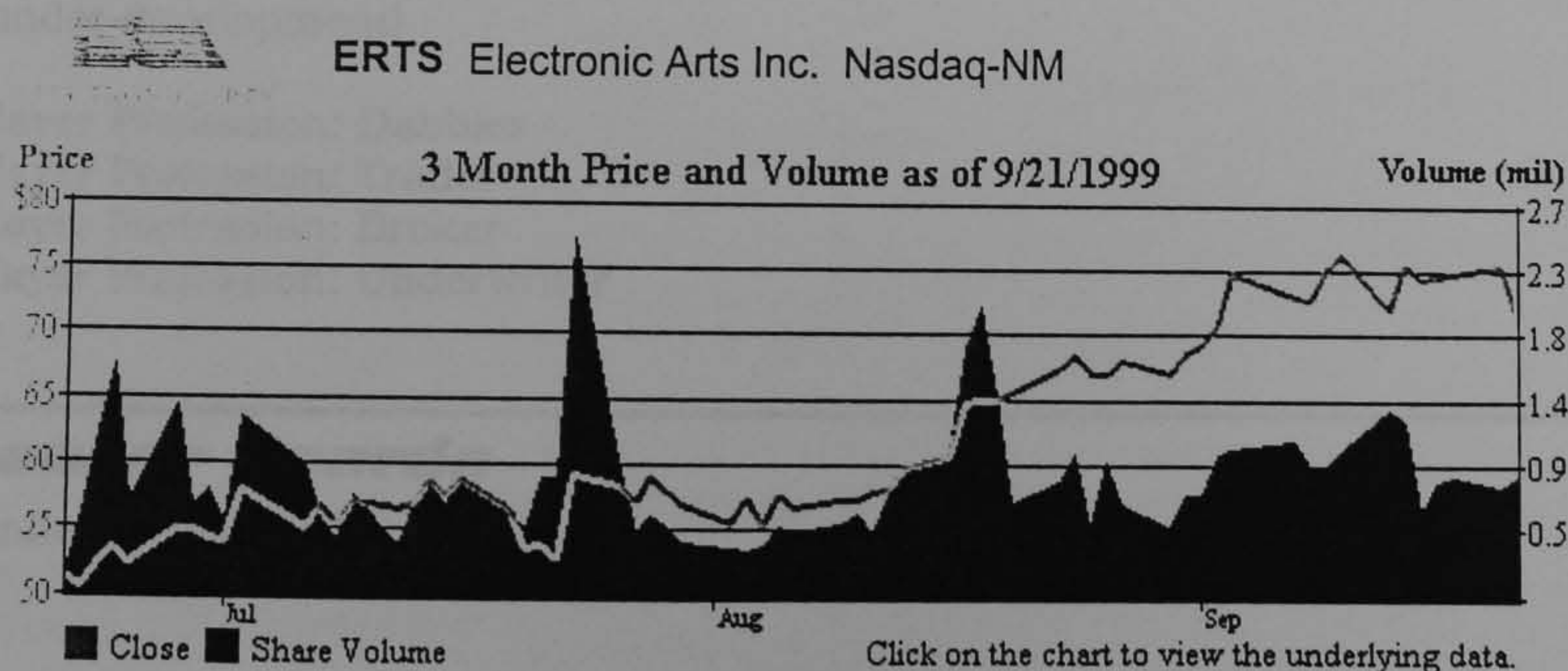
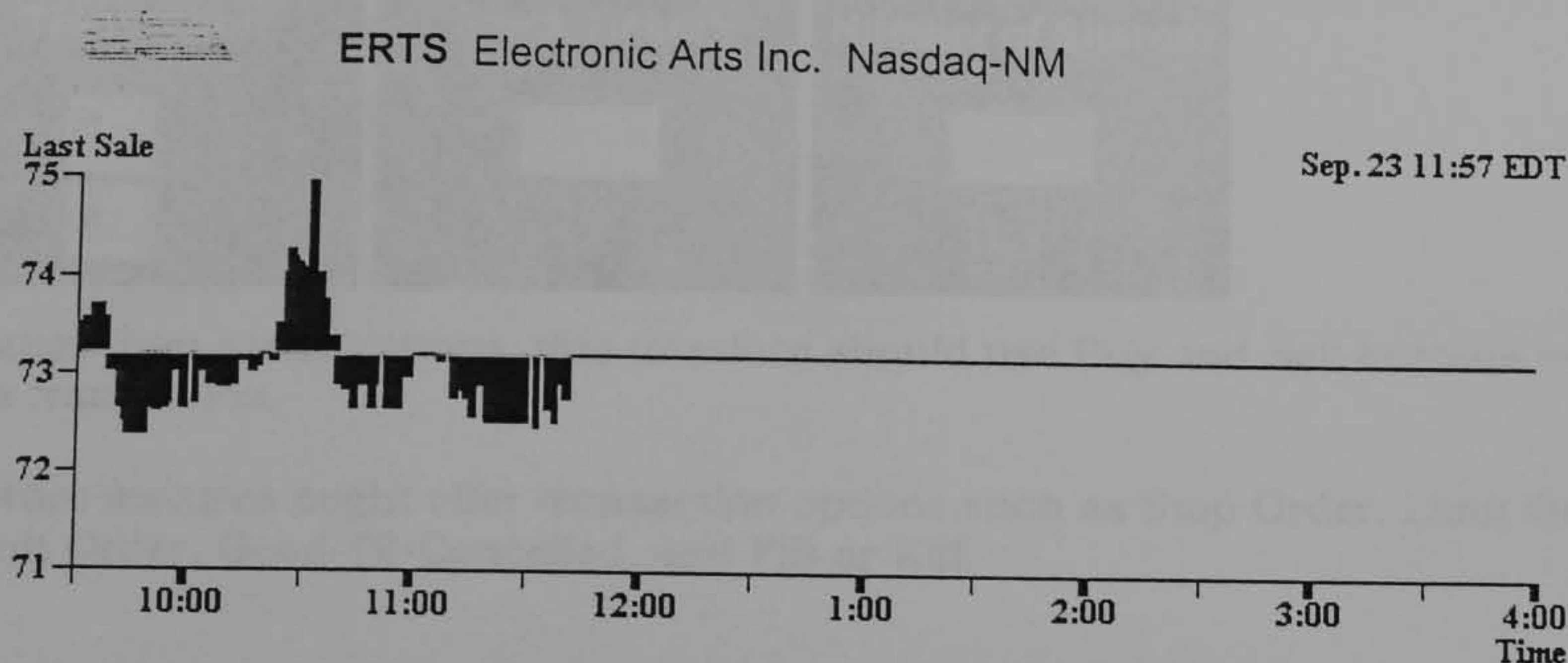
Stock Quote (continued)

A more robust interface could be presented when the player selects the Stock Symbol:

ELECTRONIC ARTS (NASDAQ:ERTS)				
<u>Bid</u> 72.25	<u>Ask</u> 72.44	<u>Bid x Ask</u> 1x3	<u>Volume</u> 203,700	<u>Last Price (Time)</u> 72.50 (11:29)
<u>Open</u> 73.50	<u>High</u> 74.97	<u>Low</u> 72.12	<u>Prev Close</u> 73.19	<u>Change (%)</u> -0.69 (-0.94%)
<u>52 Week High</u> 76.94	<u>52 Week Low</u> 33.25	<u>EPS</u> 0.00	<u>P/E</u> 64.15	<u>Shares Outstanding</u> 62,316,000
<u>Ex-Div Date</u> 00/00/00	<u>Div Frequency</u> not avail.	<u>Div Amount</u> 0.00	<u>Yield</u> 0.00	<u>Volatility</u> 35.67

Stock Charts





Intraday



3-Month

Stock Portfolio Entry

A real world example (with bells and whistles):

B/S	#	Ticker	Qty	Buy Date	Cost Basis	Current Price	Change	% Change	Gain	Market Value	Research
B	1	MSFT	1,000	09/15/99	93.6250	93.5625	-0.0625	-0.07%	-62.50	93562.50	   
Totals :										-62.50 93562.50	

- ☐ Buy more stock in a particular company.
- ☐ Sell a stock position in a particular company.
- ☐ View chart on historical data.
- ☐ View income statement.
- ☐ View balance sheet.
- ☐ View profile on company.
- ☐ View research on company.
- ☐ View company SEC filings.
- ☐ Additional research links.

Simple Transaction Interface

This is a typical market order interface (missing the button to execute the order):

Transaction:

☐

Buy

☐

Sell

Stock Ticker:

Number of Shares:

Rather than radio buttons, this interface should use Buy and Sell buttons to expedite the transaction.

Future features might offer transaction options such as Stop Order, Limit Order, Stop Limit Order, Good-Til-Cancelled, and Fill-or-Kill.

Scenarios

(under development)

- Player Profession: Dabbler
- Player Profession: Trader
- Player Profession: Broker
- Player Profession: Underwriter

Leader Boards

(under development)

Other

Dividends

The Commodities Exchange

Overview

Mining resources and trading goods from planet to planet is at the heart of Privateer. Specific planets will have commodities markets, and players will be able to become successful dealers in these marketplaces, thus finding a whole new economic subgame.

Player goods can be placed on the Commodities Exchange for sale at a later date, even while players are offline, with their bank account being credited by the appropriate amount.

The purpose of the Commodities Exchange is to buy or sell Commodities.

Commodities are all game objects that the player can acquire through mining, harvesting, trading, crafting, exploration and discovery, or (possibly) stealing. Commodities include anything from natural resources and personal equipment to capitol ships and orbiting planetary defense systems.

Commodities are bought and sold directly through the **Commodities Exchange**; a galactic-wide database containing all posted Commodities, available through the player's PDA interface.

Vision and goals

Goals

- ☐ Provide a simple but full-featured system for players to buy and sell goods that they have crafted or acquired.
- ☐ Provide a way to sell both tangible and intangible goods.

Vision Doc

- ☐ ... will be easily extensible by the Live team
- ☐ ... must support as low end a machine as possible, to maximize the audience
- ☐ ... should have a world that feels alive
- ☐ ... will offer motives and rewards for short session play (<30 minutes)
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... should blur real life and the online character
- ☐ ... must be playable via the Web in some fashion
- ☐ ... must encourage and support a community
- ☐ ... must give tools to players to shape their own experiences
- ☐ ... should encourage a sense of player ownership in the game world

Walkthrough

The Commodities Exchange interface becomes available through the player's PDA whenever the player is within transmission range of a Commodities Exchange Server.

The items on the Exchange can be filtered to suit the needs of the player browsing the database. All Commodities that can be offered on the Commodities Exchange can also be traded between individuals through the Secure Trading Interface (see corresponding document.) Large items that cannot be carried in the character inventory are traded as digital deeds.

One distinct internal attribute of all items is whether they are Tangible or Intangible. Tangible items include such things as Equipment and Resources. Intangible items are those items that can be stored digitally in the player PDA, including blueprints, books, and even deeds for structures and ships. Also, "newbie" items cannot be sold on the Commodities Exchange.

Players with access to a Commodities Exchange may **purchase** an item from any system in the galaxy. Credits are transferred and the purchased items become their property. Transfer of tangible item(s) becomes the responsibility of the buyer. Intangible items are transmitted instantly to the player's PDA.

Tangible items placed up for **sale** on the Exchange can only be sold from the location where they are physically stored. To facilitate the most simple and open-ended transfer of goods, a sale is considered complete only when fulfilled at the Exchange (planet, station, cap-ship) where the original request for goods was posted. These are managed on a first-come, first-served basis. There is a chance that the player will arrive at an Exchange to find that another player has already fulfilled the order for the goods he carries. This eliminates the complexity of maintaining pending sales as well as "exploits" that could ruin the enjoyment of other players.

Example:

Mindy has acquired several thousand cargo units (CU) of high-grade Raphium from nebulae near her home system. Raphium is plentiful in her part of the galaxy, but not so in other systems. Earlier today, Mindy was browsing the Commodities Exchange for Wanted items and found a buyer on O'Reilly III looking for Raphium in significant quantities. The buyer is offering 500CR for each CU of the raw resource. Even after accounting for her direct costs of ship fuel and system maintenance, she can make a killing in this deal. The buyer is located in WildSpace and Mindy has no plans to leave TameSpace -- what few weapons she has are barely enough to hold off the Garwogs and Bugaboos that attack her outpost. Her only option is to hire someone to deliver the cargo for her. Fortunately for Mindy, she has a friend, a trustworthy cargo-jockey named Xerxes, who would be perfect for the job. Xerxes' fees will eat into her profits, but Mindy will still make enough to buy that Quantum Spacifier she's had her eye on. She quickly sends FTL-Mail to the buyer about the order and puts out the word to Xerxes...

A Note on Weights and Measures

Items sold on the Commodities Exchange are sold in quantities of individual Units or Cargo Units. All cargo bay capacities are rated in Cargo Units (CU).

Exchange Locations

A star system must contain a ConFed or Syndicate jump beacon to support the flow of Commodities Exchange data. A planet must have an active Commodities Market structure for a planetary Commodities Exchange to be available. The Commodities Market houses the Commodities Exchange Server and Cargo Bay capable of holding xxx CU. Purchasing Warehouse structures and linking them to the Commodities Market will increase CU capacity.

Fees

A fee may be required to list an item in the Commodities Exchange. The fee is configurable by the controlling Governor. Listing fees will vary in permanent SafeSpace (newbie) zones.

Perishables

There is an opportunity to assign decay or life expectancy values to objects stored in a Warehouse. This could apply to animals, vegetables, pharmaceuticals, and other perishables.

Commodities Categories

Within the Commodities Exchange database, items can be filtered to include only the following **subcategories**.

- **Equipment**
 - Armor
 - Clothing
 - Medical
 - Tools
 - Weapons
- **Intellectual Property**
 - Schematics
 - System Data (Maps, Planetary Data, etc.)
 - Books
 - Music
- **Pharmaceuticals**
- **Resources**
 - Contraband (*see related section*)
 - Natural Resources
 - Animals
 - Gases
 - Liquids
 - Minerals
 - Vegetables
 - Foods
 - Will include resources from the Animals, Vegetables categories.
 - Fuels
 - Metals
- **Structures**
 - Agricultural
 - Communications
 - Entertainment
 - Housing
 - Medical
 - Mining
 - Repair
 - Research
 - Retail
 - Weapons
- **Vehicles**
 - Armor
 - Cargo
 - Engines
 - Equipment
 - Scanners
 - Shields
 - Weapons

- **Contraband**

- Items in this category may only be purchased from Commodities Exchanges linked to the Syndicate.

Item Summary Interface

Each of the **boldfaced** item attributes listed below are displayed in a column within the item summary (row). Selecting the item summary will bring up more detail on the item.

- Status
 - Auction
 - For Sale
 - Wanted
- Item
- Price
- Author
- Location
- Other: **Auction**
 - Bids
- Other: **all, if applicable**
 - Expiration

Interfaces will be offered both on the player PDA as well as the web site. An example of the item listing is shown on the following page. Underlined items can be clicked on to reveal information about the Status, Item, Author, or Location. A method to provide a web link in the "Other" section may also be provided.

Item Summary Interface

Raykor Commodities Exchange

Location: Raykor
Time: 0000.00:00.00
Filters:

Status (all)	Item	Price Bid	Author	Location	Other
	Qty:			Star Maps	

StarFinder

Listings (7 of 2319):

Status	Item	Price Bid	Author	Location	Other
Auction [R]	Qty: 1000 cu Quadrotriticale	Min.Bid: 2200 ^{CR} Reserve: <met>	Reaver	Agraria	Bids: 20 Tribbles not included!
Auction [D]	Qty: 1 of 100 Ritalin's Super Battery	Min.Bid: 180 ^{CR}	Ritalin	Rygel	Bids: --
Auction [R]	Qty: 1 of 1000 cu Purified Raphium	Min.Bid: 500 ^{CR}	Mindy	Nebulon	Bids: 3
For Sale	Qty: 1 Capitol Ship, "TCS Compton"	2,000,000,000 ^{CR}	Dweezil	Earth (Sol)	One owner. Great condition. Los Angeles Class, PT-40 TCS Compton, Unequipped!
For Sale	Qty: 1 of 2 Quantum Spacifier Bonus: +10 Jump	218,000 cr	Caliburn	Rygar	
For Sale	Qty: 1 of 23 "For Sale" Signs	10 ^{CR}	Longshot	New Austin	
Wanted	Qty: 1000 cu Purified Raphium	500,000 cr	Gardada	O'Reilly III	
Wanted	Qty: 1 F'nargin Blaster, TL8 Add: Scope, TL8 Bonus: +8 Attack Bonus: +10 Aim	340 ^{CR}	Xerxes	Brandywine	Must be in new condition!!

Item Detail Interface

TBD

Auction Types To Emulate

Reserve Price Auctions

- A reserve price is set and must be met before the item will sell.
- Bidders do not know what the reserve price is, only that the reserve price must be met.
- A bidder must meet or beat the reserve price and be the highest bidder to win the auction.

Dutch Auctions

- The seller must list the quantity of items for sale and the starting bid.
- Bidders must specify the quantity and price they wish to purchase.
- All winning bidders pay the same price. The winning bid is the lowest bid that meets or exceeds the starting bid price.
- The earliest bidders get the goods if there are more items than winning bidders.
- The highest bidders receive the quantities that they bid on.
- Partial quantities can be refused by winning bidders.

To Be Addressed

- ☐ How to specify exact attributes of items listed in order to auto-fulfill orders.
- ☐ Global vs. Local Commodities Exchange
- ☐ ConFed vs. Syndicate Commodities Exchange
- ☐ Optional Trading skill / level as a prerequisite to using any Exchange.
- ☐ Corporate vs. Individual Commodities Trading
- ☐ Is warehouse space pre-allocated with each buy or sell order?
- ☐ Potential exploitation of Exchange warehouse limits via listing buy or sell requests with impossibly high prices.
- ☐ Ban lists for each individual Exchange.
- ☐ What happens to perishable items when they 'decay'?
- ☐ Advertisement banners as a Money Sink
- ☐ How to evaluate the quality of listed items.
- ☐ Listing fee management.
- ☐ Web trading.
- ☐ More filtering detail.
- ☐ Maximum credits able to be represented.
- ☐ Time system.
- ☐ More interface.

Companies

Overview

The document covers the basics of guild/corporation/business formation only. Guild benefits, abilities, controls, vendors, etc. are to be covered in separate documents. Some of the Guild Formation actions have been gleaned from UO and UO2 design.

Vision and goals

Goals

- ☐ Make Guild formation simple and relatively automated.
- ☐ Offer customizable management and maintenance parameters to make guild control reflect a player's needs.
- ☐ Make guilds the desirable method by which people gather together in groups to pool resources, items, and experience.
- ☐ Create reward based money sinks to keep the economy stable.
- ☐ Encourage veteran guilds to recruit new players.
- ☐ Create organized rivalries outside the "roaming gangs" principle of traditional online experiences.
- ☐ Create a "Good" guild system and an "Evil" guild system that compliments the two playing styles with equivalent role-play rewards.

Vision Points

- ☐ ... should have an intuitive user interface
- ☐ ... should empower players to take action against those who are obnoxious
- ☐ ... must minimize human administrative intervention
- ☐ ... must be playable via the Web in some fashion
- ☐ ... must permit means of "playing" while offline
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should blur real life and the online character
- ☐ ... will offer tools for player-driven content
- ☐ ... will provide privileges to experienced players
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... should encourage specialization
- ☐ ... must support many different styles of play
- ☐ ... should have little touches that add depth
- ☐ ... will offer players lots of rewards
- ☐ ... *must encourage and support a community*
- ☐ ... *must give tools to players to shape their own experiences*
- ☐ ... *should encourage a sense of player ownership in the game world*
- ☐ ... *should have ladders and leader boards for everything in the game, including social activities*
- ☐ ... *will support a wide range of in-game communications systems*
- ☐ ... will offer robust support for players to form guilds and organizations

Walkthrough

In Privateer Online, the term guild is for design/production clarity. In game, players will form partnerships that evolve into businesses, which evolve to corporations. Two players will be able to form a partnership, more a business, and more still a

corporation. This hierarchy is skewed in the case of an evil guild...or crew, family, and syndicate.

This creation/formation process is simple and mostly automated, with a deduction of scaling fees based on the number and level of members. During creation the players will be able to modify the *name, mission statement, purpose options, colors/symbol, org. chart/hierarchy*.

With some of these configurable items come an extensive series of parameters to be described individually below.

Game Example: The newbie company with starry eyes

Jimmy's character Irin has been playing PO for a while. He's got an okay ship and an okay friend base...but he's growing tired of being the loner. He's seen the instant corporate comm. Channel access other folks have...the shared datanode info streams...the corporate bank accounts, the company ships/weapons/items...the discounts on corporate merchandise...the fast money made from stock. All of these things have made him hungry to be part of a winning team.

He organizes a meeting between him and four of his friends (Stimpy, Altercom, Julius, and Seether). He proposes that they form a company called Intertech Sevices. Their combined experience, data node access, and reputations position them nicely to benefit from the perks a business gains. Instant radar, shared assets, access to multiple ground facility info...All of these things, plus a unique stock presence makes the group drool.

Stimpy has a unique Tech 5 Shield Generator with 50% higher energy efficiency than a comparably crafted item. He's just managed to set up a facility on a mineral rich planet with a well-stocked warehouse of resources that can automatically manufacture these items at half the rate a human can craft by hand. Jimmy instantly sees the Intertech brand on this bad boy as INSV makes its debut on the stock market.

Jimmy's not just a privateer anymore...he's a player.

Game Example: The Vicious Raiders of Mollerhaug.

Horshack, Bile, Whiplash, and FunnyMonkey have a group of about ten others they go hunting with. They stick to wild systems and pick off inexperienced traders and over confident heroes. They've begun running contraband and have their eyes on a wild space asteroid field for a possible base.

*They decide **they** need access to the gold of organized benefits. But they can't form an official company! They're outlaws! However, they do know of the Syndicate (or the firm, the triad, the mob, the family, the outfit, whatever folks call them in their sector).*

Horshack had just reached a reputation level that a "rep" from the Syndicate had contacted him. He could get "made" and form his own crew!

Bile recommends they think carefully. He just made contact with a crew looking for members and he's pretty confident that they could take the operation over and reap the benefits of an established group.

Ahh...the politics of evil...

Systems design

Good Vs. Evil

The first step in any player's thinking is what their guild/company is about:

Do we want to kill, exploit, destroy, and/or smuggle? OR Do we want to protect, train, create, and/or trade?

The former has a few more issues to deal with than the latter.

In the latter's case registering a guild is a simple affair of pulling up your PDA and visiting the official CorpRegistry and filling in the forms that follow (See Guild Formation). It only takes two people to form a guild, but dues are generated that must be kept up with or the guild will default and be deleted. This as well as other payment models (number of members, level of members, etc) will be covered in more detail in **Guilds: Structure.**

True criminals will wish to form their own order without having to worry about the investigative eye of Confed or CorpRegistry. They can form covert alliances or arrangements with legitimate corporations, just like real life, but answer to a different power: The Syndicate (placeholder name!!!).

A player can only form his own evil guild once he/she has achieved a reputation level that attracts the "attention" of the Syndicate. This attention is in the form of messages that state, for example, "We have seen your progress and would like to present you with an attractive option..."

At this time the player can form an evil guild (be "made" by the Mob) that shares all the same in-game benefits as a legitimate corporation...just with inverse reputation requirements where necessary. A player can only join an illegal guild before the reputation requirement if they are "made" by another player.

A band of pirates/criminals that wishes to organize under the guise of a legitimate corporation is subject to far more account/reputation scrutiny by the powers that be. This is accomplished by a series of reputation checks to determine where guild revenue is being generated; guild items are procured from, etc, etc. Each violation of interstellar law results in fines, both small and massive that will make it next to impossible for a guild engaged in overt illegal activities to stay in the loop.

Also, AI Pirates will still target and destroy criminals who hide behind the guise of legitimacy without the blessing of the Syndicate.

These issues of legality/legitimacy will also be covered in more detail in **Guilds: Structure.**

*{Note: The practical data structure and maintenance of an official Corporation vs. an illegal Crime Ring is identical. It is merely a shell to differentiate the Heroes from the Villains. It allows people to role-play more comfortably within the environment they wish to explore instead of all guilds (good and evil) being registered with one agency regardless of intent. The differences lie solely in immediacy of **availability**, contextual **framing** and inverse **reputation** checks. This will be covered in a separate document.}*

Guild Formation

When a player decides to form a guild they access their PDA and pull up the official CorpRegistry, or the secret socket into Syndicate.Net (Note: all these cheese ball names are placeholders). A player can be a part of three guilds at any given time, but can only lead one at any given time. Did I mention UO2 design was more than helpful here?

They then have to complete the following:

- **Name:** Name of Guild (see Names and Mission Statement)
- **Mission Statement:** Description of Purpose/Credo
- **Purpose Options:** Purpose of the guild established for quests, commodities, goods, and other data (see Purpose Options)
- **Colors/Symbol:** Choose guild colors and construct guild symbol from templates (see Colors/Symbol)
- **Hierarchy:** Player establishes Hierarchy for Guild (see Hierarchy)
- **Election:** Player sets initial Leader election rules (see Election)

Names and Mission Statement

When a player enters in the guild name, it is of course compared against our existing obscenity filter, and then checks against the filter using a system that looks for each alphabetic character in sequence - ignoring spaces and non-character symbols.

The Name is then compared to the current list of Guild Names. If it matches an existing one it is not accepted. If it is accepted, then it is registered and no one else can take that guild name (on that server).

When a player enters his credo/mission statement it is similarly filtered, checked and registered.

{EGG: An optional "Translate to Latin" option is provided allowing a reasonable Latin Credo for the foundling company. This utilizes any of a hundred online translators. }

Purpose Options

These differ for the formation of a good or evil guild. Appropriate options for either are provided with some overlap in the case of Trading, Protection, etc.

