## A Quick note on using the app – please read, so I don't have to hear about how you hurt yourself

This is a virtual reality application, meaning that when using it, you are not a participant in the outside world, even though your senses of orientation and proprioception are being used to control you projection into the virtual world.

YOU SHOULD ALWAYS BE SEATED WHEN USING THIS APPLICATION. NEVER EVER EVER GET UP AND MOVE AROUND. VERY BAD THINGS ARE GUARANTEED TO HAPPEN IF YOU THINK YOU CAN KEEP THE VIRTUAL AND THE REAL STRAIGHT AT THE SAME TIME.

The best way to use this app is seated in an spinnable office chair in the middle of a room. You will need to turn and look a lot to use this application. If you don't have a helmet so you can use the device as a VR helmet I strongly recommend you get a Durovis Dive 7 headset.

## First Time

When First Light starts, you are in orbit of the Earth.

Say "exit orbit" or "leave orbit". Once this has been recognized the system will tell you by voice you are leaving orbit and the spacecraft will begin to fly in the direction you are looking. Go find the Moon. As you approach the Moon, say "one half speed" or "half speed" or "halve speed" to slow down. You can also say "increase speed" and "decrease speed", or even just "faster" or "slower" to fine tune your speed. Try orbiting the moon.

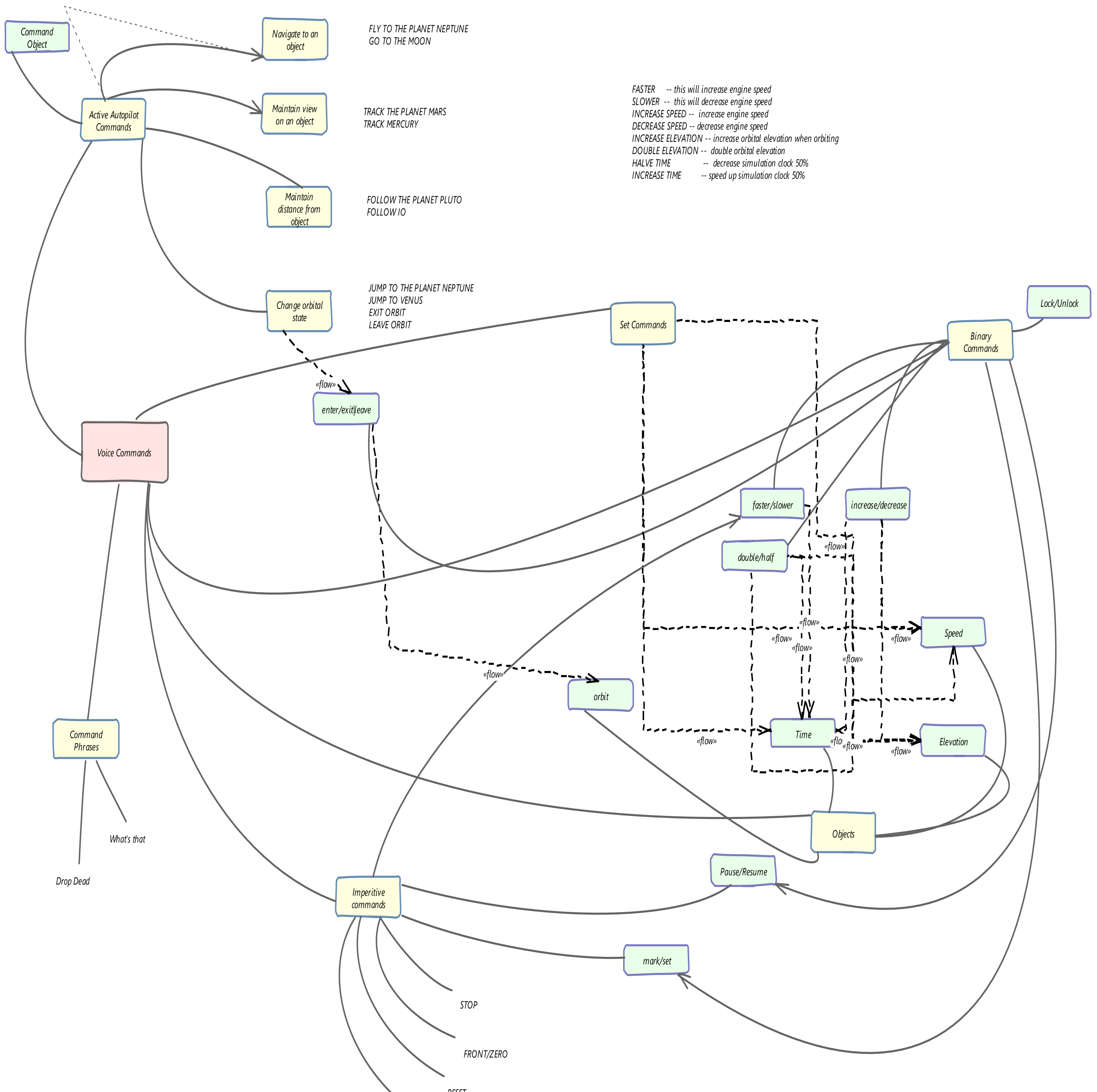
Say "Fly to Jupiter" or "Fly to the planet Jupiter" or "Go to the planet Jupiter". You will fly to Jupiter and enter orbit. Look around while you are orbiting. Then say "exit orbit" or "leave orbit". Go explore the Jovian moons. You may want to control the simulation clock speed at this time, some of those moons spin quickly. Say "decrease time" and "increase time" to adjust the simulation clock. You can also say "double time" several times but you might not like the result. Now try this with other planets. Try moons to, but remember. Pronunciation is tricky. And you can't "fly to the moon io" because the system wouldn't be able to figure out which moon you meant.