The player will use checkboxes to select options like:

- We are player killers
- We are protectors
- We are traders
- We are crafters

Our Mission Generation engine will then use these options to filter tasks for their Guild Mission Board that is customized for their purposes. (Did I mention UO2 rocks?) We can also use this information for other TBD purposes.

Color/Symbol Options

After establishing the Purpose options they will be able to configure the major and minor colors for their guild.

They will also be able to create a Guild Symbol. This is a simple paper doll interface. First the player selects the symbol template (circle, square, triangle). This template is broken into an invisible grid that symbols can be dragged and snapped onto. These symbols are representative of both in-game skills (Engineers, Pilots, Trade, etc.) as well

as a colorful array of extras. The players can then drag and position these icons in a snap-o-matic fashion inside the symbol template.

{EGG2: Allow players access to the dimensions of the iconography and allow them to create their own symbols for insertion in the game ALA Half Life's spray paint option.}

The Major and Minor colors are applied to this symbol and then it is checked against the existing guild symbol database (after converting it to its unique number). If it passes the existing checks then that symbol is also registered and unique to the guild.

This is a decal that can be applied to guild ships, structures, uniforms, accessories, email watermarks, etc.

Hierarchy

Similar to UO2, a guild's hierarchy has the default ranks of Leader and Member after initial construction. This can then be fully modified, added to, etc. The guild's structure (President, Vice-President, etc.) after the ranks, rules and powers are associated generates an Org Chart. Guild Members using their PDA can reference that to check up on who is who within the corporate/crew structure.

Election

The player must establish the initial ground rules for how leaders are elected and maintained. Options for this will of course vary dependent on whether or not the organization is of a criminal or legal nature.

Some of the Election methods include:

- Support
- Majority Support
- King of the Hill
- Wealthiest Member
- Member with Highest/Most Notorious reputation
- Member with most stock in company

The last entry leads me to mention that Elections and other Guild activities will be fully outlined in the **Guilds: Structure** document. We will enable players to live out shareholder dreams. After playing Privateer Online some young entrepreneurs will have a fun introduction to corporate politics...and cutthroat attrition.

Interface design

TBD

Tasks

To be addressed

Companies - Tools

Overview

As stated in *Companies - Formation*, guilds are actually partnerships and businesses run by groups of players. In order to make guilds an attractive option we will offer a series of tools, abilities, and options that make them feel like real game play elements instead of glorified friend lists. Covered here are an initial sampling of the tools we will provide guilds, and how guilds can declare war against each other.

Vision and goals

Goals

- ☐ Provide players with a rich toolset to make their guild as specialized as possible
- ☐ Make the guild experience a deeper and richer one than has been offered in the past
- ☐ Encourage player guilds by offering attractive perks.
- ☐ Make War more of a Game instead of just bloodshed.

Vision Points

- ☐ ... must have an intuitive user interface
- ☐ ... must be easy to get into, as a new player and as an advanced one
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... must support character building and advancement
- ☐ ... will offer tools for player-driven content
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... should blur real life and the online character
- ☐ ... will provide privileges to experienced players
- ☐ ... must support many different styles of play
- ☐ ... must have little touches that add depth
- ☐ ... must be playable via the Web in some fashion
- ☐ ... must encourage and support a community
- ☐ ... must give tools to players to shape their own experiences
- ☐ ... should encourage a sense of player ownership in the game world
- ☐ ... will offer robust support for players to form guilds and organizations.

Walkthrough

Covered in this document:

- Rank/Hierarchy
- Guild Perks, Resources and Sharing
- Affiliation Broadcasting
- Guild Chat
- Guild Info Page
- Guild Calendar
- Guild Mail
- Guild Vendors
- Declaring War: The Rules Of Engagement

DDR NOTE: This is the first overview of the abilities we wish to empower guilds with. This is going to be one of the most revisited docs as far as game community is concerned. At the moment these are brief top level descriptions of the functionality we initially would like to see.

Systems design

Rank/Hierarchy

As briefly described in **Guilds - Formation** the guild leader will be able to reorganize the power structure of their guild at any time. This is to be handled org chart style with the leader able to set the level (position in ladder) and associated title (i.e. Lvl1, CEO), name, number of votes, sharing level, ability to recruit, etc.

The player will be presented with a wizard that requires the following info:

- Level – The level of the position on the chart (i.e. CEO is 1)
- Rank/Title – The title of the position
- Answers To: What player or title does this position answer to (i.e. Designer answers to Lead designer, etc.)
- Name – The player name in this rank.
- Additional Properties – These are the optional additional parameters the player can set. Can be set to none.

The exact limits and types of guild member powers/properties are TBD...

Gameplay Example:

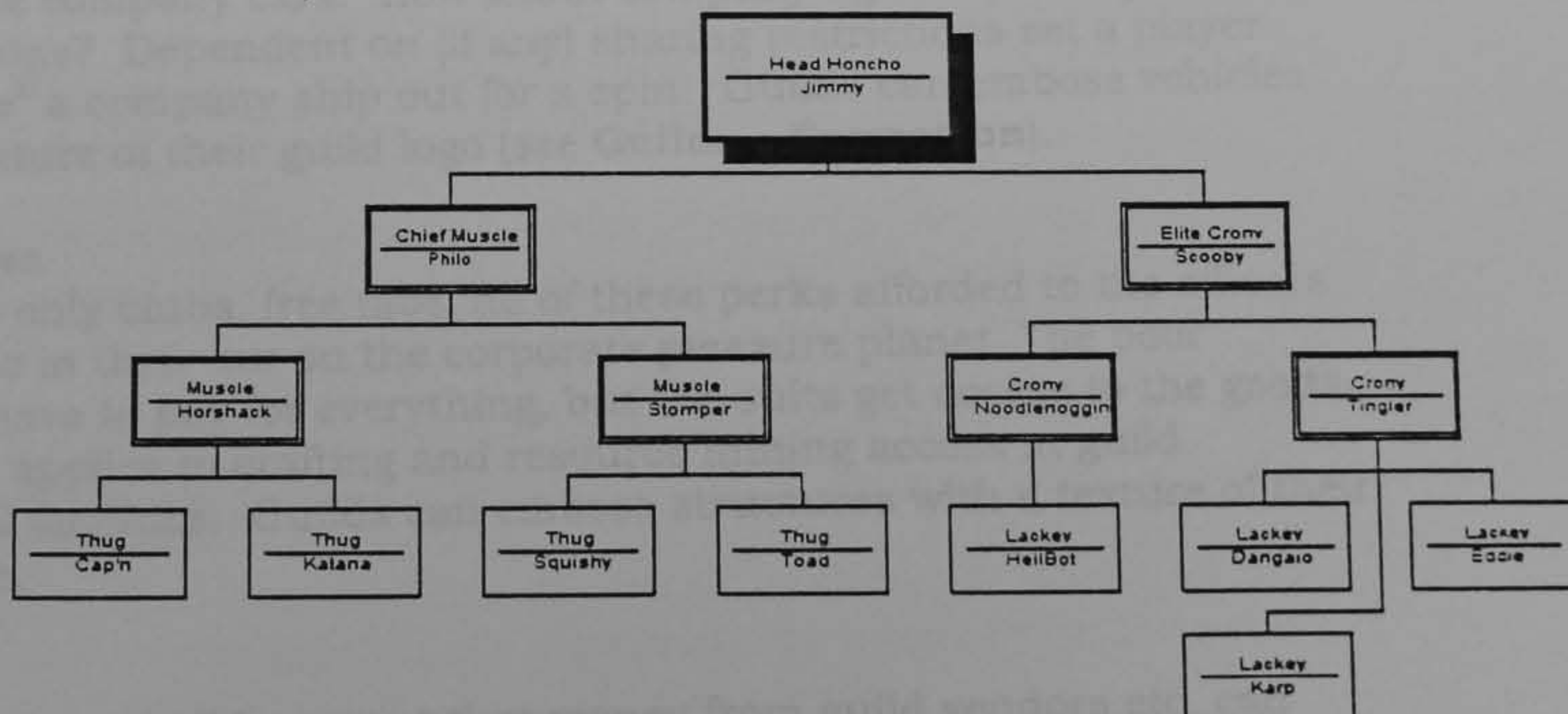
So Jimmy decides to start a guild with his friends.

He fills out the first level, which has the default fields of Title/Rank (Leader) & Player Name (Jimmy). He adds to this the optional properties of Sharing Level (a sharing access level can optionally be assigned to all guild owned items, objects, vehicles, etc.), number of votes, "Can Recruit?" "Answers To".

He then sets the limit to total number of level 1 members. This defaults to 1.

He enters in his information for the options and then submits it. He then sets up

two level two slots (Elite Crony and Head Muscle) and so on and so forth until he reaches the bottom level which he sets to unlimited. If his guild ever gets too big to handle he can either revoke recruiting access or set a limit to the bottom level to prevent new recruits from coming in.



When he completes it an org chart is generated that he can scroll through and/or search. He can individually tweak properties on the Org Chart which is

then reflected in the ascribed powers of the individual..."So, Colin, you've done good work for us, but we don't have an available higher position...here, let me give you two more votes and increase your sharing access. I bet you'd like to try out the company's new light fighter, wouldn't you?"

Guild Resources and Sharing Access Levels

There are many perks for joining a guild.

Resources

Data Node Access

One of the biggest perks of being in a guild is shared information. As guild members hard link to data nodes, the guild gains access to the specific system data these nodes provide.

So a player can actually access the current status and scope of a system via its datanode uplink without having to visit the system to manually link to the data node located there.

Reputation Standing

Dependent on the guild's reputation, aspects of the individual's reputation may be summarily raised or lowered by the standing of the guild. This may not sound like a perk, but think of it this way: "Tommy was a nobody working a McDonalds, and then he was hired on as Chief of Security at Lockheed! Wow! Now Tommy gets recognized as a VIP places!"

Items

Not everyone can afford, craft, or even find certain items in the game. A guild provides its members with an inventory of guild items they can use. Any guild assigned item being used by a player can be tracked by the guild itself. "Hey, where's the powered armor gone?" "Lemme check...John has two of them and Scooby has the other..."

Vehicles

We all love company cars. How about company fighters, transports, or capital ships? Dependent on (if any) sharing restrictions set a player may "take" a company ship out for a spin. Guilds can emboss vehicles with a texture of their guild logo (see **Guilds - Formation**).

Structures

Members only clubs, free tabs, all of these perks afforded to the officers of AmalCo in their bar on the corporate pleasure planet. The poor tourists have to pay for everything, but the suits get access to the goods. This also applies to crafting and resource mining access at guild industrial facilities. Guilds can emboss structures with a texture of their guild logo.

Money

Guilds setup a guild account that money from guild vendors etc. can reside. Members can be given withdrawal rights up to a credit limit by the acting treasurer (default Guild Leader).

Guilds generate revenue, a percentage therein can be distributed to the members after all guild dues/fees are paid.

Guild Leaders can also establish member dues that players have to pay to remain in the guild. This is TBD in conjunction with overall guild dues and payments.

Sharing Access Levels

When guilds start the notions of ownership as covered in the Ownership document are all that is necessary. However, as a guild grows larger and larger, leaders may wish to place limits on the access to guild structures, items, equipment and case that members have.

Anything that can be freely used by guild members can having a sharing access level ascribed to it (vehicles, items). A quick check between the member's current access level and the object in question indicates that player's ability to utilize it.

Sharing Access is not just for physical objects; it also has to do with information. Setting up what members can access detailed financial information etc. is illustrated more in the Guild Info page section of this doc.

Affiliation Broadcasting

All characters will have the option of broadcasting their guild affiliation in their character profile. This allows for secrets and spies. ☺

Guild Chat

On the player's PDA interface they will gain a guild chat button with the color/symbol of the guild. This button will enable a guild chat channel that is basically a dedicated secure communications hook up that enables constant contact between members.

We are discussing the possibility of an enemy guild member "hacking" into a guild chat channel.

Guild Info Page

As well as guild chat, the guild leader (or assigned member(s) will be able to configure an Info page. This may contain new recruit notices, link to member listing (with profile links), current financials, status of war, available inventory, ships, etc.

Essentially this will be where a player goes to find out everything they need to know about a guild's inner workings. The guild leader can decide what information is to be displayed and the level of guild member sharing access can be assigned to information.

So Jimmy (CEO) can keep Squishy (new recruit) from being able to parse the company's financials and inventory, but allows him access to general news, etc.

It will also contain an optional section that can be written by the guild leader, or whoever has edit rights to the guild info pages. This might be a feature on future tactics, an upcoming battle, etc.

It will also contain a day minder that links to the Guild Calendar.

This is to be designed further.

Much, much further. ☺

Guild Calendar

This is a calendar that the guild leader, or those assigned access, may enter information. The current day of the calendar is displayed on the main Guild Info Page.

This is an I-CAL style calendar with a summary month-at-a-glance mode with links to daily items.

Sharing Access can be applied to Calendar items as well. So the grunt won't see the executive meeting schedule, etc.

Guild Mail

The guild leader or member with administrative access can send mail out to all the guild members.

This could have a real world component as well.

Guild Vendors

For small guilds, extreme control over inventory won't be an option. But the tools we build now are envisioning guilds that could grow to a hundred members or more! Everyone needs to have rules for their company store at that size.

A guild can configure its vendors to give variable discount rates on products to members and/or free access to specific inventory items (individual or type) they carry.

Declaring War: The Rules of Engagement

Most games allow guilds to declare war against each other...but why? There never seems to be point other than getting to fight each other.

Warring guilds can fight each other in tame space as bounty hunters and pirates do.

We wish to formalize the process of guild warfare to support different game play styles. We also wish to provide objectives for Guild Wars aside from basic brutality.

Putting in hooks to detect guild warfare and post notices of conflict on News.Net are rich concepts worth exploring.

The key steps to declaring and starting a guild war are as follows:

Send a Declaration of War

The first step in starting a guild war is sending a formal declaration to the enemy guild. This declaration lists the requirements that must be met in order for the war to end. There must always be at least one requirement.

Potential examples include:

- War ends when one side {surrenders}.
- War ends when one side {declares peace}
- War ends when one side {loses 50 ships}.
- War ends when one side {is the first to generate x credits from sales within (sector/quadrant)}
- War ends when one side {surrenders} OR {loses 50 ships}.

Depending on how creative we are with this is means that guilds can declare either Financial Warfare, or Physical Warfare. The potential requirements are simply rules we create that encourage guild competition while framing a distinct and recognizable end to the conflict.

The leader who crafts the declaration can also specify any rewards at the end of the battle (first person to reach this has to ____).

Garrett (Rich)

After crafting the declaration, the guild leader sends it to the enemy guild leader.

Accept the Declaration

The enemy guild leader must accept the declaration for the war to be on. The rules of the declaration are then put into action. Once the rules for the war ending have been met, the war ends.

Interface design

TBD.

Tasks

TBD.

To be addressed

TBD.

Walkthrough

Automated installations can only be done in a specific area of the game world. Some installations may not be appropriate for a specific environment, such as a window installation in space.

There are six categories of automated installations: Harvesting, Processing, Manufacturing, Storage, Vending, and Defense. Each of these categories is used to define what a specific type of installation functions as.

Since bandwidth and processing time are major concerns, these installations will function with self-contained databases and will not need to be updated. Otherwise, they are designed and used only as database tools.

To help facilitate in the development of automated installations can be linked to automate the transfer of materials from one type of installation to another and eventually link multiple installations to create a fully functional production/vending circuit.

For example, Jimmy decides to build an ore processing plant on the planet Jovis. Jimmy uses his resources to build an ideal spot for placement and, using the standard building placement interface, places the ore processing plant. Jimmy then decides to place a vending machine next to the ore processing plant.

Automated Installations

Overview

Automated installations are meant to cover all types of facilities that are not considered housing functions. Purposes such as {resource gathering, resource processing, storage, resource trade that does not fall under the *Commodities* market, and defenses} fall under this domain.

Vision and goals

GOALS

- ☐ Remove the tedium of resource harvesting. (ala Ultima Online)
- ☐ Remove the tedium of the manufacturing process for mass production.
- ☐ Offer automated defenses to protect celestial and terrestrial assets.
- ☐ Provide an easy-to-use interface for linking installations to reduce tedium.
- ☐ Reduce bandwidth and processor costs by having installations be only processed when accessed, and not a constant process.

VISION

- ☐ Have an intuitive user interface
- ☐ Should keep a small database size
- ☐ Should have a low target bandwidth
- ☐ Expose new strategies throughout the game to the player
- ☐ Offer motives and rewards for short session play
- ☐ Never make the player feel ripped off by the game mechanics
- ☐ Encourage a sense of player ownership in the game world

Walkthrough

Automated installations can exist in space (celestial) or ground (terrestrial). Some installations may not be appropriate for a specific environment, such as a vending installation in space.

There are six categories of automated installations: Harvesting, Processing, Manufacturing, Storage, Vending, and Defenses. Each of these categories is used to define what a specific type of installation functions as.

Since bandwidth and processing time are major concerns, these installations will function with self-contained simulations *only* when accessed. Otherwise, they are dormant and exist only as database fields.

To help facilitate in the *ease-of-use*, automated installations can be linked to automate the transfer of materials from one type of installation to another and eventually link multiple installations to create a full production/vending circuit.

For example, Jimmy decides to build an iron-ore extractor on the planet Xerts. Jimmy uses his resource scanner to find an ideal spot for placement; and, using the standard building-placement interface, places the iron-ore extractor. Jimmy then leaves the planet to let the iron-ore extractor do its thing.

Several days later, Jimmy uses his PDA to check on the iron-ore extractor. He is notified that it might take a moment to get a communications link with the installation (our fictional way of justifying non-instantaneous pauses for database activity). [While he is waiting, the simulation runs in the background that uses the current time/date and the last-accessed stamp to compute what has happened in the time elapsed.] Jimmy then receives an in-game report detailing how much ore has been produced and any special events that may have occurred.

Systems design

Types of installations

There are four types of installation categories:

Harvesting: Harvesting installations are resource gatherers. Usually their efficiency is affected by where they were placed. Harvesting facilities have a small amount of storage, but should be linked to Storage facilities for better effectiveness.

Processing: Processing installations convert raw resources into refined resources. They can be linked directly to harvesting installations to provide a constant flow of refined resources. Depending on the materials used during the construction of a processing installation, they can be made modular. This modularity allows them to change out engineering and software packages to be able to process multiple raw resources. (See *Crafting* for further discussion on engineering and software packages.)

Manufacturing: Manufacturing installations are the large-scale crafting facilities. Once a *blueprint* has been uploaded into a factory and the necessary materials are present in its storage (or linked storage), it will begin to mass-produce the item detailed in the *blueprint*. (See *Crafting* for further discussion on *blueprints* and materials.)

Storage: Storage installations are used to store resources that have been harvested or deposited by an owner. Different types of storage facilities exist to handle different types of resources and also may have different storage amounts.

Vendor: Vendors are discussed in a separate document; see *Vendors* document.

Defense: Defenses are used to protect the *owner's* other properties from attack or theft. The defenses have a variety of different settings to allow the *owner* a range of options in how the defense installation functions. (These options "to-be-designed")

Time stamp system

One of the main goals of resource harvesting, manufacturing, and processing is to avoid the tedious nature of these that is present in the current crop of similar products. To this effect, all relevant installations will use a system that allows the player to do other things, or even be offline, while the installation continues on as normal. This system is called the Time Stamp system.

Whenever a player accesses an installation, it compares the current time and date with the *last-accessed* stamp. (Accessing includes doing a query on the installation *even* when the planet, the installation is on, is offline.) It uses this difference in current time and *last-accessed* to simulate what has happened in interim. Once the simulation is complete, it then stamps the current time and date as the *last-accessed* stamp *ObjVar*.

This simulation of time-elapsing is crucial to this system. In-between accesses, there is no background code taking up processor time or bandwidth. The simulation only runs during accesses, similar to power management on computers these days. When no one is accessing, it is not taking up any resources other than a database entry.

Once an access occurs, the simulation uses a set of algorithms to simulate how much resources are gathered or processed, how many items are manufactured, what random events have affected the installations, and any other pertinent data.

Accessing information

It will be important to detach the simulation algorithm from the planet server in some way. It will be very important to be able to run this simulation regardless of the status of the planet server. Since a database will be used to store all data, the installation database should be able to manipulate and change the data stored at any time.

*For example, a player should be able to log into the game and access the installation from anywhere. When the player accesses the information on the installation, it should process the simulation first, and then reflect the new data in the report given to the player. Also, it should update the installation's data in the database to reflect the simulation. Finally, it should update the last-accessed stamp on the installation. This should occur even if the planet server where the installation resides in is **offline**.*

Random Events

During a simulation of time elapsed between the time stamps, one of the modifiers is random events. These events are designed to add a sense of tension and fun to resource harvesting. A random event could be either a good or bad event. These are analogous to the *Community Chest* and *Chance* cards in the popular board game *Monopoly*.

Since random events can effect more than a single installation, these must be tracked and applied to all necessary groundside structures when they occur. Some sort of event database is "to-be-designed".

Linking different installations

The different types of installations can be linked together to automate tasks such as transferring from harvesting to storage or from storage to being listed on a nearby vendor. The requirements for linking are:

- The installations must be within x distance from each other
- Installation owners must have access rights to all installations in the set. (See *Ownership* document)
- All installations in the set must be functional and fully repaired
- A resource and money requirement must be paid to link them

Once a set of installations is linked, they can interact with each other without human intervention. For example, a harvesting installation can deposit resources directly into a storage facility. An option can be set that transfers excess resources from a storage facility to another storage facility or a vendor if desired. A simple rule set will need to be generated to handle a linked set of installations' behavior.

Accessing Linked Facilities

When a facility that is linked is accessed, it should activate all linked facilities and run the simulations for each. Once done, it should update all of the *last-accessed* stamps accordingly.

Interface design

The placement and basic functionality of automated installations is covered in the Housing document. The section in this document will specifically deal with automated installation concerns.

Linking Dialog

The linking interface is used to link nearby buildings, so that a player may automate the interaction between them. The Linking Dialog is not always needed. If an owner places two installations near each other, they can choose to automate the linking process. The following information is needed:

- Browse (Forward/Back) between the linked set of installations
Allows the player to jump to the previous or next installation in the set.
- Add an Installation to the linked set
Allows the player to add an installation, via a visual selector, that meets the linked criteria.
- Disconnect Installation from the linked set
Allows the player to remove an installation from the linked set.
- Fulfillment Priority
Depending on the installation type, this dialog presents one or two lists of installations. One represents a list of installations (in the linked set) that feed into this installation. The other represents a list of installations (in the set) that feed out of this installation. Some installations such as harvesting types will not have the "feed into" list.

Harvesting Interface

The following information is needed in addition to the standard interface defined in *Housing*:

- Power up/Power down toggle
- Current Production Rate (measured in amount/time period)
- Estimated Amount remaining to be harvested
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Processing Interface

The following information is needed in addition to the standard interface defined in *Housing*:

- Power up/Power down toggle
- Change out of Modular Processor (see above)
- Current Production Rate (measured in amount/time period)
- Amount left to be refined.
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Manufacturing Interface

The following information is needed in addition to the standard interface defined in *Housing*:

- Power up/Power down toggle
- Upload new *blueprint*.
- Delete current *blueprint*.
- Current Production Rate (measured in amount/time period)
- Estimated number of finished product that can be produced with current inventory of materials (including linked storage)
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Storage Interface

The following information is needed in addition to the standard interface defined in *Housing*:

- Add/Remove/Destroy from Storage
- Inventory report of stored Resources/Items
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Vendor Interface

This is covered in the separate *Vendor* document. Please refer to that for specifics on Vendors, but the following will be present at the very least:

- Add/Remove/Destroy from Storage
- Inventory report of stored Resources/Items
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Defensive

The following information is needed in addition to the standard interface defined in *Housing*:

- Power up/Power down toggle
- Ammo/Power status
- Security Options (These options "to-be-designed")
- Link Dialog (for Linking or examining linked Installations)
- Ownership Dialog (see *Ownership* document)

Tasks

The following tasks are needed to implement Automated Installations:

- Artwork needs to be generated for the different automated installations
- Artwork needs to be generated for the link interfaces
- Scripts need to be generated to handle the Time Stamp systems.
- Code needs to be generated to handle the simulations even when planet is offline.
- Random event system needs to be generated in script.

To be addressed

- List of all specific automated installations needs to be generated and categorized.
- How installations are deployed celestially.
- Define the security options for defenses.

Ownership

Overview

Players in *Privateer Online* will be able to designate access and *ownership* of all structures and ships that they possess. This concept of *ownership* is crucial in determining who is assessed the maintenance costs of any specific structure or ship. As well, *access* allows the *owner* to designate other players or groups who can use a structure or ship that they possess.

Vision and goals

GOALS

- ☐ Allow players or companies to retain a structure or ship as an asset. This designates them as responsible for any maintenance fees or taxes assessed to said structure or ship.
- ☐ Allow players or companies to designate who can use or operate structures and ships that they own.

VISION

- ☐ Encourage and support a community
- ☐ Encourage a sense of player *ownership* in the game world
- ☐ Offer robust support for players to form guilds and organizations

Walkthrough

Ownership

Ownership is used to track whom is responsible for any fees associated with a specific structure or ship. It is also listed as an asset for tracking purposes on the *owner*.

The *owner* can either be an individual or a company. If the *owner* is a company, someone inside the company is designated as the *maintainer*. If the *owner* is an individual, not a company, then they are the *maintainer* by default.

Access List

An access list is maintained on each property (be it structure, ship, or installation) to determine who is able to access and use that property. The owner of the property maintains the access list, and the *owner* is the only individual capable of modifying the access list of a property that they own.

For example, Jimmy just bought an iron-ore extractor and places it planet-side. His buddy, Philo, is a budding surveyor and asks Jimmy for access to the iron-ore extractor. Since Jimmy is the owner, only he can grant access to Philo. Once Philo has access, he can keep tabs via the installation reports, power it on or off, and take the extracted ore elsewhere to sell. Philo cannot destroy or sell the building, nor can he give someone else access.

Later, Jimmy decides that the iron-ore extractor is too much of a drain on his time, so he sells it to Philo for 1000 credits. Once the transaction occurs, Jimmy is wiped from the access list, and Philo is designated as the new owner. Philo could add Jimmy to the access list if he wanted to. Philo also recently built a storage facility nearby, but since he didn't own the extractor till now, he could not link them. Now he links them so he doesn't have to manually transfer the ore to the storage facility.

Systems design

Ownership

Ownership is established, initially, by who purchased/manufactured a structure or ship. *Ownership* of a structure or ship brings with it several responsibilities. All costs associated with routine upkeep, taxes, fines, and etc are assessed from the *owner's* bank account. The *owner* is the only one who is able sell/dismantle structures or ships; and, the only one who can transfer *ownership* to another.

Designating Ownership

Ownership can be designated via a simple dialog that is present in housing, installation, and ship menus. Once *ownership* is transferred, the current *owner* is removed from any association to the *owned* object, including the access list. The new *owner* has the choice to wipe the current access list. If this is an installation in a linked set, it will be removed from the set unless the *owner* is present on the other installations' access lists. The new *owner* must add the previous *owner* to the Access list if that is desired.

Companies as Owners

A company can be declared as the *owner* of a structure or ship just like an individual player. The only difference is that an individual player still must be designated as the acting *owner*. This is automatically designated as the CEO of the company. Possible other options could include a designated title or position within a company that can be given this as an option. (See *Guild/Company Tools*) Regardless of the individual who is *acting* as the *owner*, all fees are assessed from the company's account.

Access

Access is used to determine who can or cannot operate a specific structure or ship. Access implies use but not *ownership*. Someone who is on the access list on a structure can freely enter the structure; use any of the facilities present in the structure, and add/remove items from the inventory of the structure.

Someone who is on the access list on an installation can check the status of an installation, power it on or off, and add or remove from storage.

Someone who is on the access list on an installation may link this installation to one that they *own* or *maintain*.

Someone who is on the access list of a ship can operate and reload the ship, but *cannot* sell the ship or sell/refit add-ons. If a ship is destroyed, the insurance will replace the ship as per normal to the current player. (See *Insurance* document for the way we prevent abuses of insurance)

Access Lists

The way an *owner* designates access is through an access list. The *owner* uses a simple, intuitive interface to add players or companies to the list. Once added, these players follow the rules of access established above. Selecting a player or company as a new addition should use a standard name search tool, called name-picker controls in this document. This search tool is explained in more detail in a different document.

Companies in Access Lists


Companies can be designated in access lists just like individuals. This is simply a wildcard to state that anyone belonging to the company is on the access list.

Interface design

Ownership Dialog

The *ownership* dialog is a small, simple interface that allows the current *owner* to designate a new *owner*. In the case of sale of a structure or ship, the designation can be tied to an exchange of items, resources, or credits. Once *ownership* has been transferred, the previous *owner* is removed from the access list.

Necessary fields include a field (with drop-down list controls) for selecting the structure or ship to transfer *ownership* for, the location of said object, a field to show the prospective *owner* (with name-picker controls), a toggle for choosing whether to tie the transfer to another exchange, and the requisite OK/CANCEL buttons.



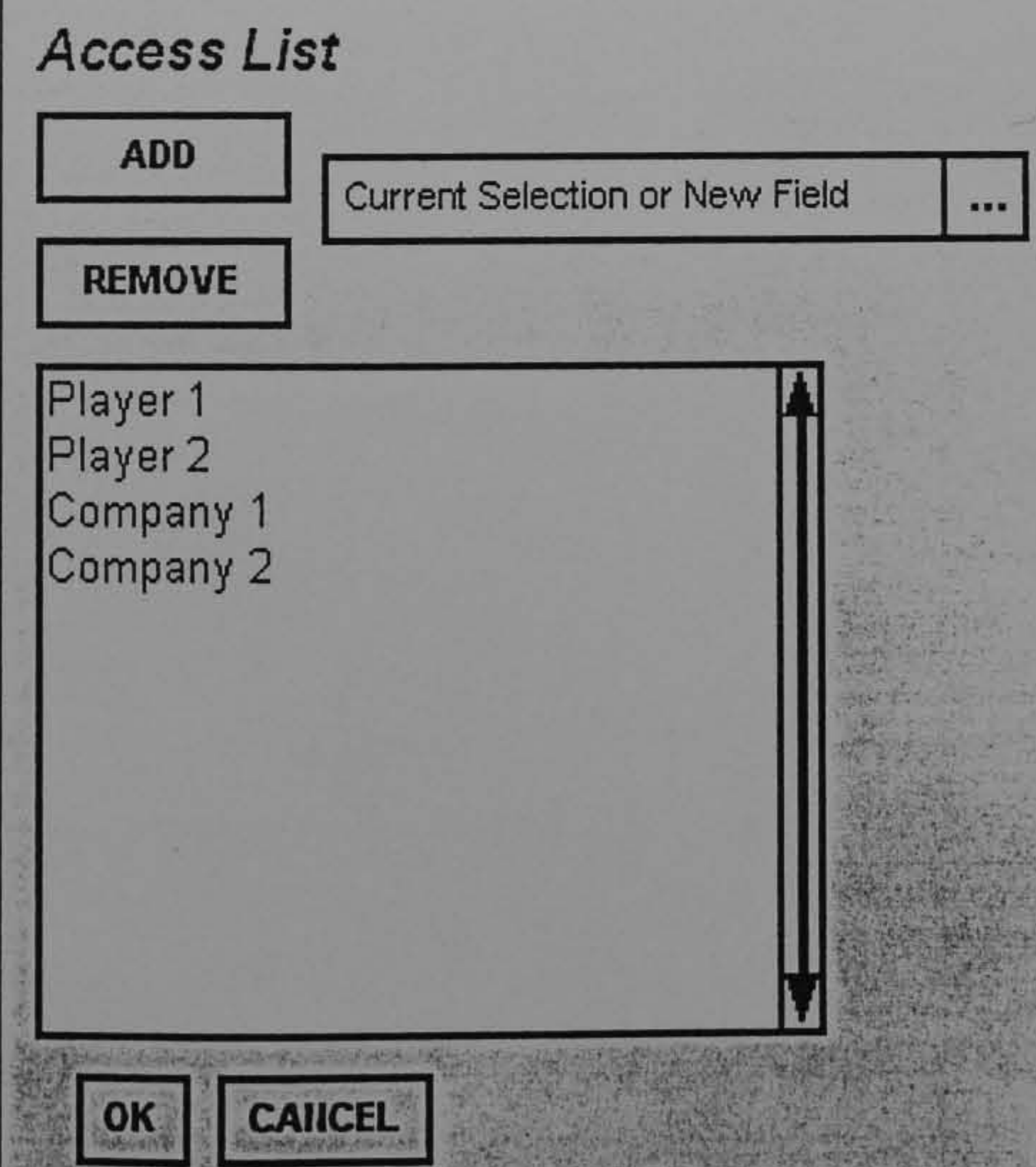
The **Ownership Transfer** dialog box contains the following elements:

- A text field labeled "Structure Name" followed by a downward-pointing triangle icon.
- A text field labeled "Location".
- A text field labeled "Prospective Owner" followed by an ellipsis icon.
- An unchecked checkbox labeled "On Exchange".
- Two buttons at the bottom: "CANCEL" and "OK".

Access List Interface

The access list interface is used to allow players or companies to use a structure or ship without *owning* it.

The elements need are a list field to display the current access list, a field display the



The **Access List** interface contains the following elements:

- An "ADD" button.
- A text field labeled "Current Selection or New Field" followed by an ellipsis icon.
- A "REMOVE" button.
- A list box containing the following entries: "Player 1", "Player 2", "Company 1", and "Company 2".
- Two buttons at the bottom: "OK" and "CANCEL".

current selection from the access list or from the name-picker controls for a new addition, ADD button for new additions, a REMOVE button for removals from the current list, and the requisite OK/CANCEL buttons.

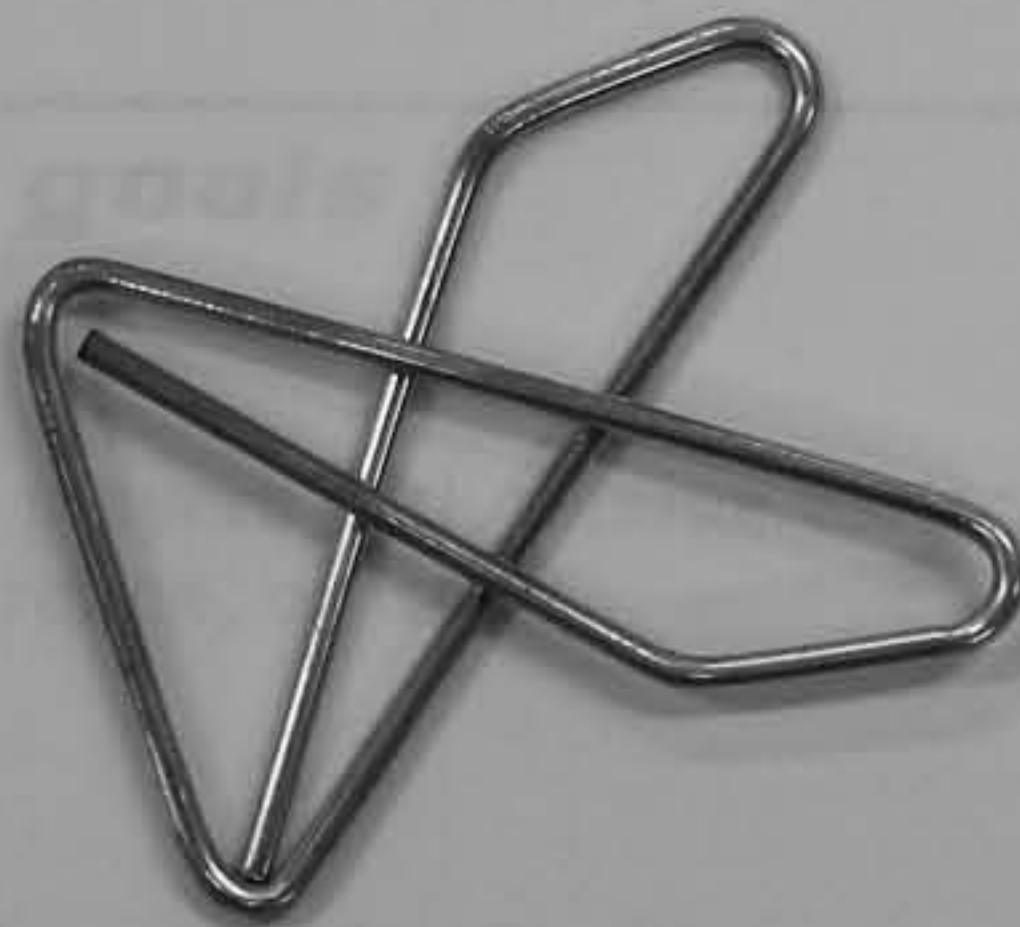


Tasks

- Art generated for the interfaces for Owner designation and Access Lists.
- Code generated for handling owners and access lists.
- Code generated to handle companies as owners or present in access lists.
- Code generated for a name search tool.

To be addressed

- A method to view reports on installations and structures that a player has access to, as opposed to specifically owning.
- A method to view ships that a player has access to, as opposed to specifically owning.



Walkthrough

Resource Categories

- Natural Resources
- Foods
- Fuels
- Metals
- Pharmaceuticals
- Contraband

Resources

Overview

This document broadly defines the categories of in-game resources required for everything from colonization and construction to space flight and the stock market. Resources lie at the heart of every player endeavor – they are harvested, refined, traded, used, hoarded, and protected.

(From the Vision Doc)

Raw resources are required for crafting of all sorts, as well as being required for fuel for many types of buildings and installations. There will be a range of installation types and resources to mine, creating ladders of advancement for prospectors and explorers.

To provide a variety of resources, descriptions can be mixed and combined, images can be hued, and attributes modified. Cephidian Grain and Xylani Oats require the same process for harvesting the resource but from separate star systems. Cephidian Grain is a pale blue, while Xylani Oats are light orange. When refined, Cephidian Grain makes a strong antibiotic. When combined with the right materials and refined, Xylani Oats become one of the most effective explosives in known space.

Vision and goals

Goals

- ☐ Provide the players with a large variety of resources that require different skills to harvest, refine, or trade.

Vision Doc

- ☐ will be easily extensible by the Live team
- ☐ will offer motives and rewards for short session play
- ☐ should encourage specialization
- ☐ should have a world that feels alive
- ☐ should support character building and advancement
- ☐ should have little touches that add depth
- ☐ should have secrets and lore to it
- ☐ will offer players lots of rewards
- ☐ should provide a sense of exploration

Walkthrough

Resource Categories

- Natural Resources
- Foods
- Fuels
- Metals
- Pharmaceuticals
- Contraband

Natural Resources

These are raw resources harvested or captured directly from their native environment. An attribute of all Natural Resources classifies them as either Renewable or Nonrenewable. Renewable resources generally fall into the Vegetable or Animal subcategories. Some Gases or Liquids may be renewable but at a much slower rate.

- Animals
- Vegetables
- Minerals
- Liquids
- Gases

Foods

These are raw or refined food resources derived from one or more Natural Resources, usually from the Animal and/or Vegetable subcategories. While food consumption is not required for a player to continue playing their character, Foods may be required for the Colonization process and also provide a reason for player socialization.

Fuels

These are refined fuel resources derived from one or more Natural Resources. Fuels are consumed by a wide variety of player actions, especially through transportation and refinement processes.

Metals

These are refined metal and alloy resources derived from one or more Minerals, required for a wide range of manufacturing and construction processes.

Pharmaceuticals

These are refined medicinal resources derived from one or more Natural Resources. Most Pharmaceuticals provide beneficial affects to individuals and possibly colonies.

Contraband

Some resources are inherently dangerous, such as viral agents or , and therefore classified as Contraband by ConFed.

Base Statistics and Advancement

Overview

In *Privateer Online*, each *avatar* is quantified by a set of basic statistics. These statistics are used to help determine and modify specific actions and/or events in the game. The majority of the statistics discussed in this document relate to the ground-based portion of *Privateer Online*. Any statistic not discussed here, will be discussed explicitly in the document that uses that statistic.

Vision and goals

GOALS

- ❑ To allow a player to affect his avatar's ability to function and interact during ground-based activities through his choices of basic statistics and their increase/decline.
- ❑ To represent statistics (other than Money) using a descriptive measurement instead of a numerical measurement. The point is to simplify the presentation on the player side, while retaining the depth of a numerical system.
- ❑ To represent these statistics visually on an avatar throughout play.
- ❑ To create a simple system of Money as credits. Using an interface, money can be withdrawn, transferred, used to purchase, and anything else without using a visual object to represent it.

VISION

- ❑ Provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ❑ Make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ❑ Encourage specialization
- ❑ Make players work to remain at their current level of achievement in the game
- ❑ Support character building and advancement
- ❑ Have little touches that add depth

Walkthrough

The basic statistics and their descriptions are as follows:

- Health (HEALTH): Health is a measurement of how much damage a given character can endure before going unconscious.
- Strength (STR): Strength is a measurement of a character's muscular strength and overall physique.
- Coordination (CRD): Coordination is a measurement of a character's reflexes, agility, and hand-eye coordination.
- Money (\$): Money is a numerical value of how much buying power a character possesses at a given time.

The major components required for this system are:

- A visual representation to gauge the various levels of Strength and Coordination.
- An interface for determining the initial values for Strength, and Coordination during initial avatar customization.

- Artwork (meshes and textures) to represent the differing physique models for representing the basic statistics.
- A set of interfaces to handle all types of transactions that use Money.

Systems design

Initial Determination of Basic Statistics

During initial *avatar* customization, the player will be presented with a set of interfaces to visually choose their *avatar*'s physique. Their choice of physique will determine the initial statistics of Strength and Coordination. This initial choice will also set the minimum level that Strength and Coordination can decay to.

Ratings

A descriptive rating, not a numerical rating, represents the three basic physical statistics. A descriptive rating was chosen to avoid the "number syndrome" common to most games that use statistics.

For Strength and Coordination, the following ratings are used (from worst to best):

Pathetic: Lowest possible rating, only reachable if a statistic was rated Inferior during customization.

Inferior: Below average. (Lowest possible starting score)

Mediocre: Average.

Notable: Slightly above average. (Highest possible starting score)

Superior: Normal Human Limit. Reachable through training and certification if starting rating was Mediocre.

Extraordinary: Above Normal Human Limit. Reachable only if starting rating was Notable and maximum training and certification.

For Health, the following ratings are used (from best to worst):

Full Health: This is the full health state.

Lightly Wounded: This state is reached when an *avatar* takes a wound.

Moderately Wounded: This state is reached when an *avatar* is lightly wounded and takes a wound. Some damage situations can cause a moderate wound from Healthy.

Seriously Wounded: This state is reached when an *avatar* is moderately wounded, and takes another wound. Some damage situations can cause a serious wound from Healthy.

Critically Wounded: This state is reached when an *avatar* is seriously wound, and takes another wound.

Unconscious: When an *avatar* is at critical and takes another wound, it goes into unconsciousness. If any form of medical attention is used on an *avatar* within 30 seconds, it will return to a critical rating.

Dead: If 30 seconds expire without medical attention, an *avatar* dies.

Strength

The Strength statistic will be used for the following purposes (and more may be added): Depending on the rating, it will increase or decrease the *avatar*'s resistance to damage when taken (in tandem with Coordination).

The rating will determine access to different types of equipment, such as powered armor or heavy weapons.

The rating will modify damage inflicted with a melee weapon.

Coordination

The Coordination statistics will be used for the following purposes (and more may be added):

Wing Commander Online: Privateer Base Statistics and Advancement

Depending on the rating, it will increase or decrease the *avatar's* resistance to damage when taken (in tandem with Strength).

The rating will modify an *avatar's* chances of successfully shooting or striking a target in ground-based combat, be it ranged or melee.

The rating will modify an *avatar's* attack speed in melee combat.

Changes to the Basic Statistics through Gameplay

A player can modify their *avatar's* basic statistics in a variety of ways during game play. The full range of options to improve and decrease the basic statistics is TBD. One of the primary ways is through the *Fitness* skill tree. This will provide a player with skill training to increase their *avatar's* Coordination and Strength, which in turn will be reflected visually on an *avatar*. Conversely, if a player chooses not to train in the basics of the *Fitness* tree, their *avatar's* Coordination and Strength will decay to a minimum point over time. (This minimum being set by their initial choices during *avatar* creation.)

Money

Money is a statistic representing the player's buying power. It is a fairly self-explanatory stat, and will fluctuate constantly as they buy, sell, and complete missions. It is important to note that money will **always** be represented as a numerical statistic. It will never be an object that can be dropped, equipped, and so forth. When exchanging money with other *avatars* or buying/selling, it will simply change the statistic to reflect the transaction.

Tasks

Below is a top-level list of tasks necessary to complete the production of this section.

- Artwork must be generated for the different body style ranges.
- An interface must be generated for initial *avatar* customization.
- A set of interfaces must be generated for the differing types of monetary transactions.
- Code needs to be implemented for interacting with the *Fitness* skill in both improving and decaying an *avatar's* physique.

To be addressed

Below is a list of areas that still need additional design work before this section can be sent to production.

- A set of descriptive levels need to be designed for Coordination and Strength.
- An interface must be designed for selecting all pieces of initial *avatar* customization.
- The full range of actions or events that can modify Strength and Coordination need to be designed.

Avatar Customization

Overview

Players in *Privateer Online* will be able to customize the look and actions of their online representation or *avatar* in a variety of ways. The initial customization will be their ability to affect and change their *avatar's* physical appearance during character creation. Throughout play of *Privateer Online*, the player will also be able to further customize their *avatar's* appearance in the form of *equipment*. *Equipment* being defined as clothing, armor, weapons, or any other item that may affect the appearance of a player's *avatar*. Also, players will be able to express their emotions through their *avatar* with animated gestures called *emotes* that the player's *avatar* will perform upon command.

Vision and goals

GOALS

- To give the players an emotional attachment to their avatar in the virtual world, through the following:
 - To give the players a wide variety of options to customize their appearance.
 - To provide a visual and tangible sense of accomplishment by displaying gained equipment on their avatar.
 - To provide the player an avenue of displaying personal taste in the physical appearance of their avatar.
- To give the players an enhanced sense of immersion in the virtual world, through the following:
 - A wide range of identifiable animated gestures to reflect a full spectrum of emotion.

VISION

- Customization must be easily accessible to a mass market audience and a casual player
- Customization should have an intuitive user interface
- Customization should blur real life and the online character
- Customization should have big visual payoffs
- Customization should have little touches that add depth
- Customization must give tools to players to shape their own experiences

Walkthrough

The major components required for this system are:

- Artwork (models and textures) for an array of body pieces, skin textures, hair, etc.
- An interface for customizing the appearance of an avatar at character creation.
- A method for choosing body style, facial construction, hair choices (including facial hair), skin textures and hue (including body hair), aging choices, and height.
- Artwork (mesh and textures) for an array of wearable equipment including clothing, armor, and weapons.

- A method to craft and customize a wide range of equipment; such as clothing, weapons, armor, jewelry, etc.
- A method for layering different types of equipment so that an avatar reflects 100% of what a player chose as equipment. [Optimal Situation, if not possible, fallback is art replacement for equipment]

Systems design

Initial Avatar Customization

The first opportunity for a player to customize their avatar is during initial character creation. They will be given the following choices:

- Gender
- Body style (from obese to emaciated), with an interface for affecting different points about the body differently. For example, a player may choose to have very thin legs yet have a paunchy stomach.
- Skin texture and hue. (Hues should cover the range of human skin tone) This area should also allow for choosing the amount and coverage of body hair.
- A face selection should be made up of choosing among different facial feature sets (such as brow, eyes, eyebrows, nose, lips, ears, and cheeks). Each set will contain a library of pieces for the player to choose from. (Eye color should be selectable from this area as well)
- Hairstyle can be selected by choosing different styles, lengths, textures, and hues.
- Facial hair can be selected by choosing different styles, lengths, textures, and hue for three distinct areas: Moustache, Beard, and Sideburns
- A method for the player to add visual aging effects to their avatar. (Including additive effects to current selections such as gradual graying of hair, hair loss, skin wrinkling through textures, and skeletal changes.)

Initial Avatar Customization Interface Goals

Each different area of initial customization should use simple controls found in any common Windows dialog library (such as slider bars, radio buttons, etc). The interface dialog themselves should fit contextually within Privateer Online's interface and look and feel. The benefit of using dialogs commonly found in Windows is to immediately make the player comfortable with the interface. The overriding priority is to strip away clutter and complexity, and the goal should be to have visual feedback for each and every option.

Continuing Avatar Customization

During the course of play, there will be several opportunities and situations that will affect the initial physical choices made by a specific player. The majority of these will require specific player action to initiate a change, but a select few will require specific player action to avoid.

The player through game mechanisms will be able to change:

- Physique by training and certifying in Fitness skills. These will allow the player to change his body shape as he increases through the ranks of Fitness skills.
- Physique by NOT training a minimal level in Fitness skills, and maintaining this level. The longer they neglect this, the more visual effect will be seen on the physical representation of the avatar. Note: this is an extremely gradual change.
- Hairstyle; including length, style, texture, and hue. This can be done at a specific facility (TBD).
- Facial Hair; including length, style, texture, and hue. This can be done at a specific facility (TBD).
- Body Decoration; including tattooing, scaring, branding, and piercing. This can be done with a specific Social skill and the required skill materials.

Equipment as Avatar Customization

The goal of customization with equipment is that each wearable or equip able item must be reflected as some sort of art on the player's Avatar. This includes multiple layers of clothing and armor in multiple positions on the body; as well, it includes items carried in both hands. If layering is deemed not feasible, we will proceed with the substitution method used in current technologies.

To accomplish this, we will need to employ several key technologies such as mesh deformation and using transparencies in textures to reflect the layering of equipment. For example, if Joe the tech decides to wear a nice shiny purple shirt under his work coveralls, the purple shirt must be visible under the coverall straps.

To reiterate, layering is the desired solution, if it is not feasible, we will implement a substitution method for equipment representation.

Emotional Responses via Animated Gestures (Emotes)

A wide range of character animations will be required to allow the player the widest possible spectrum of emotions. An initial list should include such animations as waving, cheering, rude gestures (yes, these are necessary), crying, kissing, and many others. A full list is TBD.

These emotes should be selectable easily from a dialog and also should be available to be mapped to hotkeys, or set up as automated responses via a options dialog.

We also will offer an emote construction dialog that will allow the player to string various animations together to create complex gestures, using the library of animated gestures that ship with the game. As well as any we add post-release. [Possible Post-Release Feature]

Tasks

Below is a top-level list of tasks necessary to complete the production of this section.