"Stop" will cancel all active autopilot programs and make you motionless.
"Reset" will reset the Tango motion capture.

"What's that" or "What is that" will tell you something about what you are looking at.

"Drop dead" will terminate the application.

The full speech command set is (poorly) documented below.


Command Object
Active Autopilot Commands
Navigate to an object
FLY TO THE PLANET NEPTUNE
GO TO THE MOON
Maintain view on an object
TRACK THE PLANET MARS
TRACK MERCURY
Maintain distance from object
FOLLOW THE PLANET PLUTO
FOLLOW IO
Change orbital state
JUMP TO THE PLANET NEPTUNE
JUMP TO VENUS
EXIT ORBIT
LEAVE ORBIT
FASTER -- this will increase engine speed
SLOWER -- this will decrease engine speed
INCREASE SPEED -- increase engine speed
DECREASE SPEED -- decrease engine speed
INCREASE ELEVATION -- increase orbital elevation when orbiting
DOUBLE ELEVATION -- double orbital elevation
HALVE TIME -- decrease simulation clock 50%
INCREASE TIME -- speed up simulation clock 50%
Voice Commands
enter/exit/leave
«flow»
Set Commands
Binary Commands
Lock/Unlock
faster/slower
increase/decrease
double/half
Speed
Time
Elevation
orbit
Objects
Pause/Resume
mark/set
Command Phrases
What's that
Drop Dead
Imperitive commands
STOP
FRONT/ZERO

## Active Autopilot Commands

Autopilot commands are those that cause your spacecraft to be manipulated for you by the all knowing autopilot. The autopilot can keep you in orbit of a body, it can fly you from any location to any body, and it can keep your position, or your viewpoint fixed on any body.

It should be noted that flight calculations are exceptionally poor and need to be replaced with something that has a better understanding of orbital mechanics - however that conflicts with any idea of fun, as if you could only use realistic orbital mechanics, this simulation would be tedious and boring. For the moment, simple hit detection is used, too many flight plans are worked out right through the Sun, and there's a bit of banging about trying to get around it.

When an autopilot command is active, you will see a status indicator for it in the upper left view for your left eye. Still not sure what to do about overlays on stereoscopic views.

### Maintain distance from object

The Follow command is used to cause the spacecraft to maintain its distance from an object. This command remains in effect until a "Stop" command is given.

"Follow Earth"
"Follow the planet Earth"

### Maintain view on an object

The Track command is used to lock the camera on an object. This lock will be maintained until a "stop" command is given. The two operative terms in the sentence are the track verb and the target -

Short form - "Track Jupiter"
Long form - "Track the planet Jupiter"

Note you cannot say "Track the moon Io" because moon and Io would both be recognized as targets and the system would complain. To help google speech out, try "Track the solar body Io"

### Navigate to an object

Fly is the command to have the system automatically plot and follow a course. Note that orbital injection/ejection math is not currently in place, so the transition from arrival to in-orbit is somewhat abrupt.

"Fly to Jupiter"
"Fly to the planet Jupiter"

### Change orbital state

You can immediately enter orbit of any body, regardless of current location by using the JUMP command, e.g.

"Jump to the planet Neptune" means that on the very next frame you expect to be in orbit of Neptune.

When FLY or GO are used, then the trip to the destination is performed, concluding with transition to orbit sans benefit of any pesky calculations about orbital injection.

The system is capable of taking orbit on any orbital entity. The math for a nice orbital injection is busted, so the transition from "I can orbit this" to "I am orbiting this" is instantaneous, and admittedly a bit jarring. Orbital injection math is hard hard hard.

Orbital speed is automatically calculated to align with the bodies spin rate, with an adjustment to make the view 'interesting'. Note that if you attach to say, a Jovian moon with an outrageous spin rate, you're going to have an outrageous spin rate to.

### Getting into orbit

Fly to the planet "X", as in
"Fly to the planet Jupiter" or "Fly to the Moon" will fly you to the target and place you in orbit.
GO can be used in place of Fly, as in
"Go to the planet Neptune"

Jump to the planet "X", as in
"Jump to Mars" or "Jump to the planet Earth" will immediately place you in orbit of the named body, without any need to travel there first.