- Artwork for the body style ranges must be generated.
- Artwork for skin textures and hues must be generated.
- Artwork for hair textures and hues must be generated.
- Artwork for all three areas of facial hairstyle must be generated.
- Artwork for all hairstyles must be generated.
- Artwork for all face modular pieces must be generated.
- Artwork for aging texture overlays and other aging effects must be generated.
- An interface for initial appearance generation will need to be constructed using the interface language.
- Code needs to be implemented for interacting with the *Fitness* skill in both improving and decaying an *avatar's* physique.
- An interface for modifying hairstyle and facial hair changes will need to be constructed using the interface language.
- Code needs to be implemented for allowing the modification of both hairstyle and facial hair changes.
- Artwork for body decoration textures and meshes must be generated.
- An interface for applying and removing body decoration will need to be constructed.
- Code needs to be implemented for interacting with the *Social* skill that involves body decoration and its implementation on an *avatar*.
- Artwork and textures must be generated for all types of *equipment*.
- Code needs to be implemented that allows for complex layering of *equipment* so that it is logically visible on an *avatar*.
- Animations for gestures will need to be generated for all *emotes*. These must be generated with regards to modularity for the *emote* construction kit.
- An interface for constructing complex *emotes* will need to be constructed.
- An interface for mapping *emotes* to hotkeys will need to be constructed.

- A menu interface for engaging *emotes* will need to be constructed for use during play.

To be addressed

Below is a list of areas that still need additional design work before this section can be sent to production.

- A full list of hairstyles and facial hairstyles must be generated.
- A full list of modular face pieces needs to be generated.
- A full list of aging effects needs to be generated.
- An interface must be designed for selecting all pieces of initial avatar customization.
- A facility for modifying hairstyles and facial hairstyles changes must be designed.
- An interface must be designed for modifying hairstyles and facial hairstyles.
- A body decoration Social skill needs to be designed.
- A full list of body decoration art must be generated.
- An interface needs to be designed for adding/removing body decorations.
- A full list of gestures needs to be generated.
- An interface needs to be designed for constructing emotes.
- An interface needs to be designed for mapping emotes.
- Determine the feasibility of equipment layering.

VISION

- Easily accessible to a wide target audience and a casual player
- Offer a great onboarding experience for the first 30 minutes
- Satisfy the newbie in the first 5 minutes
- Easy to get into, as a new player and as an advanced one
- Offer a strong support system so that players feel welcomed into the game and so the game structure is friendly
- Offer clear goals to achieve
- Offer a chance to meet mentors (such as a guide) in the first five minutes

Walkthrough

Several different mechanics will be used to help adjust the new player to the game environment. Some of these mechanisms can also be used to train experienced players in features that are implemented post-ship by the live team.

The following is a list of the current mechanisms to be implemented, but it is important to note that the feature set may be revised several times before ship.

Simulator: The simulator can be used to help a new or experienced player train in almost any game system from space combat to crafting. Also, as the game evolves in post ship, it can be used to train all players in features that are added later in the Privateer's life cycle.

Graphics: Real time in contact with new players cannot be discounted. Based on the performance of the graphics in currently shipped products, we will be able to adjust our graphics design during beta and have it stable at launch.

In-Game Tutorials: Explaining tutorials are useful as the first line of entry for new players. It allows them to familiarize themselves with the basic functions of the game before moving on to more complex starting activities such as the simulator training or having a personal guide. Also, using the tutorials as player profiles will help assist

Advancement: Starting Activities

Overview

Helping new players accustom themselves to the game is probably the single most important facet to building and retaining a customer base. This document intends to give a high level survey of the mechanisms that will be used to help the new player learn the game.

Vision and goals

GOALS

- ☐ Introduce the new player to the basic game mechanisms needed to begin play.
- ☐ Help the player identify areas that might be of interest to them.
- ☐ Offer human assistance and interaction during the initial moments of their experience.

VISION

- ☐ Easily accessible to a mass market audience and a casual player
- ☐ Offer a great newbie experience for the first 30 minutes
- ☐ Satisfy the newbie in the first 5 minutes
- ☐ Easy to get into, as a new player and as an advanced one
- ☐ Offer a strong support system so that players feel welcomed into the game and so the game atmosphere is friendly
- ☐ Offer clear goals to newbies
- ☐ Offer a chance to meet someone (such as a greeter) in the first five minutes

Walkthrough

Several different mechanisms will be used to help adjust the new player to the game environment. Some of these mechanisms can also be used to train experienced players in features that are implemented post-ship by the live team.

The following is a list of the current mechanisms to be implemented, but it is important to note that the feature set may be revised several times before ship.

Simulator: The simulator can be used to help a new or experienced player train in almost any game system from space combat to crafting. Also, as the game evolves in post-ship, it can be used to train all players in features that are added later in the Privateer's life cycle.

Greeters: Real human contact with new players cannot be discounted. Based on the performance of the greeters in currently shipped products, we will be able to adjust our greeter design during beta and have it viable at launch.

In-fiction Tutorials: Beginning tutorials are useful as the first line of entry for new players. It allows them to familiarize themselves with the basic functions of the game before moving on to more complex starting activities such as the simulator training or having a personal greeter. Also, using the tutorials as player profilers will help assist

the player in determining what might be interesting activities to pursue throughout the game.

Availability of Basic Equipment: It will be extremely important to supply the new player with an array of basic items to help them get a foothold on their activities when they are just beginning. These basic items will be of the lowest quality and will be cheap to purchase, allowing players to get a feel for the game systems before investing large amounts of time in acquiring more desirable items.

Apprenticeship/Mentoring: see the document on *Apprenticeship*.

Systems design

The Simulator

The simulator will be used as a flexible catchall tool to help accustom the new and experience player to new game mechanics and let them practice their skills at space flight, combat, racing, ground-based combat, and any other aspect of the game we wish to structure in the simulation. The simulator will also be a very powerful tool for extending the game with new features and allowing players to learn these new features in a relatively risk-free environment. The only downside for the player is that none of the experiences can be used for in-game rewards other than increasing their simulator gains.

Some examples of simulator scenarios are:

The Gauntlet: In this simulation, the player(s) would start with a minimally configured ship with low powered weaponry. They would be pitted against increasingly more difficult AI controlled ships in different space settings such as asteroid fields, free space, etc. After each wave of enemy, they would be given incremental increases in their simulator ship. Also, data would be stored on the player that allows them to re-enter the Gauntlet at the level they left off.

The Arena: Similar to the gauntlet except that it would require multiple players. The players would face each other in different space settings with the high scorer of each round getting an incremental increase in their simulator ship.

The Science Fair: This simulation would take place on a pre-generated ground environment that had an unlimited supply of different types of materials for crafting. Allowing players to familiarize themselves with the *build templates* that they have access to. It would also allow them to try more exotic types of materials to let them experiment.

Greeters

Similar to the current system implemented by Player Relations and the UO Live team, Privateer Online will have volunteer-based greeter system that allows new players a chance to get a human-based introduction to the game.

In-Fiction Tutorials

Through initial avatar customization and the first time game experience, a player may elect to participate in several tutorials. These tutorials are structured to help train new players in the basic functions of the game such as operating their PDA, etc.

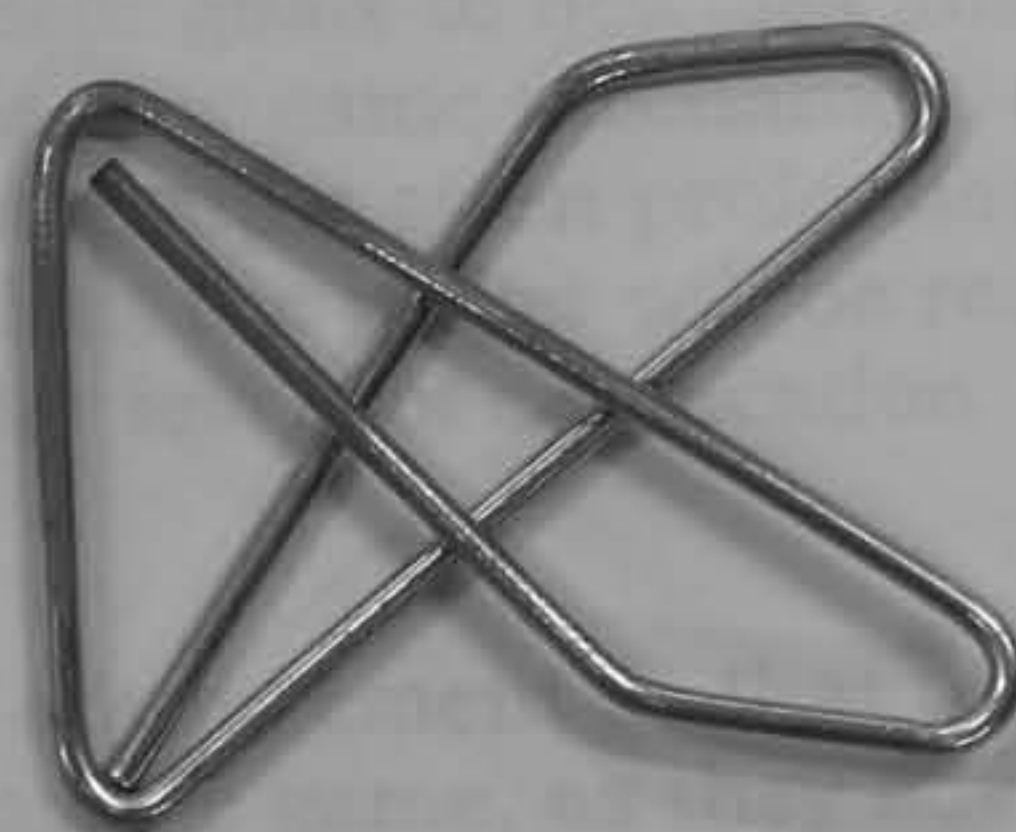
There will also be tutorials that help the player identify what areas of the game that they might be interested in. An example would be a system similar to the OEM additions to Ultima Online.

Availability of Basic Equipment

To further encourage new player participation during their initial experience, a full array of basic materials, equipments, ship add-ons, and other items will be available in unlimited supply at a minimal cost to the new player. These items are the lowest quality available in the game, but they allow the players to instantly equip themselves for any type of activity they wish to pursue.

Apprenticeship/Mentoring

To encourage continual support of new players from the existing player community, experienced players will be given incentive to help mentor the beginning players. (The *Apprenticeship* document covers this in further detail.)



Advancement: Profession Tree

Overview

Advancement in *Privateer Online* will ultimately be governed by up to five barriers: One-time Monetary barrier, Recurring Monetary barrier, Reputation barrier, Related Tree barrier, and Accomplishments barrier. While it will be uncommon to see all five barriers on any given certification, at least one barrier will always be present. These barriers will be in place to regulate the player's time and effort taken to complete a certification. Also, the information and ability to certify should be in the player's reach at all times. There should be time and effort involved in overcoming the barriers, but not in obtaining the certification itself.

Vision and goals

GOALS

- ☐ The information on specific advancement should be instantly accessible to the player at all times to offer clear goals to newbies and advanced players alike.
- ☐ The five barriers will provide new game mechanics, etc., as they are discovered during a player's advancement up a given profession tree.
- ☐ At higher levels of certification, the player will be required to put forth effort and time to maintain these higher levels of certification

VISION

- ☐ Provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ Have a technology tree that unfolds over time
- ☐ Permit players to switch professions without starting the character over
- ☐ Make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ Encourage specialization
- ☐ Make players work to remain at their current level of achievement in the game
- ☐ Support character building and advancement
- ☐ Never make the player feel ripped off by the game mechanics

Walkthrough

Advancement in a specific profession tree is governed by up to five barriers. These barriers are used to moderate and balance the advancement as a player gains certifications in a given profession. Each barrier is not necessarily required for every certification of a given tree, but one barrier will always be present. The five barriers are:

1. One-time Monetary barrier
2. Recurring Monetary barrier
3. Reputation barrier
4. Related Tree barrier
5. Accomplishments barrier

A typical Gameplay example:

Jimmy's avatar Irin is a merc for hire, someday being totally capable of cleaning the galaxy of any scum for a price. Currently, Irin is still a Certificate 1 Pilot. He just completed his first mission, and decides to check the requirements for advancing to Certificate 2. To do this, he activates his PDA and accesses the Certification section. Accessing his current file, he clicks on the ADVANCEMENT INFORMATION button under the Pilot skill tree representation. The display reports that he needs a one-time registration fee of 300 credits [One-time Monetary barrier] and at least six kills in space combat [Accomplishment barrier]. Realizing he has not quite met these goals, Irin uses the PDA to access the current system's mission board to find work to put him on his way to Certificate 2.

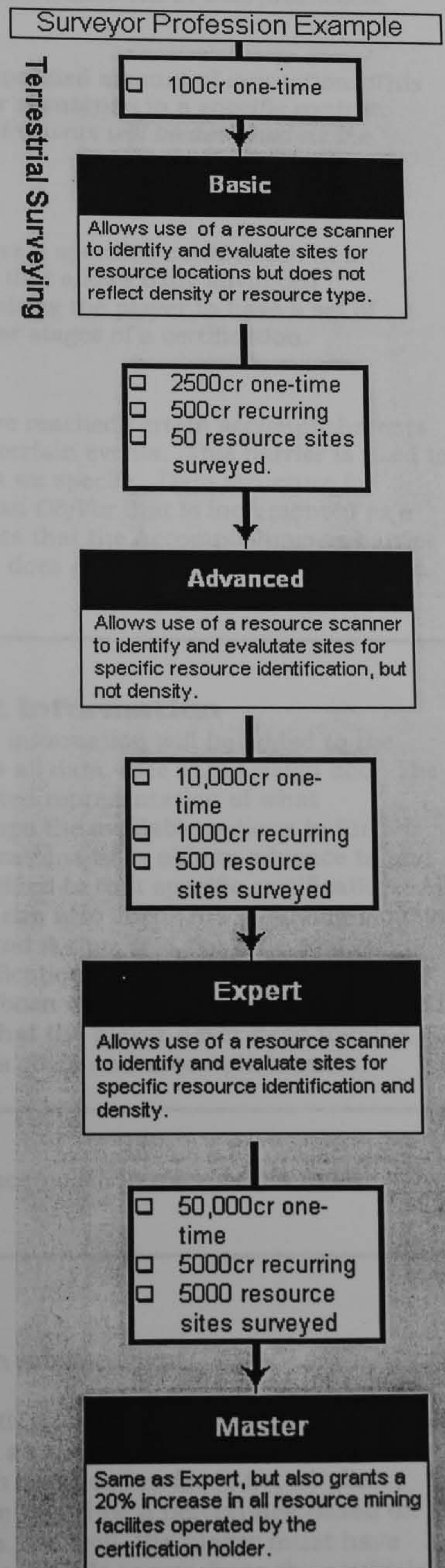
Systems design

One-time Monetary barrier

Each Certification will cost a specified amount of money. This is a one-time fee that is paid to advance to that given certification. Each progressive certification will require a larger fee. This fee is paid when the player chooses to advance the certification.

Recurring Monetary barrier

To maintain a given certification in a tree, a specified due must be paid per a specified time amount. Each progressive certification will require a larger due to be paid per time period. Not all certifications will have a recurring monetary barrier, but most advanced certifications will. The recurring fee will be deducted from the player's money account automatically, and if the resulting deduction is insufficient, the player will be notified and given



a grace period before he is demoted to the previous certification in that profession.

Reputation barrier

Advancing to certain certifications will require a specified amount of reputation. This may require a certain level of overall reputation or reputation in a specific context. *Since the Reputation system is TBD, the exact requirements will be designed as the Reputation system is fleshed out.*

Related Tree barrier

Certain certifications will require that a player have a specified certification in a separate profession tree. This is a balancing tool that allows truly advanced certifications be treated as specializations by requiring the player to have a set of different certifications before advancing to the later stages of a certification.

Accomplishments barrier

Certain certifications will require an *avatar* to have reached certain accomplishments via incremental counters that is manipulated by certain events. This barrier is used to require the player to put forth an amount of effort we specify. Data structure for accomplishments would be something similar to an *ObjVar* that is incremented as a specific action is performed. It is important to note that the Accomplishments barrier can be made up of several accomplishments, and does not necessarily mean a single accomplishment barrier per certification.

Interface design**Tree Information and Advancement Information**

The interface for viewing profession advancement information will be added to the multi-function PDA that the player uses to access all data. (See PDA Design doc) The specific section dealing with professions is a tracked representation of what certifications the player has acquired and also maps the available options to further certifications. If a player selects one of the certifications he is able to advance to and clicks the ACQUIRE button, they will then be certified in that specific certification. All monies will be deducted at that time. The player can also check his recurring monetary contributions per certification, when the next round is due, and the total cost of recurring contributions. If a player selects a certification that he has not met the criteria for, it will display which criteria have not been met. The goal of the interface is to be as simple possible but have enough depth that the player never need touch a manual or help file to reference their certifications goals and accomplishments.

Tasks

Artwork for the Profession Information and Advancement interface needs to be generated.

To be addressed

- Generate two example Profession trees.
- Discuss Additional Options to Certification Advancement

Player Trainers: An additional profession with its own set of sub-games could be added that allowed players to develop *avatars* as trainers. These trainers could purchase *certification kits* at a lower price than certification fee is set at. To discourage exploitation, this reduction of price would be a percentage based on the player's certification in the *Training* profession. As well, the player must have completed his certification in any profession to be able to purchase the equivalent *certification kit*. The player could then set up a facility for other players to come and purchase *certification kits* from him, or alternatively the player could become a

traveling trainer who was able to certify at a reduced fee. These *certification kits* would require resources and credits similar to any other manufactured good. The highest levels of certifications in certain skills perhaps could in fact only be obtained from player run facilities or player trainers.

- Generate an interface mock-up for the Profession Information and Advancement dialog.
- Generate the actual Profession trees.
- Generate the Reputation System document, and revise this document with those changes.
- Design how to scale the one-time and recurring fees so that they do not fall victim to deflation.

Vision and goals

Goals

- 1. To leverage older players as a way of training newbies
- 2. To incentivize older players to be friendly towards newbies
- 3. To enhance community building spirit in the game
- 4. To provide a motive for players to become integrated into the community

Vision

- 1. ... must be easily accessible to a mass market audience and a casual player
- 2. ... must offer clear goals to newbies
- 3. ... must offer a strong support system so that players feel welcomed into the game and as the game atmosphere is friendly
- 4. ... must offer a chance to meet someone (such as a mentor) in the first few minutes
- 5. ... should provide an advancement system so that players discover new skills and powers as they play the game, as they advance
- 6. ... must support a reputation system and advancement
- 7. ... must provide a place for players to meet and build bonds with other players
- 8. ... will provide a place for players to meet and build bonds with other players
- 9. ... will offer players lots of opportunities to interact with other players
- 10. ... must encourage and reward players who help other players
- 11. ... should encourage a strong relationship in the game world
- 12. ... should have features for everything in the game, including social activities
- 13. ... will try to fulfill all of Alex's Kim's principles of community building

Walkthrough

- Players can establish a formal "mentor-apprentice" relationship between two characters.
- The mentoring character derives direct benefit from the association, in the form of reduced certification fees and availability of higher certification levels.
- The apprentice derives direct benefit from the association, in the form of availability of higher certification levels.

Systems design

Forming a mentor-apprentice relationship

A character may petition a player who has an appropriate level of certification to become an apprentice of that player. Apprentices are required to be at a certain level of

Advancement: Apprenticeship

Overview

Players will be able to take "apprentices," players less advanced than they, which will provide an implicit character development advantage to both the master and the apprentice.

Vision and goals

Goals:

- ☐ To leverage elder players as a way of training newbies
- ☐ To incentivise elder players to be friendly towards newbies
- ☐ To enhance community-building spirit in the game
- ☐ To provide a motive for newbies to become integrated into the community

Vision:

- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... must offer clear goals to newbies
- ☐ ... must offer a strong support system so that players feel welcomed into the game and so the game atmosphere is friendly
- ☐ ... must offer a chance to meet someone (such as a greeter) in the first five minutes
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... must support character building and advancement
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... will provide privileges to experienced players
- ☐ ... will offer players lots of rewards
- ☐ ... must encourage and support a community
- ☐ ... should encourage a sense of player ownership in the game world
- ☐ ... should have ladders and leader boards for everything in the game, including social activities
- ☐ ... will try to fulfill all of Amy Jo Kim's principles of community building

Walkthrough

- Players can establish a formal "mentor-apprentice" relationship between two characters.
- The mentoring character derives direct benefit from the association, in the form of reduced certification fees and availability of higher certification levels.
- The apprentice derives direct benefit from the association, in the form of availability of higher certification levels.

Systems design

Forming a mentor-apprentice relationship

A character may petition a player who has an appropriate level of certification to become an *apprentice* of that player. Apprentices are required to be at a certain level of

certification before they can be taken on, in order to prevent abuse by just making throwaway newbies to enhance apprenticeship standing.

The *mentor* must formally accept the apprentice in order for the relationship to be officially established.

Benefits to the mentor

The number of apprentices serves as a pre-requisite for specific certifications. In particular, the higher reaches of professions can never be reached without having at least one apprentice, and there are branches of certain professions that have higher requirements.

You can't open up the higher levels of engineering research without five "graduate students" to back you up, for example.

Once the mentor has reached the threshold of apprentices, he or she is now forever after eligible to reach the levels of certification that require apprentices, even if the apprentice is later lost.

The number of apprentices can also affect the fees you pay for certification. The guilds that run the certification racket look kindly upon the recruitment process, and therefore having apprentices earns you a discount on your own periodic dues.

Benefits to the apprentice

Certain profession certifications are more costly if you do not have a sponsor of a higher status than you. Acquiring a mentor therefore saves you significant amounts of money in initial certification fees (it does not affect the ongoing dues, however, only the initial fee).

It is possible to have a "member of the school of" effect going on later on in the game's development as well, whereby students acquire some of the crafting characteristics of their mentor (such as perhaps their maker's mark, or skill bonuses).

Under specific group conditions (eg, in a party, squadron, or on a capital ship) the apprentice can receive bonuses to his skill success rolls if his mentor is a member of the same group, based on a proportion of the amount by which the mentor's skill exceeds the apprentice's.

When a mentor pilot takes out an apprentice pilot in his squadron, his expertise in gunnery may actually enhance the apprentice's targeting ability.

When a mentor surveyor is in a party with his apprentice, the apprentice has better skill results on his random rolls for surveying.

Interface design

- A special interface will be required for the skills display that allows mentors to name individuals as apprentices.
- Mentors must be able to see who their apprentices are.
- Apprentices must be able to quit their mentors at any time, from anywhere via a simple command.

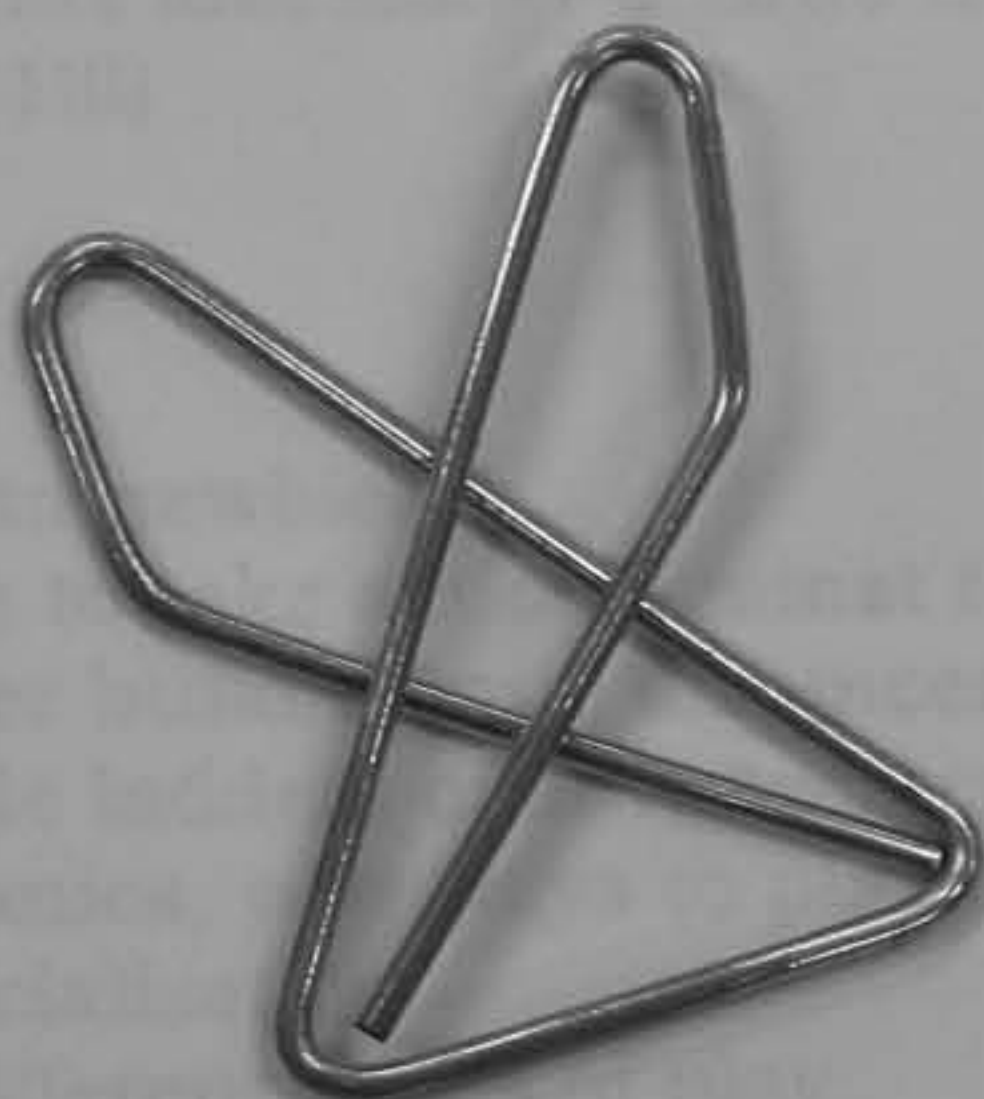
Tasks

- Integrate apprenticeships and mentoring into the requirements for specific certification levels.

- Create interface for petitioning a mentor
- Create interface for accepting an apprentice
- Create interface for losing an apprentice

To be addressed

- How will skill bonuses work?
- All the interface stuff needs sorted out.
- What are the profession tree cutoffs?
- What are the certifications available to "teachers" (those who accumulate lots of apprentices)?
- How would "member of the school of" work?



Reputation

Overview

If ever there was a catchall system, reputation is it. Your rep defines who you are in WCO. Be it your standing on the leader boards, your financial information, the number of pirates you've scragged or heroes you've burned, your reputation is how you know where you stand in the world. This is one of the most intensive systems WCO will have, hooks from it will be in nearly every other system we design and implement. As such, it will be continually modified and grown in tandem with other documents progress. This doc should present a high level view of what we wish it to do and how it might do it.

Vision and goals

Goals

- ☐ Provide ways for players to know what kind of people they are dealing with
- ☐ Provide ways for players to warn others as to what kind of people they are dealing with
- ☐ Create deep reward systems and leader boards that lure players into friendly competition (I must be #1!!!)

Vision Points

- ☐ ... must offer clear goals to newbies
- ☐ ... must empower players to take action against those who are obnoxious
- ☐ ... must support character building and advancement
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... should encourage specialization
- ☐ ... must support many different styles of play
- ☐ ... must have little touches that add depth
- ☐ ... must make use of humor and personality
- ☐ ... must never make the player feel ripped off by the game mechanics
- ☐ ... will offer players lots of rewards
- ☐ ... must have defining moments that are easily remembered. These could be player created.
- ☐ ... must encourage and support a community
- ☐ ... should encourage a sense of player ownership in the game world
- ☐ ... should have ladders and leader boards for everything in the game, including social activities

Walkthrough

The topics to be discussed include:

- Where do we start from and where do we go?
- What will Reputation determine?
- Perceived Reputation and Actual Reputation
- Murder and Mayhem – PKs and Anti-Pks...
- Lies and Deceit – Enabling Players to rate their peers, EBay style.

- Princes and Paupers – Who has the best financials?
- Champions and Losers – Sports and Competition

DDR NOTE: As stated, this is simply a high-level overview...details to come as parallel systems are fleshed out.

Systems design

Where do we start from and where do we go?

Building off of the work done in UO and furthered by the design for UO2 we hope to craft a reputation system that not only gives players adequate information about the types of folks they play against but to create new systems of ranking, prestige, and infamy with automated News.Net reports, leader boards and other devices.

What will Reputation determine?

Reputation determines whether you serve the forces of good or evil, how good a pilot you are, how successful a businessman you are, and a host of other individual's rankings.

For our purposes it will rate your alignment as a criminal, a citizen, or a hero. This will then shut down and open up areas of the game for whichever direction you head.

For example: A pirate may no longer be able to enter a Safe Space system because of the body count he left in his wake, but he might be given access to illicit systems and cloaked beacon locations that goody two shoes can't get the maps to.

Most importantly it will allow other players to know where you stand in the universal scheme of things. "Will you kill me or not?" "Can I trust you with this delivery?"

Perceived Reputation and Actual Reputation

Perceived reputation is how other player's see you, but doesn't affect your access to systems and Confed or the Syndicate's propensity to want you dead. Leader boards, news.net, public commentary/profiles and other in game devices support this.

Actual reputation is based on criminal or heroic acts. 'Getting caught smuggling, murdering a transport full of miners, taking out a pirates' nest...these are the things that effect actual reputation.

Murder and Mayhem

The simplest method of tracking a player's reputation is by what they kill.

Killing AI enemies, AI friends, neutral players, good players, and evil players all add up in the end. Whether you target good or evil will determine the reputation you head for.

Exactly how many kills are required to move through the scale as well as the scale's dimensions is TBD.

The greater a hero or villain you are will increase not only your personal reputation, but your public visibility. If you are the criminal with the most murders or hero with the most saves on a

The dream is to create the same desire for stardom that folks have in the real world then giving them the opportunity of dealing with the pressures of overexposure once they reach that goal.

server, News.Net may actually track your progress through space, reporting on your every move like the celebrity you are.

Slip into villainy and you might become the target of Confederation endorsed bounty hunters looking to settle the score and cap a reward. Kill one too many bad guys as a bounty hunter or hero and the Syndicate may take a contract on you, sending Syndicate endorsed bounty hunters after you.

Bounty hunters are neutral to an extent, but their reputations slide up and down the same way heroes and villains do. You just have to decide if you're Boba Fett or Steve McQueen.

Smuggler's Blues

Smuggling in and of itself **does not** lower your reputation. Getting caught smuggling, on the other hand does. How this impacts the reputation scale is TBD.

Lies and Deceit

For players who engage in fraud on the commodities market (failed deliveries, wrong item) we will implement an EBay style player profiler so that other individuals can evaluate what kind of seller/trader/trucker they trust their credits with.

This system, when fully designed, should be the players' primary method of handling and rating those they respect or despise.

If a player's profile rating falls below x level and stays there for x period of time it will effect his actual reputation.

Princes and Paupers

The more money you have the more important you are...right? In WCO this is the case. Your financial Reputation can effect your personal reputation through the means by which you made your money (mining planets dry, vs. pleasure planet gardens vs. illicit drug manufacturing/distribution).

The richer you get the more visible you become in the game, much like sports heroes and criminals.

News.Net local flash: Famed industrialist {name} known for his mining, refining, and fabrication operations within the {system} system landed on Earth today.

Champions and Losers

WCO will offer a wide range of Simulator tournaments and real space "arena" events. Participants in these games will follow a traditional sports ladder and be ranked accordingly.

We will have an expansive leader board system with ranking for stats etc.

Leader boards

We hope to have leader boards for just about every activity.

Who is the best weapons manufacturer? Who is the most dangerous pilot? What company has the most valuable portfolio? Who is the best hairstylist?

Your presence on these boards is an indication of your perceived reputation amongst other players. These things may influence your actual reputation, but for the most part leader boards are cosmetic rewards that promote competition across the skills, certifications, and actions found in the game.

- 

-
- A close-up photograph of a standard metal paperclip, likely made of steel, resting on a light-colored, textured surface. The paperclip is bent into its characteristic shape, with one end forming a small loop and the other end being slightly curved. The lighting is soft, highlighting the metallic sheen and the sharp edges of the wire.

Leader Boards

Overview

Leader boards provide “king of the hill” style ladders for players to climb as part of their advancement. Even after maxxng out their potential advancement in a given profession, they can still compete in terms of which player is the best in a given tracked endeavor. Since there is no “winning” this (as a lower-down player on the ladder can always overtake you) this provides an ongoing challenge to the advanced player. Leader boards also provide ways to measure standing in game areas that are social endeavors, such as barkeeping, etc.

Vision and goals

Goals

- ☐ Provide a means of ranking all types of endeavors within the game
- ☐ Provide an “infinite game” that does not lead to “maxxing out”
- ☐ Provide milestones and advancement mechanisms to players
- ☐ Give data viewable on the website

Vision

- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... should have an intuitive user interface
- ☐ ... must have a high retention rate for subscribers
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... should encourage specialization
- ☐ ... should make players work to remain at their current level of achievement in the game
- ☐ ... must support many different styles of play
- ☐ ... will satisfy Achiever-oriented players
- ☐ ... will satisfy Socializer-oriented players
- ☐ ... will satisfy Explorer-oriented players
- ☐ ... will satisfy Killer-oriented players
- ☐ ... should support character building and advancement
- ☐ ... should have little touches that add depth
- ☐ ... will offer players lots of rewards
- ☐ ... should have defining moments that are easily remembered
- ☐ ... must encourage and support a community
- ☐ ... should have ladders and leader boards for everything in the game, including social activities
- ☐ ... will offer robust support for players to form guilds and organizations
- ☐ ... will try to fulfill all of Amy Jo Kim’s principles of community building

Walkthrough

Every conceivable piece of data that can be ranked regarding players and buildings will be extracted from the database and used to generate leader boards. We will provide custom searches for specific high-profile leader boards, but players will also be able to make custom queries.

Leader boards are available both on the website and in the game.

Systems design

The game database is stored in a standard database format that can be queried by a web server. The web server is able to do queries and rank the results via simple server-side algorithms, then generate the webpage dynamically and serve it to the web client. The technology here is not at all dissimilar to what is already in common use on the current Ultima Online website as well as in games such as the recently released Unreal Tournament Demo.

For in-game leader boards, the same process is followed, except that it is a game client, not a web client, doing the querying and resultant display.

Note on a TBD feature: If the client has an embedded web browser, then there should only need to be one piece of code written, on the web side, to be able to integrate full functionality into the game.

Examples of leader boards we wish to have available include:

- ☐ Most ship kills
- ☐ Most successful missions run
- ☐ Most kills with a given ship type
- ☐ Most kills with a given weapon type
- ☐ Most pirate kills
- ☐ Most bounty hunter kills
- ☐ Most planets discovered
- ☐ Most installations
- ☐ Most installations per type
- ☐ Most productive installation
- ☐ Largest cities
- ☐ Most popular public building
- ☐ Most popular public building by type
- ☐ Most profitable business
- ☐ Most profitable business by type
- ☐ Largest guild
- ☐ Deadliest guild
- ☐ Most read news writer
- ☐ Highest reputations by type
- ☐ Etc.

Interface design

The standard leader boards should be available within the game via the PDA. A given leader board is simply a top ten list. Selecting a given line item on the list may or may not generate a report on that particular business, guild, or ship.

Tasks

The bulk of the work should be in doing the web interfaces and search queries.

To be addressed

- ☐ What exactly are the items we can search by?
- ☐ Will there be an embedded web browser in the client?

Planet Spawning

Overview

Planets in *Privateer Online* are algorithmically generated terrain with a wide range of characteristics. They are generated from a fixed random seed so that maps can be easily sent down to the client as a single number or small series of numbers. When planets are idle for an extended period of time, we delete the planet and its associated jumpgate.

Vision and goals

The vision is that of an ever-expanding map that changes based on the actual play habits of the players. As they choose to go forth and discover new jumpgates, they actually force new space volumes and planets to be created. Planets also afford a different play experience that is not as skill-centered as space combat, and one that broadens the potential audience of the product. Ground also offers the greatest scope for social interaction in the game, and has a much smaller bandwidth requirement than the space section.

- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... should have a sense of discovery
- ☐ ... should capture "sci fi fans who are Internet connected" whether they are gamers or not
- ☐ ... must have a high retention rate for subscribers
- ☐ ... must provide barriers to exit
- ☐ ... should have low support costs
- ☐ ... should have a low target bandwidth
- ☐ ... will be easily extensible by the Live team
- ☐ ... must be a scalable platform so that it remains viable several years after launch
- ☐ ... will make use of technology that allows updating the game on the fly after launch
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... will make use of "rock-paper-scissors" relationships so that no one specialty, item, or skill is paramount
- ☐ ... will offer tools for player-driven content
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... should have a world that feels alive
- ☐ ... must support many different styles of play
- ☐ ... will satisfy Achiever-oriented players
- ☐ ... will satisfy Socializer-oriented players
- ☐ ... will satisfy Explorer-oriented players
- ☐ ... will satisfy Killer-oriented players
- ☐ ... should have secrets and lore to it
- ☐ ... should provide a sense of exploration
- ☐ ... will offer great cultures to experience
- ☐ ... must encourage and support a community
- ☐ ... should encourage a sense of player ownership in the game world

Walkthrough

- The base map
 - Existing jumpgate linkages
 - Already established planets
- Discovering a new planet
 - Opening a new jumpgate
 - The associated space volume
- The terrain generation routine
- Flora and fauna
- Climate and atmosphere types
- Closing a jumpgate

Systems design

Generation of space volumes

Space volumes will be generated by either missions or by players discovering them. Discovery of a jump point is done only by players at the top of the profession ladder for surveying and cartography.

Space volumes are generated via a fixed random seed. When the seed is used, its location on the map is noted and it is reserved for that location only. Duplicate generation routines exist on both the client and the server side, and the only element transmitted in order to replicate the map is the seed itself. This minimizes the amount of data transmitted and gives us an arbitrarily large universe.

The skybox proper is generated using standard routines for placement of stars and background nebulae that are generated fractally. Major elements to be placed within the space volume should include:

- ☐ Gas clouds of various types and colors
- ☐ Asteroids, either single or in a belt
- ☐ Derelict ships
- ☐ Planets

See the document on special encounters for details on possible features.

Planets/stations within volumes

Some types of features in space are doorways to other volumes (jump points, planets, space stations) and others are merely static objects within the volume that can serve as obstacles, special playfield modifiers (eg, nebulae that cost extra fuel to navigate and that impair visibility), or targets or spawn points of targets.

Doorways have random seed numbers associated with them as well, and a subset of the generation routine appropriate to the type of volume the feature will be (eg, planet, station, more space) is run in order to generate the external appearance of the feature in this space volume. Using a small version of the full algorithm, we can thus make the planet have the correct atmosphere and terrain colors when seen from space, without actually doing the complete map generation.

Generation of planetary terrain

The requirements for planetary terrain are that it appear dramatic and alien, with a wide range of potential variability in appearance and environment.

Planets are generated using the same random seeds as space volumes. The terrain will be either a height map or a true arbitrary mesh (height map is the default solution at the moment) generated via a custom routine that is tweaked to maximize hyper-realistic

terrains with exaggerated cliffs, valleys, and other features. This is because of the overhead oblique perspective we intend to use, which tends to foreshorten vistas.

One possible method for the generation of this height map is to superimpose via additive blending several artist-crafted shapes for standard terrain features (slip fault lines, water eroded valley patterns, oxbow lakes, etc) on top of a basic height map field, then running a noise filter on the whole thing to cause it to have variability.

Texturing of the terrain is ideally done via procedural textures, with transitions between two adjacent differing textures created dynamically. Standard algorithms for tile-based transition patterns (such as those used in every Ultima since Ultima VI) should be used. The angle of specific polygons can be examined in order to replace standard terrain textures with cliff faces or embankment textures.

All terrain textures should be huable, so that we can make use of the same routines on as many planets as possible. In general, a planet's color makeup should consist of three primary colors: soil color, "grass" or ground cover color, and rock color. The actual colors used should be within a circular range of the selected hues.

Terrain types that we will need will include, among others:

- ☐ Ice
- ☐ Grasslands
- ☐ Desert/exposed rock
- ☐ Jungle
- ☐ Sand
- ☐ Forest
- ☐ Water

See the documents on *Flora and Fauna* for information on the population of the planets.

Other characteristics that will need to be generated include:

Planet size:

- *Normal (the standard 256x256m grid)*
- *Tiny (for use only with asteroids and other small bodies)*

Atmosphere:

- *None (requires special breathing apparatus)*
- *Breathable (no special equipment required)*
- *Poisonous (requires special breathing equipment)*
- *Corrosive (requires special breathing equipment, and damages equipment)*

Resources

- *A renewable resource (agriculture, vented gas, life forms)*
- *Non-renewable resources (minerals, fossil fuels, etc)*

Colors for textures:

- *Grass or equivalent*
- *Soil*
- *Rock*
- *Liquid*
- *Lighting*
- *Weather effects*

Densities for above colors/effects within textures:

- *Grass*

- Soil
- Rock
- Liquid
- Light
- Weather effects

Collapse of volumes

Volumes of map will be collapsed automatically after there have been no players logged in on the volume for a half hour (or other time span arrived at to minimize server impact). The dynamic objects present on the planet will be saved, and the process terminated. It will relaunch when a player once again sets foot in the volume. When the volume is relaunched, all static data such as the map is regenerated from the fixed random seed.

Interface design

Tasks

- Process management system for launching and destroying volumes.
- Skybox generation routines.
- Nebulae generation routines.
- List of environmental factors for space volumes (nebulae, planets, etc)
- Art assets for environmental factors.
- Terrain heightmap generation routine.
- Terrain texturing routine.
- Procedural texture routines.
- Texture color sets or texture sets for varied terrains.
- Statistic data for varied effects of terrain (impassability, etc)

To be addressed

- What are the precise lists of textures?
- What are the precise lists of terrain types?
- What are the precise lists of space volume objects?
- What is the exact method of terrain generation?

Walkthrough

The generation routines for flora and fauna share a number of characteristics in common. Both rely on a stock library of species and features, any mix of which may be combined. In addition, the creatures also require some things, AI, and statistics.

Flora and Fauna Generation

Overview

Planets in the *PO* universe are populated by algorithmically generated flora and fauna. This is done via snap-together deforming meshes, standard texture libraries, and scaled skeletal animations.

Vision and goals

The overall objective is to make every planet in the *PO* galaxy a unique entity, without extensive handcrafting of animations, meshes, and textures. This allows a real sense of exploration and discovery of new places. In addition, providing creatures on the ground permits a combat and adventuring mechanic to exist on the ground, which can serve as a major retention factor and as a means of attracting roleplaying gamers as opposed to space sim gamers. Since gaming on the planets will be substantially lower in bandwidth costs than gaming in space, providing game mechanics on planetary surfaces can improve the contribution margin.

- ☐ ... should appeal to sci fi fans (Star Wars, Star Trek, Wing Commander) with Internet connectivity
- ☐ ... should have a sense of discovery
- ☐ ... must have a presentation that blows people away at first glance
- ☐ ... must provide a way to co-opt existing communities
- ☐ ... should capture "sci fi fans who are Internet connected" whether they are gamers or not
- ☐ ... will stretch for the mass market
- ☐ ... must keep backend costs low
- ☐ ... should keep a small database size
- ☐ ... should have a low target bandwidth
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... should have a world that feels alive
- ☐ ... must support many different styles of play
- ☐ ... will satisfy Achiever-oriented players
- ☐ ... will satisfy Socializer-oriented players
- ☐ ... will satisfy Explorer-oriented players
- ☐ ... will satisfy Killer-oriented players
- ☐ ... should support character building and advancement
- ☐ ... should have secrets and lore to it
- ☐ ... will offer great cultures to experience
- ☐ ... should provide a sense of exploration

Walkthrough

The generation routines for flora and fauna share a number of characteristics in common. Both rely on a stock library of meshes and textures, any pair of which may be combined. In addition, the creatures also require animations, AI, and statistics.

Flora

- Standard mesh types
- Standard textures
- Decals

Fauna

- Standard skeletal animations
- Standard meshes
- Standard textures
- Decals
- Child objects
- Statistics
- AI

Systems design**Flora**

We will require meshes of the following types:

- Rock/crystalline
- Fernlike
- Wide blade leaf
- Shrub/bush
- Fungus/mushroom
- Vine
- Cacti (barrel)
- Cacti (cylinder)
- Coral
- Coniferous tree
- Deciduous tree
- Cypress/willow tree

All of these meshes (and there should be several variants of each) should be deformable and scaleable. If necessary, the meshes should consist of several stock pieces (eg, trunk, foliage) that can be mixed and matched.

These basic shapes can then be textured from among the following standard texture types:

- Glossy
- "Furred"
- Bark
- Spiny
- Oily
- Rocklike
- Etc

All of these textures are huable textures, perhaps procedurally generated, so that there is a wide range of possible appearances. The colors used are derived from the colors implicit in the planetary generation data (a dominant color derived from the "grass" color, and a subcolor derived from the "soil" color). These colors are actually just the

centers of spheres in RGB space, so that there is color variation even within the color set.

Decals that can be placed atop these textures include the following:

- Puff flowers
- Star shaped flowers
- Orchids
- Etc

Plant placement is done algorithmically also, based on the terrain textures at the given location on the map and the random seed for the planet. The placement routine is run on both client and server. This means that plants are considered static objects, and are immovable and unalterable.

Fauna

Creatures are generated from the same random seed that the planet is.

Appearance

All creatures are based around a few standard skeletal animations that are scaled to sizes determined randomly.

- Blob
- Floating blob
- Monopod (one leg)
- Biped humanoid
- Quadruped (long-legged)
- Quadruped (horizontal limbs, like lizards)
- Insectile (6 legs)
- Arachnid (8 legs)
- Centipede (many legged)
- Limbless (snake)

For each skeletal animation type, a set of stock meshes are made that are deformable via random factors for variety. Examples would include having both an annelid worm mesh and a snake mesh for the limbless animation.

A substantial amount of these animations may be obtainable from Ultima Online 2.

If possible, we would make use of the morphing technology to interpolate between extremes so that we can get unique versions of these meshes. Failing that, having pre-configured deformation points so that we can use random factors to adjust them would suffice.

Textures applied to these meshes should include the following. Any texture should be designed to go on any mesh:

- Fur
- Scales (reptilian)
- Scales (fish)
- Flesh
- Carapace/chitin
- Rock/crystalline
- Feathers

- Goo/slime

These textures would be huable so that colors can be arrived at via random factors also, drawing from the colors selected as the basic colors for the planet itself (see the document on *Planet spawning*).

In addition to these factors, articulated child objects using the same technology s we plan to use for articulating objects on ships can be employed to extend the range of unique details. Such objects may include lengthy feathers, eyeball stalks, tentacles, wings, horns, etc.

Certain creatures may be completely handcrafted, particularly ones that are terrestrial in origin. In particular, we can obtain all the livestock created for *Ultima Online 2*.

Statistics and combat capabilities

The statistical data behind creatures is drawn from a set of stock statistics from a simple level-based table. There are random factors, but “under the hood” most alien creature statistics are of just a few types—perhaps 8 internal levels, for example.

Artificial intelligence and special capabilities are drawn from individual stock script libraries. A given script might give a single capability, such as hurtful scream, goo spit, etc. Individual scripts can then be attached from the total pool of special behaviors to the specific creature type. The basic scripts that are required are merely one to initiate attack (aggressiveness), one to dictate when the creature flees (wimpy), and another for ranged attacks versus melee.

Interface design

There are no interfaces specific to generating flora and fauna.

Tasks

- Arrive at the final list of plant meshes
- Investigate the exact tech required for deformable meshes and morphing
- Write the relevant code.
- Write the code supporting plant placement
- Write the code for snap-together plants.
- Create the plant meshes
- Create the plant textures.
- Arrive at the tech required for deforming and morphing skeletal animations.
- Create the animation skeletons.
- Create the creature meshes, with attachment points for child objects
- Create the creature textures
- Create a library of child objects
- Create the internal statistical table for creature levels
- Write individual AI scripts

To be addressed

- What can we use from UO2?
- Deforming meshes, morphing, and so on are a big open issue. How feasible is this?
- What is the list of AI behaviors needed?
- How many creature levels?

Player Housing

Overview

Players in *Privateer Online* will be able to buy homes of various sizes and shapes and place them on planets. In fact, upon completion of their initial training mission, a basic house deed is one of the items given to the new player. They will have an array of tools for managing the building, including defining it as a public or private building, and securing possessions within. Public buildings will have an array of add-ons intended to convert them into places of business.

Vision and goals

Player housing gives a highly important sense of ownership in the world to players. This system will give players a piece of the virtual world to call their own, that they can struggle for and feel like they have earned. It serves as a barrier to exit, because they will be unable to take this piece of virtual real estate to any other game. Player housing has proven to be one of the top retention factors in *Ultima Online* and can be expected to serve a similar role in *Privateer Online*.

Goals

- ☐ To give players a sense of ownership in the virtual world.
- ☐ To allow players to run businesses out of their owned buildings.
- ☐ To provide players with a sense of achievement.
- ☐ To give socially oriented players a concrete space in which to socialize.
- ☐ To create a sub-game for players interested in economic aspects of the game.

Vision

- ☐ ... will offer robust support for players to form guilds and organizations
- ☐ ... will try to fulfill all of Amy Jo Kim's principles of community building
- ☐ ... must encourage and support a community
- ☐ ... must give tools to players to shape their own experiences
- ☐ ... should encourage a sense of player ownership in the game world
- ☐ ... should have ladders and leader boards for everything in the game, including social activities
- ☐ ... will offer players lots of rewards
- ☐ ... will satisfy Achiever-oriented players
- ☐ ... will satisfy Socializer-oriented players
- ☐ ... should have a world that feels alive
- ☐ ... should make players work to remain at their current level of achievement in the game
- ☐ ... will provide privileges to experienced players
- ☐ ... must provide a place to share experiences and build bonds with other players
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... should empower players to take action against those who are obnoxious

Walkthrough

The major components required for this system are:

- Artwork (models and textures) for an array of basic building types.
- A way of purchasing the building deed.

- An interface for placing the building.
- Placement code to allow building placement only in valid locations.
- Methods for customizing textures and signs and the like on the house.
- Interface for managing the parameters of the house.
- A friends system for the house.
- A concept of public and private ownership.
- A system for charging money for entry.
- Bandwidth conservation techniques for houses.
- Methods for policing houses (ban lists, ejection)
- House add-ons and business types based on add-ons.

Systems design

Building types

We will require buildings of the following types:

- TBD

These buildings should exist in both standard atmosphere versions and “domed” versions for planets with inhospitable climates.

An array of textures should exist that players can select for their buildings. These textures should be huable textures, so that “painting” the buildings is possible, but should include an array of surfacing materials, architectural feels, and styles. These materials should have varying armor ratings and strengths and weaknesses.

It is possible that we can do morphing on the houses in a manner similar to what we are hoping to do with avatars and creatures.

Building deeds

Building deeds are a player craftable item. They require a massive amount of raw materials, and a level of skill in architecture. The architect gets to specify the nature of the textures, special characteristics, and the type of building to be built by the deed. The quantity of raw materials required varies based on the type of building.

Once a deed is crafted, it can be used by anyone. Architects place them up for sale.

Building placement

Simply executing a “use” action on the deed brings up a “shadow” (translucent) version of the building to be placed. When the building is placed, the building appears stage by stage, “building” itself over several frames of animation (this may be cut, if time demands—in which case the building merely springs into existence).

There are rules for building placement. A house may not be closer than 10 meters to another house on any side. In addition, the land must be traversable by a character on foot (no severe cliffs, etc). The building automatically flattens the foundation to the highest point (or an arrived-at middle height) in the terrain that the building’s footprint occupies.

Building appearance

Buildings will use an array of textures determined by the materials used to create the deed. This way the textures used on the outside of the building will reflect the defensive capability of the building and give passersby an instant impression of the strength of the place.

Beyond that, owners will be able to “paint” these textures via control mechanisms in the house. Paint jobs can be changed at any time for a small fee. Special paint jobs (or

custom textures drawn from special libraries) may only be done by characters with the appropriate certification.

Signs are always hung on the outside of the house. Using this sign gives an indication of the name of the house, and the owner's name. As with any installation, you can use this sign to send a comlink message to the owner (but it does not give you access to the owner's greeting card—it is a one-time, unidirectional message).

Furnishings and add-ons are added via the menu, and are considered to intrinsically be part of the house. Each piece of furniture is an addition to the ongoing costs of the house, and is indestructible and cannot be looted.

Building status

Environmental activity such as earthquakes, microbial infestations, atmospheric corrosion, etc, can damage houses. In the event of ground combat, so can ground-based weaponry. When a building is excessively damaged, it loses functionality. Its terminal may cease to function, furniture may disappear, and its special appearances may go away, rendering it once again a "basic" object. Items within are not lost. Different materials should be vulnerable to different sorts of actions. Repairs must be effected manually. In safe space, naturally, houses cannot be damaged at all.

Houses consume energy to maintain their vital systems. The amount of energy consumed is based on how many items must be maintained in the house. The fewer items, the cheaper the cost will be on an ongoing basis. Monies are automatically deducted from the owner's bank account anytime anyone comes into range of the house.

Building access

Public houses can be entered freely at anytime by anyone, unless the individual is specifically excluded. The exclusion list can include up to 50 individuals or guilds.

Private houses are locked via a passcode. The passcode is an owner-determined password to get the door to open. After the first time someone enters the passcode, the door remembers the retinal scan for that player (eg, the house's object id and passcode are stored on the player) and the player may use the door freely from then on unless the passcode changes.

Controlling the house

The house is controlled via a terminal mounted in the inside wall next to the door. This terminal is always present and always active. Use of the terminal by a non-owner and non-friend of the house will result in a security message being given to the user stating that they did not pass the retinal exam, or some other fictional reason for the terminal not functioning.

The terminal offers multiple pages of menu, tabbed for convenience. See the section on interface for a breakdown of the terminal's individual screens.

You can also access this terminal via the web.

The capabilities contained within the house control systems are (broken out by subsection of the interface):

Status:

- The name of the owner of the house
- A field to change the name of the house

- The number of visits the house has had (if the house is declared public). This is tracked by counting the number of times people have crossed the threshold in the last seven real days.
- The current energy expenditures for the furnishings that the owner has placed in the house.
- If the house is public, a list of monies earned by the house since the last visit by the owner.
- The current condition of the house must be presented (eg, damaged, perfect condition, etc).

Access Control:

A friend of the house is defined as being the owner of the house, or being on the owner's explicit friends list. In addition, players have the ability to automatically consider all members of their company to be part of the friends list.

- A button to get a scrolling list of all people identified as "friends" of the house. Any house friend may access this list.
- A button to add a friend. There is a limit of 50 friends. Only the owner of the house may use this option. It brings up a targeting cursor (keyboard interfaces to targeting may also be used) which allows you to target the individual.
- A button to remove a friend will list all the friends of the house with checkboxes so you can choose one to remove.
 - Only the house owner may access this command.
 - A "remove now!" button will remove the selected friends from the list, and redisplay the list with the new friends list.
 - An OKAY button will return to the house sign menu without removing anyone.
- A button to ban a player works exactly like adding a friend does. Any house friend may use the command. Players who are banned may not cross the threshold of the house. An optional setting allows all players from that company to be assumed to be banned as well. Banned players are ejected to at least three meters from the door. You cannot ban a friend of the house. There is a limit of 50 bans. A message explaining what happened is given to the victim. Banning can also be activated via a spoken comlink command (easily set on a macro).
- A button to eject a player works similar to ban, but does not store the character data, so the player is merely ejected this one time. A script with a callback is attached to the player to mark them as temporarily banned.
- A button to list the bans is usable by any friend of the house.
- A button for lifting bans works identically to the "remove friend" functionality.

Commands:

- Transfer ownership button allows the house to be sold, intact, to another player. It retains all data (passcodes, ban lists, friends) and is to be used in the event of a friend taking over an existing establishment. It brings up a targeting cursor that lets you target the person to give the house to. Any fees must be privately arranged between the players. Only the owner may use this option. The target may not already own a house.
- Collapse house turns the house back into a deed. This destroys all data stored on the house, such as passcodes, furniture, ban and friends lists, etc. The ownership flag is removed from the owner, and the house may now be placed back on a commodities market for sale. This will bring up an "are you sure?" confirmation window. There is a fee for collapsing a house.
- Change passcodes is an option only usable by the owner. This option does work if the building is public, but since the doors are not lockable, it has no effect.

- A button to declare the building to be public can only be used by the owner. This activates a trigger on the door, which can automatically charge a small fee for crossing the threshold, allowing players to charge admission to the public house. Certain building add-ons can only be placed in public buildings. This option can only be used by the house owner. Walking out of a public house asks for a reputation assessment for the house.
- A field to set the admission fee is only active if the building is public.
- A button to declare the house private appears if the house is public. When changing from public to private, admission fees are disabled, and certain house addons may automatically disappear. The reputation gump no longer appears when you walk out.
- Options for changing house paint colors on a per-wall basis exist. These are done via palettes and color swatches. Options to change the house sign artwork outside also exist.
- Additional options may appear in the event of the house being part of a formal city. See the design document on Cities for more information.

Interface design

The interface for the house's main terminal should be a tabbed window with three tabs, "Status," "Access Control," and "Commands." Each of these tabs should provide a menu with access to the capabilities listed above. On all pages of the tabbed menu, the house name should remain visible. The interface is done using the interface description language.

In general, all modifiable or dynamic data presented in the interface should be in a different color or style than the static text, so it can be called out at a glance. Example:

This house has had 17083 visitors in the last week.

The interface for keying in the passcode should be a simple text field with a prompt.

Tasks

- An array of buildings must be generated.
- An array of textures for specific materials must be generated.
- An array of paint colors suitable for said textures must be generated.
- Armor ratings and environmental susceptibilities for different materials must be generated.
- An array of signs must be generated.
- An interface for keying in a passcode must be generated.
- An interface for the house sign displaying house name, and a means of sending an OWO mail to the owner.
- The door must check players to see if they have a passcode, if the house is private.
- The door must check for the presence of the player on the ban list.
- The door must check for the company membership of a player and compare it to the list of banned companies.
- The door must check for the presence of a temporary ejection variable on the player.
- The door must check if the building is public, and offer an interface for collecting any access fees the owner may have set up.
- An interface for the house terminal must be generated, to the specs outlined above.
- The door must track the number of visits to the house and wipe visits from the eighth day back.
- The door must track monies collected, and have an interface for choosing not to go in if you do not wish to pay. It must also notify you that you have had a debit incurred.
- The house must track its current damage.

- It must be visibly apparent that the house is damaged—textures must be generated.
- The house must have a process for tracking energy consumption based on house contents, and should have a process for automatically deducting from the owner's bank account.
- The house must track friends and their names and their companies, and banned folk, their names, and their companies.
- A script for temporary ejection must be written.
- An interface for house transfer must be written.
- Code for house building must be done.
- Code for house demolishing must be done.
- Passcode changing code must be written.
- A method of selecting colors from a palette must be written.
- Provide hooks to the website to use the house terminal.

To be addressed

- A precise list of building types
- Materials required for the deed
- Defense ratings vs different sorts of harm
- Maintenance costs
- Look and style
- Animation frames for the building process.
- Precise data structure for the friends list, ban list, etc.
- What house add-ons can only be run in public houses?
- Is a pop-up gump too obnoxious for charging a fee when you enter? What about the rep window when you walk out?
- Need interface layouts for the house terminal.
- Furnishings and "stuff."
- Can we do morphing?
- What is the fee for collapsing a house?

Walkthrough

The main components required for this system are:

- A system of recognizing when a colony is eligible to become a city.
- A voting system for selecting governors.
- A list of governor powers, as taxes and such placement of government facilities.
- An interface for governor powers.
- Special buildings that only cities can create or run.
- A definition of city-only building types, along with ongoing costs for upkeep.

Cities

Overview

When enough buildings (cf the document on *Player Housing*) are present on a planet and they organize sufficiently, they can form a city. Cities have a simple method of determining who the governor of the city is, and governors have a suite of special powers that enable them to control aspects of the city. The primary goal of city formation is to develop cities into safe areas that are policed by ConFed.

Vision and goals

Goals

- ☐ To provide advanced players with a sense of achievement.
- ☐ To give socially oriented players a concrete space in which to socialize.
- ☐ To help community-building players form in-game communities around spaces they themselves set up.
- ☐ To give an end-point to the struggles that town builders have, by allowing them to create towns that become "official" parts of the game.
- ☐ To create a sub-game for players interested in political aspects of the game.

Vision

- ☐ ... must encourage and support a community
- ☐ ... must give tools to players to shape their own experiences
- ☐ ... should encourage a sense of player ownership in the game world
- ☐ ... should have ladders and leader boards for everything in the game, including social activities
- ☐ ... will offer robust support for players to form guilds and organizations
- ☐ ... will try to fulfill all of Amy Jo Kim's principles of community building
- ☐ ... will satisfy Socializer-oriented players
- ☐ ... should have a world that feels alive
- ☐ ... should expose new strategies throughout the game to the player
- ☐ ... should provide multiple ladders of advancement so that players discover entirely new game mechanics, new ways to play the game, as they advance
- ☐ ... should empower players to take action against those who are obnoxious
- ☐ ... must have a high retention rate for subscribers

Walkthrough

The major components required for this system are:

- A means of recognizing when a colony is eligible to become a city.
- A voting system for selecting governors.
- A list of governor powers as taxes and such placement of government facilities.
- An interface for governor powers.
- Special facilities that only cities can create or run.
- A definition of city-only building types, along with ongoing costs for same.

Systems design

City formation

As houses are placed on a planet, the planet's data structure automatically tallies the number of active houses on the planet. When that number crosses a certain threshold, the terminals in each house gain a new menu tab, for voting for a governor. As soon as 50% of the inhabitants of a city have thrown their vote towards a given single individual, the city is formally created (thereafter, governors are selected by simple majority).

Selecting governors

The house terminal has a menu that is a list of all players with houses on the planet (citizens of the planet). Beside each is a checkbox. Players can, at any time, change who they are "voting" for. When this occurs, the city tallies up the votes and the player with largest number of votes is named the new governor. In the event of a tie, the current governor remains in power (even if the tie does not involve the current governor).

When the governor changes, the news items on the city tab on all local house terminals are automatically updated, and an email goes out to all citizens. A news story is also automatically generated for the news system.

Governor powers

The governor gets a new tab on his house menu for governor powers. These powers are:

Citizenry

A button to bring up a list of the players who are citizens. The list also displays whether or not the player was able to make their last tax payment.

Taxation level

A simple +/- button allows the governor to set a tax rate for living in the city. Monies are automatically deducted from the bank account of the various players at a given interval if the player has so chosen. Players can choose at any time to turn off making the payment. If the bank account cannot accommodate the payment, the player is notified via OWO Mail, and the citizenry listing above is updated.

City funds

A simple value displays how much money is in the city's bank account. A governor can transfer cash to the account of any player on the planet. The governor gets to enter a text string giving a reason why this was done. Each time this is done, the city news on house terminals is updated with the text string, the recipient, and the amount.

Forcible buyback

Governors may expend a significant amount of money from the city treasury to forcibly buy back a building run by another player on the planet.

Build facility

Governors can build a number of installations that are specific to cities. The funds required for building these come out of the city treasury, and each type of item also has an ongoing cost associated with it. These can include:

- Roads. This replaces the terrain texture with standard pavement. Some roads may be "domed" for inhospitable atmospheres.
- Landmarks. These place fountains, statues, etc. There should be a selection of these, dependent on artist time.

- Bank. This is a location that allows players to deposit cash.
- Cloning facility. This allows players to clone themselves.
- ConFed Police. This facility, when present, automatically makes the planet a safe zone where no PvP is allowed.
- Commodities market and stock market.
- Arenas for sports/events (an option is a building within which a PvP zone is re-established despite the city's safe status).
- Empty general purpose buildings.
- Defenses against creatures (Command & Conquer obelisk style, or force fields).
- Atmosphere generators or domes (to cover parks or roads).

Interface design

Governor selection

When the criteria for being a city are met, the standard house terminal menu gains a new tab, for city news and voting for a governor. The news is generated automatically by the actions of the governor and is a simple chronological listing going back ten items.

The governor selection menu should be a simple menu listing the names of each citizen of the planet (defined as an individual who owns a house there—not merely a business—with a radio button beside them.

Governor powers

Lists of citizens should use whatever standard interface has been settled on for presenting lists.

Taxation should use a field that can be typed into, or buttons to increase and decrease the value.

The text string prompt for issuing city funds should be a standard text query prompt.

Forcible buyback should ideally present a zoomed out view of the planetary map, so that the governor can select which building to buy back. Failing that, a mere list of buildings with names and owners and a radio button would be sufficient. This must be accompanied by a confirm "Are you sure" window.

Ideally, facility building should use a zoomed out view with a palette of building choices, similar to *SimCity 3000*'s interface. Failing that, the terminal can merely generate deeds and the exact same system as is used for housing can be used.

The capability to zoom out and see the city as a whole would be a very nice interface perk. This may be possible for all players anyway as part of our standard camera, but must still be investigated.

Tasks

- A planet's data structure must track the number of houses built on it, and update it when houses are added and destroyed.
- There must be a triggering event to add the city menu tab to all house menus on the planet. This is probably via a query of the planet data structure at the time that the menu is invoked.
- There must be a flag differentiating formally founded cities from villages that meet the criteria to try to become cities.
- A system for tallying governor votes must be developed.
- There must be a messaging system to send city news to the houses on the planet.

- There must be a city menu tab.
- There must be a governor powers menu tab.
- A system for deducting taxes from players must exist. We must track the last payment status.
- A system for transferring funds out of the city treasury must exist.
- A system and interface for forcible buyback must exist.
- A specific list of installations must be developed.
- Artwork and variable textures for the installations must be developed.
- The code for banks, stock markets, commodities markets, cloning, safe zones, non-oxygen atmospheres, and other such dependencies must be established and complete. Refer to the appropriate design documents for each of these.
- Automatic AI for defenses must be written.
- A system for facility placement must exist.

To be addressed

- How many buildings are required for a village to become a city?
- Jeremy Gaffney suggests income tax and also customs or tariffs or sales tax.
- How is the cost calculated for placing and maintaining city installations?
- What is the art list for the City landmarks?
- Need to narrow down and refine the list of installations buildable by governors.
- To what extent can we get zoom-out capability for these interfaces?
- What about cities decaying or going away?

Replace this page
with Supporting
Systems tab

Cameras, Controls, and Interface

Overview

Vision and goals

Goals

- 1. Provide an intuitive user interface for ground subsystems, camera control, and object interaction.
- 2. Provide an intuitive user interface for ship subsystems.
- 3. Provide flexible controls for the ground camera, ship camera and cockpit system controls.

Vision

**Replace this page
with Supporting
Systems tab**

- 1. Ground Controls
- 2. Ground Camera
- 3. Flight Controls
- 4. Flight Camera
- 5. Related Controls

System Design

Ground Controls

Mouse Controls

Pivot Camera



To pivot the camera, click and hold the right mouse button while the cursor is in a non-object area of the view window and move the mouse left or right.

Tilt Camera



To tilt the camera, click and hold the right mouse button while the cursor is in a non-object area of the view window and move the mouse up or down.

Cameras, Controls, and Interface

Overview

Vision and goals

Goals

- ☐ Provide an intuitive user interface for ground movement, camera control, and object interaction.
- ☐ Provide an intuitive user interface for ship movement.
- ☐ Provide flexible controls for the ground camera, chase camera and cockpit camera controls.

Vision

- ☐ ... must have an intuitive user interface
- ☐ ... must have a presentation that blows people away at first glance
- ☐ ... must support as low end a machine as possible, to maximize the audience
- ☐ ... must have big visual payoffs

Walkthrough

This document will step through:

- ☐ Ground Controls
- ☐ Ground Camera
- ☐ Flight Controls
- ☐ Flight Camera
- ☐ Related Controls

Systems Design

Ground Controls

Mouse Controls

Pivot Camera



To pivot the camera, click and hold the right mouse button while the cursor is in a non-object area of the view window and move the mouse left or right.



Tilt Camera

To Tilt the camera, click and hold the right mouse button while the cursor is in a non-object area of the view window and move the mouse up or down.

Zoom Camera

Some mouse input devices include a third (middle) mouse button or a *mouse wheel*. To Zoom the camera view in and out: Click and hold the middle mouse button while the cursor is in a non-object area of the view window and move the mouse up or down. If a mouse wheel is available, rotate the mouse wheel forward and backward.

**Movement****Pathfinding**

To set a movement destination, *click* the left mouse button on an object or a terrain feature. (Terminates waypoint movement mode.)

Continuous

To continuously update the movement destination, *click and hold* the left mouse button on an object or a terrain feature. (Terminates waypoint movement mode.)

Waypoints

Press the CTRL key while *clicking* the left mouse button on an object or a terrain feature to establish movement waypoints for the character to follow. Translucent green waypoint lines are drawn between each waypoint. The character will immediately begin moving towards the first waypoint and will stop only when it has reached the last waypoint, or by setting a new Pathfinding destination. Waypoints cannot be established across permanent obstacles, and the current waypoint line will change color to red to indicate a waypoint cannot be set.

**Quick Object Description**

To see an object name, move the mouse cursor over the item and hover for a moment. A help balloon will appear or help text will be displayed in the status area (if shown.) The hover time is a configurable option. Ping times are a factor and it may take up to 1/2 second to retrieve and display the information.

**Object Interaction****Object Highlighting**

Objects that can be used or provide information will highlight when the mouse cursor moves over them. (The highlighting will be provided on the client side and should have no bandwidth constraints.)

Object Use

To Use an object within range of your character, *double click* the left mouse button while the cursor is over the object.

Object Interface

To interact with an object within range of your character, *click and hold* the left mouse button while the cursor is over the object. After a short (configurable) period, the Item Interaction Interface is activated over the object. The overall look of this interface is TBD, but will include functions for Use, Identify, Pickup, Drop, etc. The option list will undoubtedly change during ground combat. One of these options may be used to activate another character's Avatar interface to allow players to identify or steal from them.

Keyboard Controls

The game will have ground controls that can be bound to keys. Some of the required keyboard controls are defined below (without default key bindings, which are still TBD.)

Interface Commands

- Open PDA Interface
 - Open Avatar Interface
 - Open Overhead Map
 - Open Inventory
- Targeting Commands
 - Combat Toggle
 - Target Last
 - Target Next
 - Target Self
 - Target Cancel
- Movement Commands
 - Move Forward
 - Move Backward
 - Move Left
 - Move Right
 - Run Toggle
- Camera Commands
 - Zoom Camera In
 - Zoom Camera Out
 - Toggle Overhead Camera
 - Toggle Camera Mode
 - Toggle Look-spring
 - Pivot Camera Clockwise
 - Pivot Camera Counterclockwise
 - Tilt Camera Up
 - Tilt Camera Down
- Emote Commands
 - Bow
 - Nod
 - Headshake
 - Wave
 - And many more...
- Help

NOTE: All alphanumeric, SHIFT, BACKSPACE, tilde, bracket, minus, plus, slash, backslash, colon, quote, comma, period, and ENTER keys are defined as communication keys and cannot be bound without a CTRL or ALT key.

Ground Camera

The purpose of the Ground Camera is to provide the player with a flexible, third person view of their character and immediate surroundings. While a limited-tilt, isometric overhead view may not provide as much visual "bang for the buck" as a first person camera, it does have several advantages that meet technical and visual criteria set for the game.

First and foremost, limiting the view from displaying the horizon requires less bandwidth due to the reduced number of polygonal elements visible in the view frustum¹.

¹ The viewing frustum defines the volume of a virtual world that is visible within the 2D representation of the world.

The 3D isometric overhead view also offers the chance to provide a more strategic interface by presenting terrain, obstacles, and dangers outside the limited "peripheral vision" of a first-person view.

Ground Camera Features

- ☐ Tilt
- ☐ Pivot
- ☐ Zoom



Normal Camera (approx. 30°)
Rotated to face character

Tilted Camera (<30°)



Third / Overhead Camera (90°)

Camera Functioning

The camera can be pointed around the character and controlled in a variety of ways.

Pivot Options

The camera can pivot in two different ways, either pivoting when the character turns, or only pivoting manually. These modes are called **Follow** and **Free**.

Follow Mode

The world will pivot around the camera to keep the player facing the top of the screen. This is the default mode.

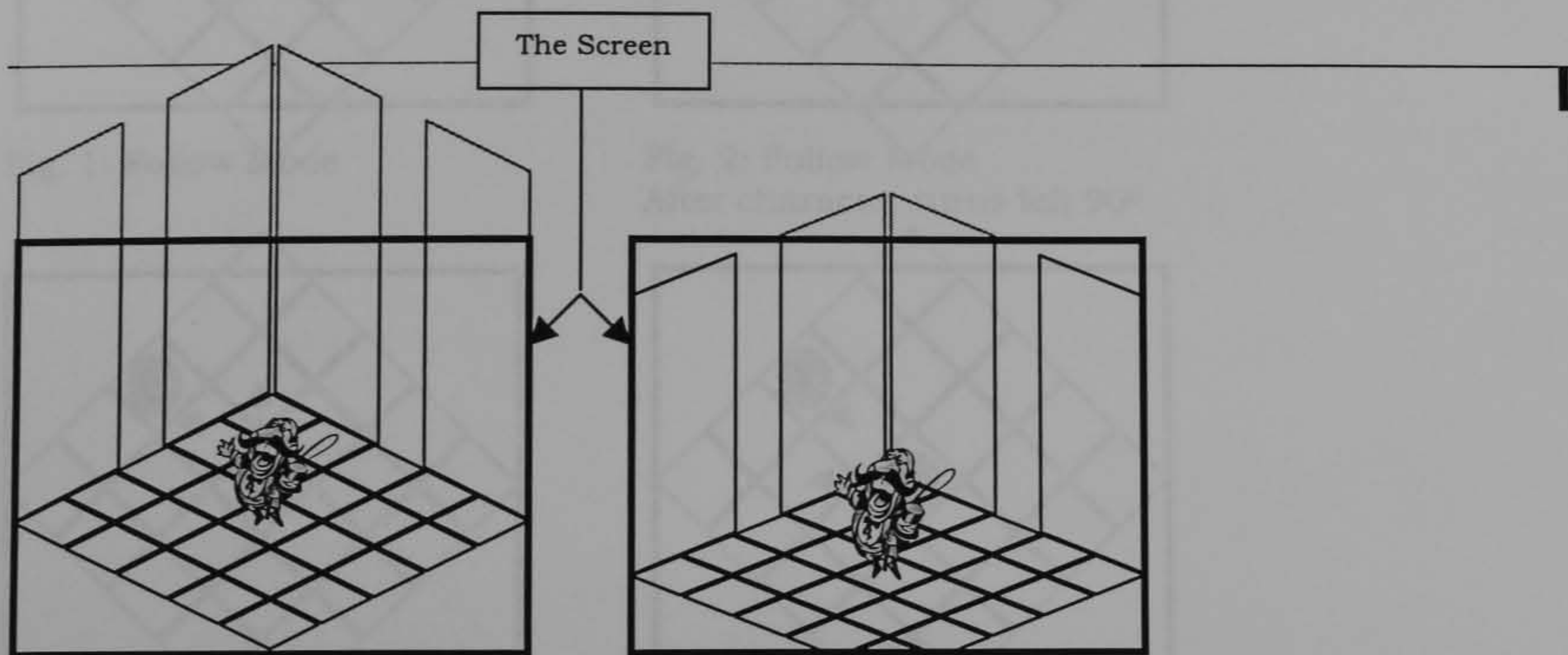
Free Mode

The character turns freely and the camera will pivot only when manually operated with a keyboard or mouse action.

Camera Tilting

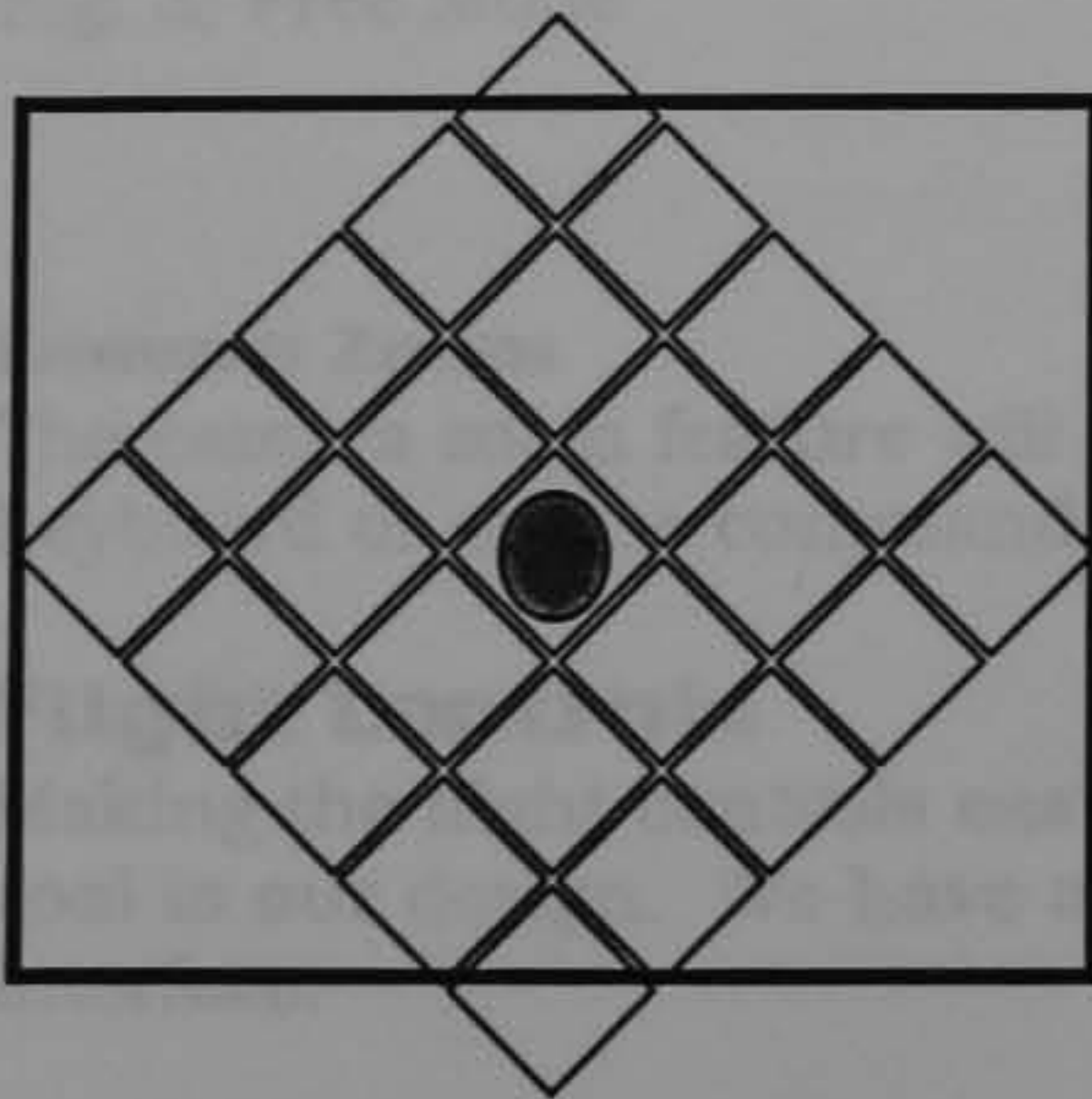
Tilting the camera allows the player to see more of their character's surroundings in the direction the camera is facing. This feature can be activated by either a keyboard or mouse action. The pivot and tilting features work together when the mouse is used.

In the image below, the camera is position in front of the character rather than behind. (Note that the perspective change that would occur from tilting the camera is not accurately represented below.)



Normal Camera (approx. 30°)
(Rotated to face character)

Tilted Camera (<30°)



Tilted / Overhead Camera (90°)

Camera Pivoting

The camera can be pivoted around the character and controlled in a variety of ways.

Pivot Options

The camera can pivot in two different ways, either pivoting when the character turns, or only pivoting manually. These modes are called Follow and Free.

Follow Mode

The world will pivot around the camera to keep the player facing the top of the screen. *This is the default mode.*

Free Mode

The character turns freely and the camera will pivot only when manually operated with a keyboard or mouse action.

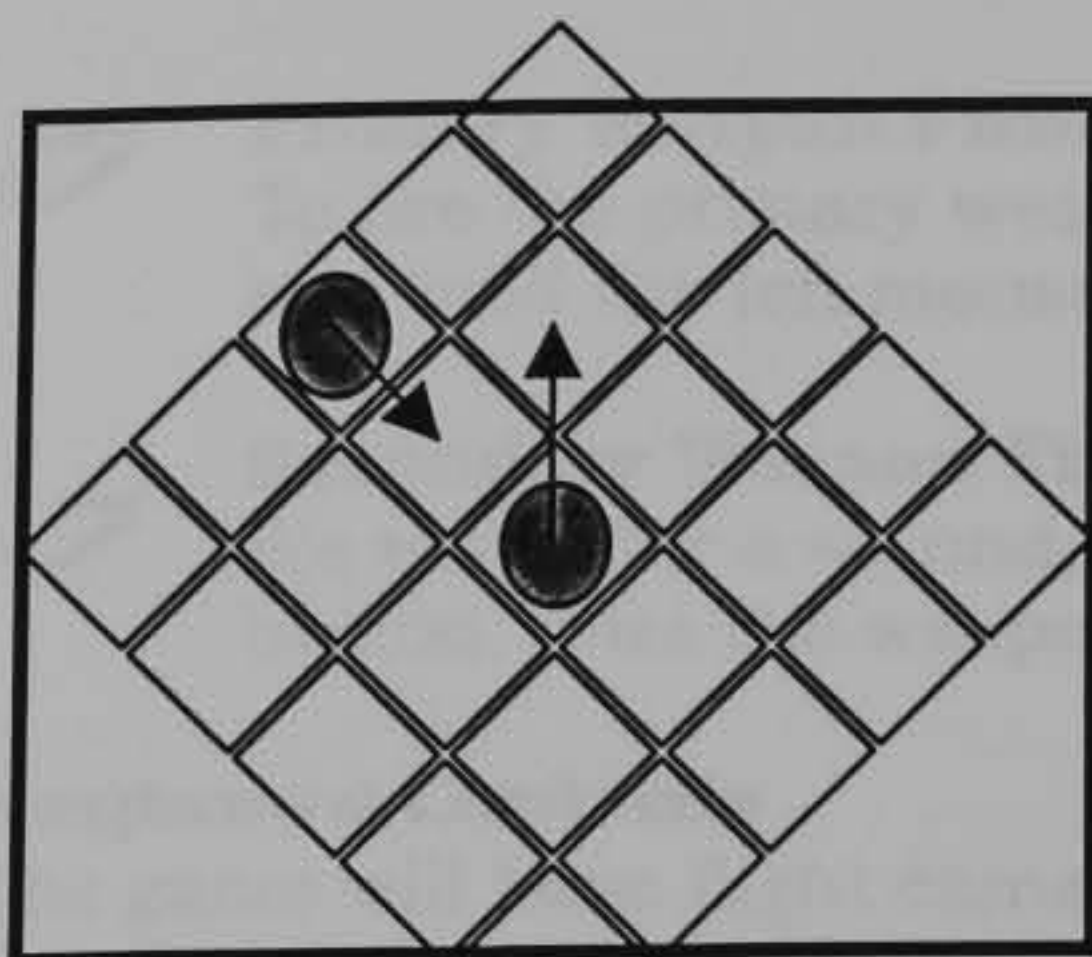


Fig. 1: Follow Mode

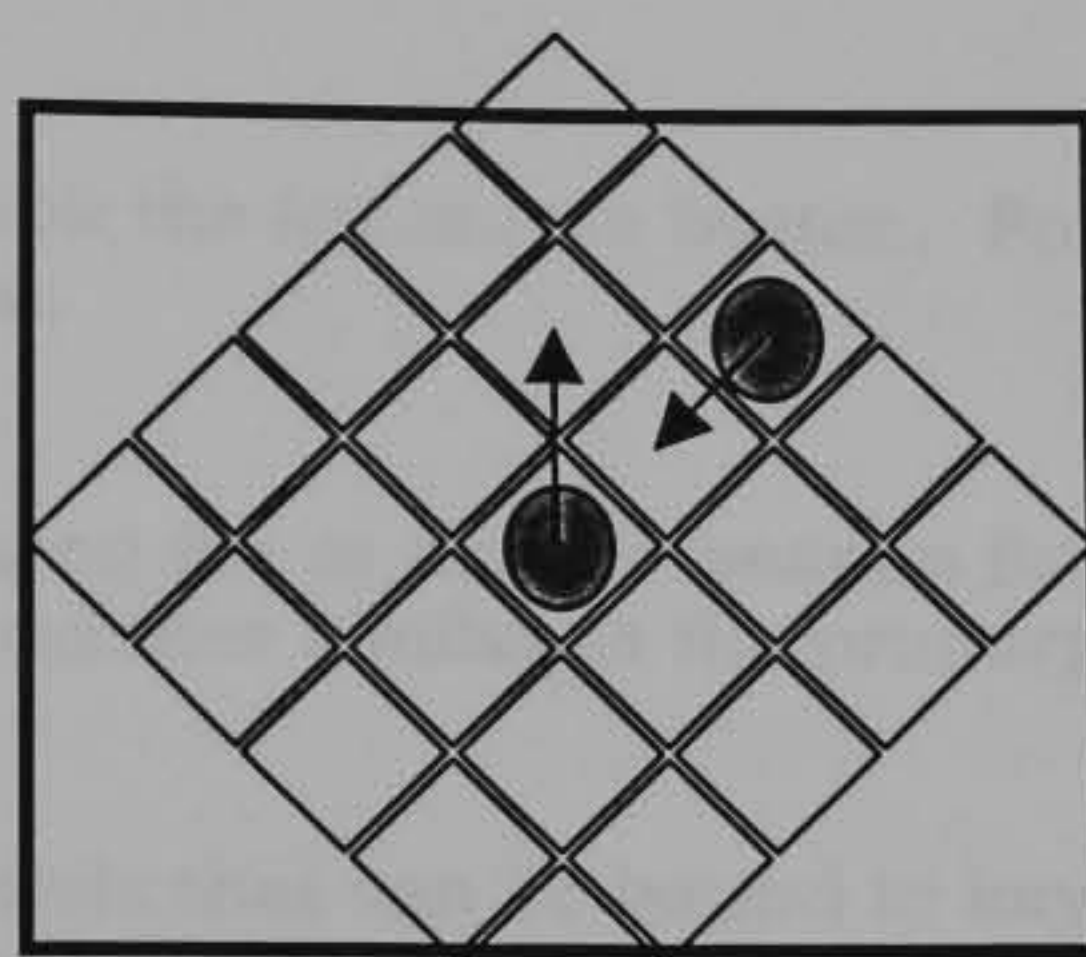


Fig. 2: Follow Mode
After character turns left 90°

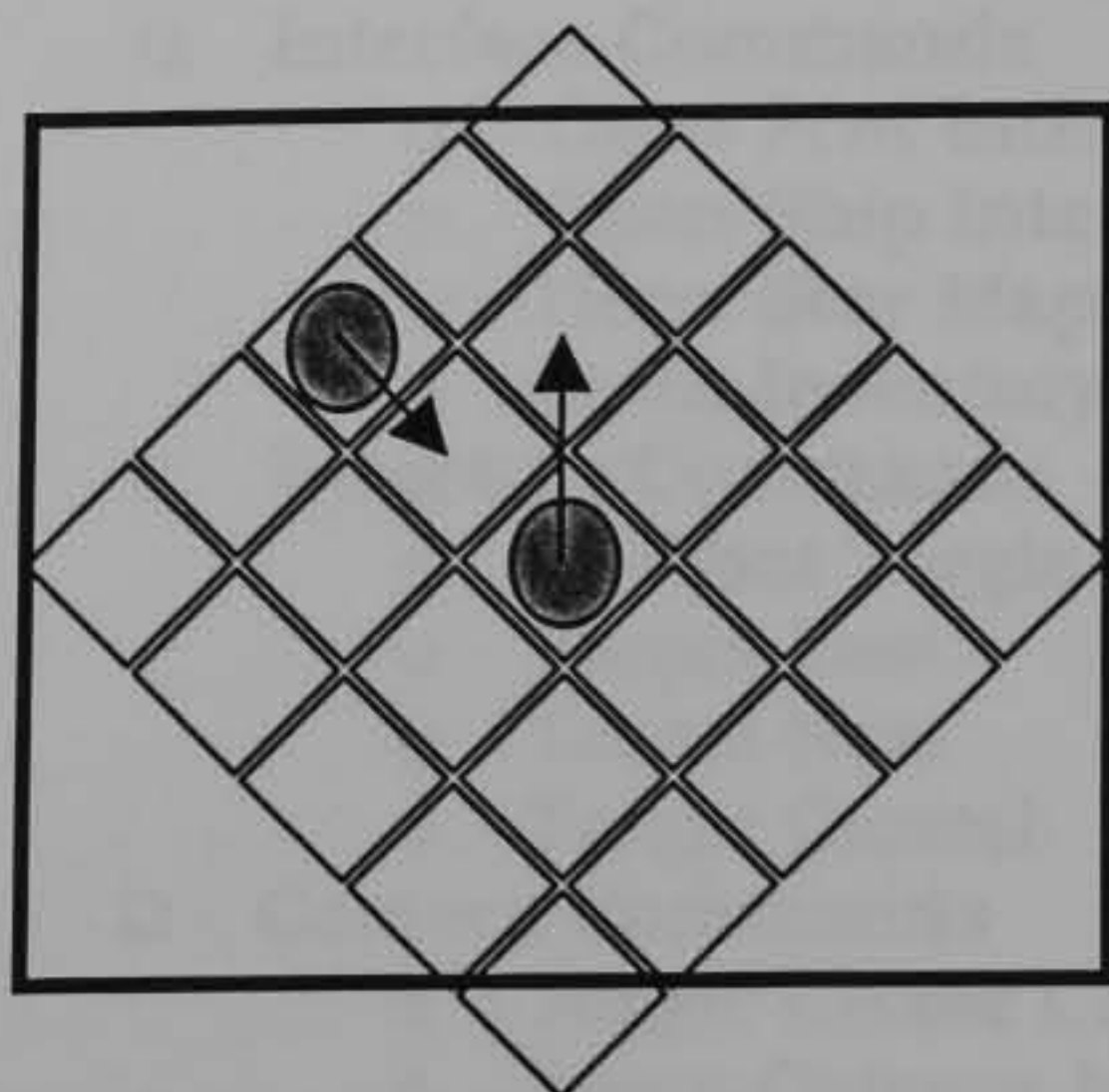


Fig. 3: Free Mode

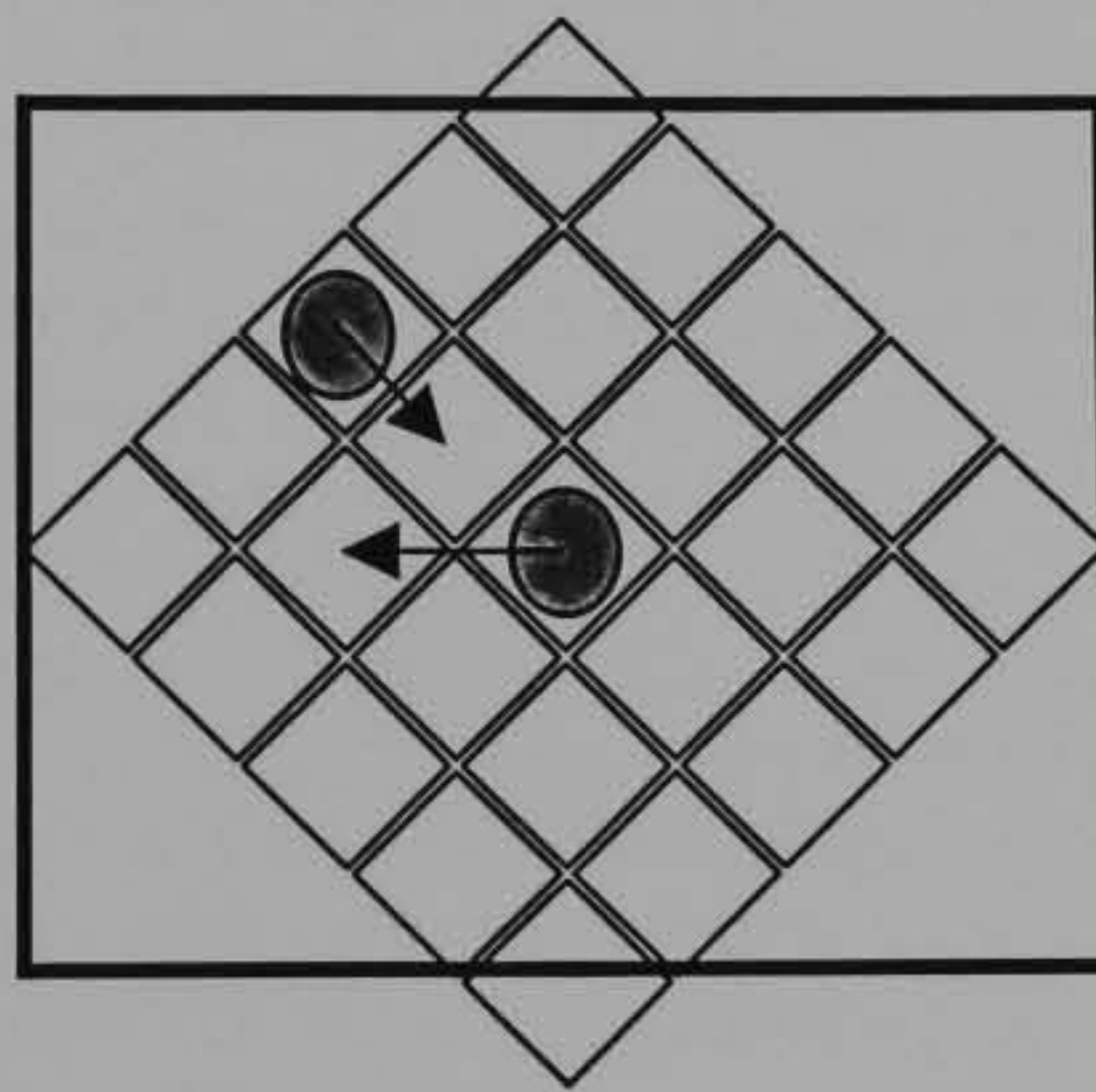


Fig. 4: Free Mode
After character turns left 90°

Camera Zoom

The camera zoom feature will have a limited range, and can be activated by either keyboard or mouse commands.

Flight Controls

Making the flight controls easily accessible to a large player market was an overriding goal in our design. We have attempted to do so by relying on a simple mouse driven interface.

Mouse Controls

The design of this feature is still TBD, but the idea is to make the interface similar to the camera pivoting of the ground camera.



Yaw

To bank (turn) your ship left or right, move the mouse left or right while thrust is enabled.



Pitch

To pitch your ship up or down, move the mouse up or down while thrust is enabled. The direction of movement will be affected by an "Invert Mouse" option.



Cockpit Camera

One feature missing from most flight-simulations is the ability to look around the virtual cockpit. This is something we plan on implementing through a mouse button or keyboard press with mouse motion.

**Primary Weapon Firing**

To fire the primary weapon, click the left mouse button. For repeat fire, click and hold the left mouse button.

**Secondary Weapon Firing**

We may offer a secondary weapon fire or missile weapon fire via the right mouse button. Fire the weapon in a manner similar to the primary weapon.

Keyboard Controls

The game will have flight camera controls that can be bound to keys. Some of the required keyboard controls are defined below (without default key bindings, which are still TBD.)

- ☐ Interface Commands
 - ☐ Open PDA Interface
 - ☐ Open Ship Interface
 - ☐ Open Star Map
 - ☐ Open Inventory
- ☐ Targeting Commands
 - ☐ Combat Toggle
 - ☐ Target Last
 - ☐ Target Next
 - ☐ Target Cancel
- ☐ Camera Commands
 - ☐ Toggle Chase Camera
 - ☐ Zoom Camera In
 - ☐ Zoom Camera Out
 - ☐ Pivot Camera Clockwise
 - ☐ Pivot Camera Counterclockwise
- ☐ Help

Flight Camera

By default, the Flight Camera displays the interior of the ship cockpit. The camera can be toggled to display the chase camera. The chase camera can be pivoted around the ship.

To Be Addressed**Ground Camera**

- ☐ Avatar browsing
- ☐ Stealing
- ☐ Item and money transfer
- ☐ Complete list of default keyboard bindings
- ☐ Interface for binding keys to commands
- ☐ Autowalk option to use object.
- ☐ Integrate with Ground Combat doc.

Ship Camera

- ☐ Further define camera movement within the cockpit.
- ☐ Integrate with Ship Combat doc.

Backstory

"Time passes without regard for the whimsy of your spirit. " --Sivar

In a galaxy torn apart by war, rebuilding requires an understanding of what has come before, what is now, and where you fit within its fragile structure. Your place in the universe resides with the Confederation.

From your home world of Earth the ennobled Confederation spread throughout the galaxy after the discovery of Jump travel; a faster than light pipeline that lead to hundreds of systems full of opportunity and secreted dangers.

Many of these systems were empty. Some were not. Confed made contact with the benign Firekkans, a peaceful race of avian beings, and the Double Helix, an enigmatic race that preferred to be left alone, and had the means to shroud their home systems from discovery.

The Confederation found its first interstellar enemy in the form of the Kilrathi. A ruthless race of expansionists ruled by a draconian government in service to a God of war, the Kilrathi did not waste time with diplomatic endeavors. A war started in 2634 that threatened to eradicate the Human race. Fortunately a hero rose from the ranks of the Confederation military to end the conflict with a decisive strike against the Kilrathi home world of Kilrah.

What followed was a time of intermittent peace interrupted by internal power struggles, then invasion by a new race hinted at by Kilrathi legend and known only as Nephilim. At the third and final encounter with the Nephilim the Confederation lost 40% of their standing fleet in an as of yet unexplained conflagration. The Nephilim, or Bugs as pilots called them, weren't seen again, however no one believes they are truly free of them.

A Kilrathi warlord by the name of Ra'Khaj took the opportunity afforded by Confed's bug problem to begin raising an army capable of re-igniting a flame snuffed out with the destruction of their home world. As Confed attempted to recover from the final Nephilim conflict, Ra'Khaj struck hard as the Kilrathi pierced a hole straight through Confederation space to Sol.

Neither the Kilrathi nor Confed could claim victory at the end of the struggle that ensued. Both sides beat each other to submission; retreating back through the razed remains of a front line a sector wide. This section is now referred to as Dead Space, and only the hardest merc, and pirate raiders thinly populate its charred systems.

It has been a hundred and fifty years since the first Human/Kilrathi war. It's only been forty since the second. The face of known space has been irrevocably changed. Alien races once benevolent have been rendered isolationist xenophobes after feeling the crush of the Confederation and Kilrathi. In particular, the Firekkans isolated themselves from the universe, developing in secret and casting off their peaceful traditions for a more expansionist regime called the NFO (New Firekkan Order).

As the rebuilding began, Confed and its internal corporations called upon the citizens to pioneer the frontier sectors deep in Confederation territory. They needed prospectors, community builders, terraformers, mercenary couriers, transport captains, and any other skilled citizen with the spirit to succeed. The reward? To be one of few to take part in the building of a new Confederation infrastructure, replete with rich resources to

exploit, new trade routes, treasures waiting to be discovered, and fame but a jump away. To be a pilot on the frontier, or the founder of a new city in a system as yet unknown, these are lures drawing the best of Confed's citizenry, and the worst of opportunists.

The risks? Pirates, Kilrathi and Firekkan Raiders, unknown life forms planet side or space side, Mercs of all flavor, harsh planetary conditions, terrestrial creatures, as well as countless celestial dangers and unknowns make the opportunity a challenge worthy of you the Privateer.

Vision and goals

This module is intended to serve as a general introduction and for design purposes only. It will serve as the backbone for the design team as well as a means for players to stay on track.

Walkthrough

System design

The design team has been working on the design of the game for a long time.

Interface design

The interface design team has been working on the design of the game for a long time. The interface design team has been working on the design of the game for a long time.

Hardware

There are three main hardware issues to the upper right corner. The first hardware issue is the CPU.

The design of the dialog box is a complex task. Messages are grouped into folders. There can be folders of folders, with one and only one open at any given time. Each folder has an icon to its left, and a list of messages. For example, "Messages for Character of Character". Every folder has a list of messages. If there are no messages for a particular character, the folder for that character simply does not appear. If any messages are addressed to the user, as opposed to one of the characters, then the first folder and contains those messages. Characters folders are distinguished by character name. Double-clicking on a folder shows the open folder and opens the selected message. When this dialog is first opened, the folder that corresponds to the user's current character is open.

Each message contained in the open folder is listed on a single line, showing the subject, the subject, and the date and time sent. Messages are sorted by date/time. If there is insufficient room by the list for the subject or subject, the text will automatically wrap to a smaller font. If the "unread messages" box is not too big, as many characters as will fit are displayed, followed by an ellipsis (...). Clicking on a message will open the appropriate buttons at the bottom of the dialog. Double-clicking on a message opens a Read Message dialog for that message.

There is a row of buttons at the bottom of the dialog: New, Reply, Forward, Delete, and Delete All.

The New button is always enabled and opens the New Message dialog.

OWO Mail

Overview

We will make use of the OWO Mail facilities provided by the Tools and Technology group.

Vision and goals

This system is intended to serve as a generic communication tool for delayed peer-to-peer messaging within Privateer Online. It will serve as the backbone for our GM support as well as a means for players to stay in touch.

Walkthrough

Systems design

The backend system is the standard OWO Mail system that exists at the time of launch.

Interface design

Mailbag

New email notification occurs on your PDA via a simple text message and a played sound. Both the message and the sound are configurable via options.

Messages

There are close and minimize buttons in the upper-right corner. The box minimizes to become the PDA.

The center of the dialog box is a scrolling list of messages. Messages are grouped into folders. There can be dozens of folders, with one and only one open at any given time. Each folder has an icon to its left, then a line of text reading (for example) "4 messages for Albatross of Andromeda." Every folder must have one or more messages; if there are no messages for a particular character, the folder for that character simply does not appear. If any messages are addressed to the user, as opposed to one of his characters, then the first folder listed contains those messages. Otherwise, folders are alphabetized by character name. Double-clicking on a folder closes the open folder and opens the clicked-on folder. When this dialog is first opened, the folder that corresponds to the user's current character is open.

Each message contained in the open folder is listed on a single line, stating the sender, the subject, and the date and time sent. Messages are sorted by date/time. If there is insufficient room in the list for the sender or subject, the code will automatically switch to a smaller font. If the "smallest readable" font is still too big, as many characters as will fit are displayed, followed by an ellipsis ("..."). Clicking on a message selects it, enabling the appropriate buttons at the bottom of the dialog. Double-clicking on a message opens a Read Message dialog for that message.

There is a row of buttons at the bottom of the dialog: New, Reply, Forward, Revoke, Delete and Names.

- The New button is always enabled, and opens the New Message dialog.

- Reply is grayed out unless a message is selected; it opens the Reply Message dialog.
- Forward is grayed out unless a message is selected; it opens the Forward Message dialog.
- Revoke is grayed out unless a message from another user is selected; it opens the Confirm Revoke dialog.
- Delete is grayed out unless a message is selected; it opens the Confirm Delete Message dialog.
- The Names button is always enabled, and opens the Names dialog.

New Message

There are close and minimize buttons in the upper-right corner. If there is any text in the Message control, hitting the close button brings up the Confirm Discard dialog.

There are four main interface elements: the To line, the From line, the Subject line and the Message control.

- The To line displays the addressee of the message. Clicking on this line opens the Pick Names dialog. When the message has not been addressed yet, this line reads "(Click here to set the addressee)" or some such.
- The From line displays the sender of the message. If the message is addressed to another user, the sender is the user's current character name. If the message is addressed to OSI—customer service, player relations, what have you—the sender is the user's real name. This line is set by the software and cannot be edited.
- The Subject line is a simple single-line edit control containing the subject of the message. The user is free to type whatever he wants here.
- The Message control is a multi-line edit control in which the user types the body of the message.

At the bottom of the dialog are two buttons, Send and Cancel.

- The Send button is grayed out unless an addressee has been selected; it closes the dialog and transmits the message to the server.
- The Cancel button brings up the Confirm Discard dialog if there is any text in the Message control; otherwise, it simply closes the dialog.

Read Message

The Read Message dialog is based off the New Message dialog, with the following changes:

- There is no minimize button, just a close button. The close button simply closes the dialog box.
- The To and Subject lines cannot be edited, nor can the Message control.
- There is an additional (read-only) line, showing the date and time the message was sent.
- There are four buttons at the bottom of the box: Reply, Forward, Delete and Revoke.
- The Reply button opens the Reply Message dialog.

- The Forward button opens the Forward Message dialog.
- The Delete button opens the Confirm Delete Message dialog.
- The Revoke button opens the Confirm Revoke dialog.

Reply Message

The Reply Message dialog is based off the New Message dialog, with the following changes:

- When the dialog opens, the To line is automatically filled in with the address of the sender. The To line cannot be edited.
- The Subject line initially reads "Re:", followed by the original subject line. Additional instances of "Re:" are discarded. The Subject line can be edited.
- Immediately above the Message control is a check box reading "Include original message". If this box is checked, the Message control displays a horizontal line, followed by the text of the original message. This control uses HTML, so "behind the scenes" what this control contains is <HR><BQ>, the original text and a closing </BQ>. This original message text cannot be edited, and if present is always attached to the end of the new message.

Forward Message

The Forward Message dialog is based off the Reply Message dialog, with the following changes:

- The To line works the same as in the New Message dialog.
- The Subject line initially reads "Fwd:", followed by the original subject line. Additional instances of "Fwd:" are discarded. The Subject line can be edited.
- There is no "Include original message" check box, but the Message control functions as if that box were checked.

System Message

The System Message dialog is opened in response to various script commands. It is most often used to address new messages to customer service. It is based off the New Message dialog, with the following changes:

- The script can specify the title for the dialog; the default title is "New Message."
- The addressee is specified by the script, so the To line is read-only.
- The subject is specified by the script, so the Subject line is read-only.
- The script can specify default text for the Message control. It can also specify whether the Message control is editable or read-only.
- The script can specify data to send along with the message. This happens invisibly to the user, so it has no effect on the interface; it is mentioned here just for programmer reference.
- Closing this box only brings up the Confirm Discard message if the user actually typed something into the Message control.

Web Browse

The Web Browse dialog is very simple. It contains the title, a close button, a Back button and a Data control. The title of the window is the title of the web page being browsed. The close and Back buttons do exactly what you think they do; if there is no previous page to go back to, the Back button is grayed out. The Data control uses the same code as the Message control from the Read Message dialog.

Web browsing is only done in response to script commands, so no other web interface features are required.

Names

The Names dialog reuses code from the Messages dialog to list the contents of the user's address book. It consists of the title ("Names"), a close button, a scrolling list of names, and four buttons (Send, Revoke, Delete and Edit).

- As with the Messages dialog, the list contains folders. There are just two folders, one labeled "System Addresses," the other "Character Addresses." As with the Messages dialog, only one folder can be open at a time. The System Addresses folder contains addresses for customer service, player relations, etc. The Character Addresses folder contains all the addresses the user has received. Within each folder, the names are listed alphabetically. When the dialog is opened, the Character Addresses folder is open, but no name is selected.
- Each name in the list is displayed on a separate line, in the form "Aileron of Milky Way (met 3/18/99)". A name can be clicked to select it, enabling the buttons. A name in the Character Addresses folder can be double-clicked to open the Edit Description dialog.
- The Send button, grayed out unless a name is selected, opens the New Message dialog, with the To line already set to the selected name. The To line can be edited normally.
- The Revoke button opens the Confirm Revoke dialog. This button is grayed out unless a name from the Character Addresses folder is selected.
- The Delete button opens the Confirm Delete Name dialog. This button is grayed out unless a name from the Character Addresses folder is selected.
- The Edit button opens the Edit Description dialog. This button is grayed out unless a name from the Character Addresses folder is selected.

Pick Names

The Pick Names dialog is a streamlined version of the Names dialog, used to select a name for the To line of a new message. It differs from the Names dialog as follows:

- When opened, the name currently listed on the To line is selected. If no name is listed on the To line, then the first name in the Character Addresses folder is selected.
- There are just two buttons, OK and Cancel. OK is grayed out unless a name is selected; it closes the dialog and fills in the To line with the selected name. The Cancel button closes the dialog without changing the To line.

Edit Description

The Edit Description dialog lists the description for a particular email address. The description is displayed in a single-line edit control, and can be edited. There are two buttons, OK and Cancel. Both close the dialog; OK commits the change, Cancel discards it.

Confirm Revoke

This dialog box brings up the message, "This will permanently delete the greeting card you gave Aileron of Milky Way. Are you sure you want to do this?" There are two buttons, OK and Cancel, which do exactly what you think they do.

Confirm Delete Message

This dialog box brings up the message, "Are you sure you want to delete this message?" There are two buttons, OK and Cancel, which do exactly what you think they do. If there is a Read Message box open for a deleted message, that box is closed when the message is deleted.

Confirm Discard

This dialog box brings up the message, "You have not sent this message. Are you sure you want to discard it?" There are two buttons, OK and Cancel, which do exactly what you think they do.

Confirm Delete Name

This dialog box brings up the message, "This will permanently delete the greeting card you received from Aileron of Milky Way. Are you sure you want to do this?" There are two buttons, OK and Cancel, which do exactly what you think they do.

Tasks

- Web browser capability within the PO engine
- Interfaces for the mail system
- Hooks to the OWO client

To be addressed

- Need to get the mail backend description from Steve Beeman.

Rendering Requirements and Specifications

Overview

Wing Commander Online: Privateer requires a rendering solution that will remain competitive in the market two years after launch. To achieve this goal, the rendering solution must provide access to all hardware acceleration that is currently available, scale from low end machines to those that don't even exist yet, be capable of generating ground, space, and user interface elements of high quality, all while maintaining an acceptable frame rate.

Vision and goals

Wing Commander Online must present two different visual environments in order to provide the player a complete experience. In one environment, the players pilot their spacecraft in the vastness of space. In the other, players interact face-to-face through their character avatars on the surfaces of planets.

The Wing Commander line of products has always provided stunning visuals, and Wing Commander Online is no exception. Not only does the rendering solution need to be impressive upon release, but must also remain competitive for at least another two years.

General Requirements

- Hardware accelerated transformation and lighting
- Multi-stage and multi-pass texture mapping
- Full alpha blend mode support
- Fallback system for varying hardware capabilities
- Efficient hardware state management
- Texture and vertex buffer management
- System for allowing game specified culling methods
- System for allowing game specified alpha polygon handling

Space Requirements

- Ability to get many ships on screen at once with an acceptable frame rate
- Algorithmic generation of space volumes with features such as asteroids, gas clouds, etc.
- Flexible camera that supports all of the views common to space-sims
- Modular ships with articulating parts
- Hueing textures and decals on ships
- Dynamic lighting

Ground Requirements

- Ability to get 50+ figures on screen at once with an acceptable frame rate
- Algorithmic terrain generation
- Non-static environment and the ability to add structures to the map on the fly

- 3rd person top down camera perspective
- Ability to use pieces-parts clothing on figures, preferably with layering
- Hueing textures
- Dynamic lighting

Systems design

Triangle engine

The triangle engine is the foundation of the rendering system. Both the space flight and ground portions of the game use the triangle engine to render their scenes. It provides a consistent interface to the 3D accelerator regardless of what hardware is being used. The triangle engine will make use of existing 3D APIs like Direct3D and OpenGL.

Meshes

Meshes are lists of triangles that are used to represent objects like fighter craft, capital ships, and buildings. They can be placed in hierarchies to allow for articulated "child objects". Additionally, meshes have a level of detail system in place that allows the game to reduce or increase the polygon count for a mesh, depending on the current scene.

Character Avatars

These are deformable meshes that are attached to an underlying skeletal system. As the skeleton is animated, the attached mesh follows the animation to provide a smooth, continuous animated mesh. This system will be used for all "creatures" in the game, including the player characters themselves. They will be customizable through modular pieces and parts to allow things like clothing and facial features. A level of detail system will also be in place to help reduce or increase polygon detail dependent on the current scene.

Planetary Terrain

In order to reduce storage and bandwidth, all planetary terrain will be algorithmically generated. This system will create interesting terrain according to planetary type and populate it appropriate flora and interest items.

Visual Effects

These include items like space backdrops, explosions, laser beams, shadows, and the like. The triangle engine must allow enough flexibility for the game to custom build these effects, and make them scalable according to the available hardware.

Culling

The culling system is the most complex of the rendering system, and the most time consuming to build. The space flight portion, and the ground portion of the game will require independent view databases (i.e. methods of storing scene information for efficient rendering). Space scenes are very sparse with a highly flexible camera system. Ground scenes are very dense with a more restricted camera system.

To be addressed

Currently, two different rendering solutions are being considered: one externally developed (Simis), and the other internally developed.

The Simis engine, Daedalus, is an all-in-one game engine, not just a rendering solution. However, we will likely not want to use each component (for example, their networking solution will not work for WCO).

Jeff Grills and his programming staff would build the internal solution. Programmers from the WCO team (and possibly other teams) could provide additional programming help.

Simis Advantages

- Potentially could reduce development time.
- Ongoing extra engine programming help from Simis engineers.
- Supports early prototyping.
- Existing support for 3D Studio MAX, Photoshop, and Multigen.

Simis Disadvantages

- Simis is located in England.
- The version of Daedalus we are currently testing is the DirectX 6 version. The DirectX 7 version (supporting hardware transformation and lighting) will not be available until around October 22, at the earliest (3 weeks from now).
- Programmer ramp-up time.
- Writing C++ "wrappers" for Daedalus may reduce the engine's performance and consume programmer time.

Internal Solution Advantages

- Built locally.
- Reuse of code from the Vision engine (and possibly the Bootprint engine, depending on how negotiations go).
- Input from all Origin programming teams will provide a solution better suited to Origin's needs.
- Origin owns the engine.

Internal Solution Disadvantages

- Time to build the engine
- Risk of building a new engine
- Simultaneously supporting multiple and divergent projects
- Team to build technology is not yet in place

Ancillary Revenue

Overview

An objective of the *Privateer Online* design is to provide as many revenue streams as possible.

Vision and goals

Goals

- Target numbers provided by the Marketing department are broken out thus for different audiences:

	Total revenue	Subscription	Ancillary
Sci-fi RPG gamer	\$180	\$120	\$60
Space combat gamer	\$90	\$60	\$30
Online sci-fi community	\$30	\$30	\$0

Vision

- ☐ Must keep backend costs low.
- ☐ Should have a low target bandwidth.
- ☐ Game must be playable via the Web in some fashion.
- ☐ Must permit means of "playing" while offline.
- ☐ Should blur the line between the game and reality.

Walkthrough

- Advertising revenue on the website
- In-game and on-website promotional opportunities
- Promotional opportunities that cross into the real world
- In-game gambling for real money
- Action figures/toys
- Web-only play model
- Additional fee for extra characters
- Premium shard for voice chat and other features

Systems design

There are different categories of ancillary revenue: Subscription, Extras and Marketing.

The goal of subscription revenue would be to offer different pricing structures for different features. Regular, Gold and Platinum accounts.

Extras are items that are purchased on an as needed basis.

Detailed below are the ideas we are thinking of.

Web Account

Cost: \$4.95 / month

Offer web play, with or without the box version of the game. The goal of the web play is to allow a player to play the game without having ever logged into the game itself. If the player does have a character in the game, then the player can also access his character and perform limited functions with him.

Web play should be communicated as an *extra* value feature such that we can charge \$4.95 / month for web access. Web access does not come free with a regular account.

Extra characters

Cost: \$2.95 / month

Since we plan to sharply limit the number of characters per account to one or at most two, allowing players to have additional characters for a nominal fee on their account could serve as a major source of additional revenue. Low to moderate backend costs.

Cryogenic storage

Cost: \$1.95 / month

If people choose to discontinue their account, we should offer them cryogenic storage

Real-Time Stock Quotes and Real-Time ticker

Cost: \$1.95 / month

For those day traders out there, we can offer real-time quotes that aren't delayed by 20 minutes

Players wishing to watch the in-game stock market more closely will have access to low-delay ticker information. Low to moderate backend costs.

High-Rollers galaxy

Cost: Whatever they want to buy.

A galaxy (shard) where the player pays real money for credits (gold) to buy what he wants. Note that this will change the game dynamics in a huge way by effectively removing the need to earn money. People will still need to explore and mine to be able to build things that they can't buy.

Real-world services

Being able to order books from Amazon.com or pizzas from Domino's in-game for a small cut off the top.

PDA Upgrades (Personal Digital Assistant)

Cost: \$1.95 / month

Character PDA's are upgraded with additional memory units (MU), allowing them to retain more in-game data such as e-mail, blueprints, and other intellectual property. Low to moderate backend costs.

Unrestricted access

Cost: \$1.95 / month

We can have areas in the game that have restricted access. It has been shown that people like to go places where others can't go. These areas will also be a less crowded.

Action figures/toys

Cost: unlimited

Of particular interest is the idea of toys that players can custom-order that match their ship or character.

Email customization

Cost: \$2.95 / month

Allow the players to customize when they receive emails (e.g. their mining facility is being attacked, their stock dropped below a certain price, they sold an item on the commodities market).

Pager customization

Cost: \$2.95 / month + initial sign up fee.

Allow the players to customize when they receive pages (e.g. their mining facility is being attacked, their stock dropped below a certain price, they sold an item on the commodities market).

Charter Edition box

Cost: \$99.95 direct to OSI

Create limited edition, numbered boxes only available direct from OSI. Included would be extra goodies, as well as 3 months free.

Designer Clothing

Cost: unlimited

Usually not a great revenue generator, but definitely a cool thing to do and creates tons of loyalty.

Real items for sale at the far end of the galaxy that you can only buy by going to that store. "My friend went to the Pollux and all I got was this stinkin' t-shirt"

Items that are only available for purchase if you are at the top of a leader board for 1 month, 3 months, 1 year, etc...

I'd avoid selling items based upon how many hours someone has logged in. Better not to promote high bandwidth gameplay.

Awards for longevity

Cost: 0

We'd create great loyalty for giving out awards for people that subscribe for a long time. Maybe a t-shirt for 6 months, plaque for 1 year, etc... The profit is difficult to calculate, but I'm certain that people would hang out a bit longer if they were close to the next award. Would also create a cult following.

Pricing plans

Cost: variable

Buy 11 months, get one free kind of stuff.

I'd avoid (or think through very clearly) about offering stuff like 1 month free for signing up a friend. I think that there will be many ways to exploit something like that.

Company (guild) ads

Cost: undetermined

Let companies (guilds) buy ads on the WCO:P website, or even in game, to promote their company.

OSI Escrow Service

Cost: 10% of transaction

OSI will hold a WCO:P item or account. The seller will give to OSI their item they are selling. OSI will then tell the buyer that we have it and it's now ok to send the money. Once OSI has the funds, they will inform both parties, release the property to the buyer, and pay the seller.

Extra bandwidth

Cost: \$4.95 / month

No clear ideas yet on what we could send as extra bandwidth. Time will tell what we can offer.

OSI Hosted Company webpace

Cost: \$4.95 / month

Since not everyone has their own personal webpace, we could host it for them for the purpose of them promoting their company.

This could cause problems if someone puts up pornography. We'll have to think through this one a lot.

Feature	Regular \$9.95	Gold \$14.95	Platinum \$19.95	extra charge
Extra characters		•	•	\$2.95 / month
PDA upgrade		•	•	\$1.95 / month
Unrestricted access		•	•	\$1.95 / month
Real-Time stock quotes			•	\$1.95 / month
Email customization			•	\$2.95 / month
Extra bandwidth			•	\$4.95 / month
Cryogenic storage				\$1.95 / month
Web account				\$4.95 / month
High-Rollers galaxy				unlimited
Real World Services				service charge
Action figures / toys				unlimited
Charter Edition box				\$99.95
Designer Clothing				unlimited
Pricing plans				variable
Company (guild) ads				undetermined
OSI Escrow service				10% of transaction
Pager customization				\$2.95 / month
OSI Hosted webpace				\$4.95 / month

Other random ideas not fleshed out, just listed so as not to forget them:

In-game gambling

Use of gambling devices in the game could use funds from the player's account, automatically deducting monies.

Website ad revenue

The Privateer Online design emphasizes the ability to conduct key game transactions via the website. This provides not only a way to maximize offline play in order to minimize bandwidth costs, but also a prime method to drive impressions on the page, making it a potential source of advertising revenue.

If we ever do a demo server, this could also be a place where ad revenue can be generated.

Promotional opportunities

Numerous leader boards and services within the game could plausibly be cross-promotional opportunities, such as a top ten writers list which is labeled as the New York Times Bestsellers via a cross-promotion, or Billboard's Top Ten musicians, Michelin Guide bars, etc.

Sell user generated content, like Descent Levels of the World.

Interface design

TBD

Tasks

TBD

To be addressed

We need to decide which of these to pursue and what they would entail.

Tasks

To be addressed

- Can we add a "privateer" account that will only grant us a small reward, but which we can use specifically to buy equipment for our first mission? (Thinking here of the model that online TV uses for its "free weekends")
- In what other way can the web play serve as a device?
- How much risk do we want to take that the idea of an CD model will be made use of this and it will be the wrong product to a large extent?

Other random ideas not fleshed out, just listed so as not to forget them:**In-game gambling**

Use of gambling devices in the game could use funds from the player's account, automatically deducting monies.

Website ad revenue

The Privateer Online design emphasizes the ability to conduct key game transactions via the website. This provides not only a way to maximize offline play in order to minimize bandwidth costs, but also a prime method to drive impressions on the page, making it a potential source of advertising revenue.

If we ever do a demo server, this could also be a place where ad revenue can be generated.

Promotional opportunities

Numerous leader boards and services within the game could plausibly be cross-promotional opportunities, such as a top ten writers list which is labeled as the *New York Times* Bestsellers via a cross-promotion, or *Billboard's* Top Ten musicians, *Michelin Guide* bars, etc.

Sell user generated content, like Descent Levels of the World.

Interface design

TBD

Tasks

TBD

To be addressed

We need to decide which of these to pursue and what they would entail.

Demo

Overview

We don't necessarily need a demo—what we need is some way for players to sample aspects of *Privateer Online* play.

Vision and goals

Goals

The goal is to bring in the largest possible quantity of potential customers, with the lowest possible load to our service.

Vision

- ☐ ... must be easily accessible to a mass market audience and a casual player
- ☐ ... should have an intuitive user interface
- ☐ ... should take no more than 30 seconds to impress the user
- ☐ ... should appeal to sci fi fans (Star Wars, Star Trek, Wing Commander) with Internet connectivity
- ☐ ... should offer a great newbie experience for the first 30 minutes
- ☐ ... must satisfy the newbie in the first 5 minutes
- ☐ ... must be easy to get into, as a new player and as an advanced one
- ☐ ... must offer a strong support system so that players feel welcomed into the game and so the game atmosphere is friendly
- ☐ ... must offer clear goals to newbies
- ☐ ... must have a great single-player experience for the first 2-4 weeks of play time
- ☐ ... will stretch for the mass market

Walkthrough

Systems design

Interface design

Tasks

To be addressed

- Can we sell or give away “timelocked” accounts that are only good on a given weekend, for which we maybe specifically lease equipment for one-time showoffs? Thinking here of the model that cable TV uses for its “free weekends.”
- To what extent can the web play serve as a demo?
- How much risk do we want to incur from the fake server on CD model? UO:T2A made use of this and it has so far never resulted in a hack.

Top Marketing Hooks

1. Play the in-game stock market and invest in other players' businesses.
2. Play much of the game via the web.
3. Colonize planets and form cities and have them be formally recognized.
4. Explore hundreds of unique alien worlds.
5. Space combat both against other players and against pirates and alien enemies.
6. Capital ships that take multiple players to operate.
7. Trade in commodities and goods from colonies across the galaxy.

IN GAME AROUND 3 MONTHS AFTER LAUNCH

The Pirkians begin an unprecedented offensive against the Confederation, moving from southern sectors with greater force and frequency than the Confederation anticipates a diplomatic solution. This leads War is declared.

6 MONTHS AFTER LAUNCH

The Confederation is seeking a few good men and women to join the fight. We release the Online Update.

Wing Commander Online: Tour of Duty

This introduces a military track for players to follow, replete with a dynamic front line in the war against Pirkia.

Around five months into the war we introduce another variable. Seeing a weakness in the overconfidence of the Confederation, Sector clans across the Dead zone and begin to attack the Confederation. This is merely increased group size and frequency, as well as new sweep locations for Khraki raiding squadrons.

At the Sixth Month point Pirkia submits, agreeing to cease all aggression and all the other crap that goes along with a conditional surrender. Sector they have started being hit by the Khraki as well, and neither Control or Pirkia has the resources for a third front war.

As the Confederation reflects its focus to the Khraki, a stunning development occurs. It occurs several of the southern sector Clans were contacted with letters to join in the war and Pirkia had Khraki space looking over their shoulder with the Confederation. The Confederation is shocked at first to welcome these people, but needs to realize it can't get. This infuriates the aggressive Khraki, and the war is on.

Plans for Extensibility (Brainstorm)

Utilizing the second disc worked into cost of goods scheme:

Wing Commander Online: Privateer

Confed is not a part of the bargain for the players at this point. The systems and function of Military service, ranking, leadership etc. need a lot of work...So at our sixth month point the following happens:

IN GAME AROUND 5 MONTHS AFTER LAUNCH:

The Firekkans begin an unprecedented offensive against the Confederation. Moving into southern sectors with greater force and frequency the Confederation attempts a diplomatic solution. This fails. War is declared.

6 MONTHS AFTER LAUNCH:

The Confederation is seeking a few good men and women to join the fight. We release the Online Update:

Wing Commander Online: Tour of Duty

This introduces a military track for players to follow, replete with a dynamic front line in the war against Firekka.

Around Five months into this we introduce another wrinkle. Seeing a weakness a loose conglomeration of Northern Sector clans crosses the Dead zone and begins to assault the Confederation. This is merely increased group size and frequency, as well as new spawn locations for Kilrathi raiding squadrons.

At the Sixth Month point Firekka submits, agreeing to cease all aggressions and all the other crap that goes along with a conditional surrender. Seems they have started being hit by the Kilrathi as well, and neither Confed or Firekka has the resources for a dual front war.

As the Confederation redirects its focus to the Kilrathi, a stunning development occurs. It seems several of the southern sector Clans were coerced with terror to join in the war and have fled Kilrathi space seeking asylum with the Confederation. The Confederation is reluctant at first to welcome these people, but needs any edge it can get. This infuriates the aggressive Kilrathi, and the war is on.

This brings us to the second CD:

Wing Commander Online: The Sivarán Code

This introduces (in all its wonderfully divisive ways) both Kilrathi society on the Kilrathi side, and new Kilrathi refugees attempting to help Confed in any way they can (Privateer, Military).

On the Confed front, these new Kilrathi bring new technology, ship types, cultural needs, structures, etc. etc.

We also introduce Kilrathi pieces parts possibilities with some chassis types. This enables Confed/Kilrah ship hybrids...keen!

The Firekkans having attempted to hold off the Kilrathi on their own, were too weakened in their war against the Confederation. Talks are opened up and an alliance is formed.

We include on this disc the art/sound components of the Firekkans, although they aren't close to being utilized yet. This leads to the second major online update:

Wing Commander Online: Firekkan Incentive

Now Firekkans are present, both in their own culture, and in the Confederation. There are new systems to explore, a new social structure to unravel, and new Kilrathi to engage. Also, we have Firekkan ships, and possible Kilrah/Confed/Firekka hybrids.

This is all just a big brainstorm...but it does represent two years of constant story points and serialized incidents we can explore.

It also pushes off some of the more complicated Confed military assets to post launch (real Capship management, dynamic front lines, etc. etc.)