## Binary Commands

Binary commands are so named because they represent bifurcations in system behavior, or, put another way, "it's always something — if it ain't one thing, it's another." (GR/SNL)

Common examples are numerical changes, where the operation splits on whether the value is being increased ("faster","increase","double") or decreased ("slower","decrease","halve").

Binary commands are a morphological classification, they are not a functional classification like autopilot commands, hence these commands are distributed across functional operations. The primary reason they are broken out is to ensure that users can expect certain basic behaviors that these commands allow for.

Consider the binaries at this time.
increase and decrease
double and half

Consider the entities they operate on,
time - the simulation clock speed
speed - the speed of your spacecraft
elevation - your orbital elevation, when you are orbiting something

The basic sentence form is
{binary} {object}, as in "increase speed" or "halve time". In order to aid the speech recognizer you can use these in larger sentences, such as "increase the engine speed" or "halve the simulation time"
double/half

## enter/exit|leave

Currently only meaningful with respect to orbit

"Enter orbit" is broken - Use "Jump to planet X"

"Exit orbit" or "Leave orbit" will cause you to remain in the same location, but you will no longer be locked into orbit of the planet, i.e. you can move around.

Your transferred velocity on orbital ejection will be a function orbital bodies angular velocity. The net result is that your speed is sufficient for exploring the neighborhood, i.e. visiting a planets moons.

### faster/slower

### increase/decrease

### mark/set

## Virtual Command Objects

Binary commands operate on these objects.

### Speed

refers to the speed of the spacecraft, i.e. how fast you move through space when you are free flying.

Your speed is automatically set on each orbital ejection in order to give you an optimal velocity for the neighborhood.

"Slower" -- reduce speed 5%
"Faster" -- increase speed 5%
"increase speed" -- increase speed 5%
"double speed" -- double the speed
"one half speed" -- reduce speed 50%

Note that your speed is affect by the simulation clock. If you make time go ten times as fast, you will also move ten times as fast. Be forewarned.

### Time

the word time refers to the system timebase, which controls how fast the simulation moves. The timebase runs from 1 to 1000, where 100 is the default speed when the application starts. When the timebase is set to 1 then the system has little perceptible motion, when set to 1000 you can watch the planets trace out their orbits, and the moons frantically orbit them as they do.

"faster time" == increase time by 5%
"slower time" == decrease time by 5%
"double time" == double simulation speed

"set time to 100" or "set time 100" will set the system to the default speed, whereas "set time to 1000" will probably make you very very dizzy.

## Elevation

The orbital distance from the orbital body center,i.e. how high off the surface of the body you are orbiting

"increase elevation" - increase elevation 5%
"decrease elevation" - decrease elevation 5%
"set elevation to 3" -- directly set elevation - just because you can do this does not make it a good idea. For an example, orbit jupiter, "set time 1000". "set elevation 100" and you've got a ride you probably won't like.

## Command Phrases

Command phrases are sentences that must be spoken as is, i.e. its not some kind of trickery with word patterns. Phrases are used for certain key operations, and for sentences that don't have analogous uses.

### Drop Dead

Speaking the sentence "drop dead" is how you can tell First Light to terminate.

### What's that

"What is that" or "What's that"

The system will attempt to tell you about what you're looking at. Currently it considers everything in scene, which means things you can't even see with the headset on. The focus cone will be tightened.

The goal is that when you say "What's that" and you get a response like "That is the moon Io of Jupiter", you can then say "Go to it" where it is understood to mean Io. Currently this is not understood :-(

## Imperative commands

### LOOK

Sentence should contain an orbital system entity name - the camera will be aimed at that on the next pass - the effect is ephemeral, i.e. it is not like TRACK which will follow the object

Examples

LOOK AT THE PLANET SATURN
LOOK AT THE SOLAR BODY MIMAS

### Lock/Unlock

General case - lock indicates a desire to reduce system variability, unlock a desire to increase it

The only current case is for locking the pose controller to the spacecraft, i.e. the spacecraft automatically assumes the heading of the pose controller.

So you just say "lock" to lock the pose controller to the spacecraft and "unlock" to unlock the controller from the spacecraft

## FRONT/ZERO

The camera is realigned with the operators current viewpoint, i.e. they do not move, but the view rotates such that they are now looking in the direction of travel. This is used when the concept of forward has gotten a bit fuzzy. This can occur when the spacecraft is under the control of an autopilot program and the operator has been looking around.

Either the word FRONT or ZERO can be used

## Pause/Resume

Preserves the current velocity and autopilot programs and then performs a stop operation. The preserved velocity and autopilot programs may be reloaded at a later point by saying "resume" - the system will do its ineffective best to understand how the sudden changes in position and orientation affect the plans (poorly)

"pause" - push current autopilot state and then clear current state
"resume" - load current autopilot state from state preserved by preceding "pause" command

RESET
One word command to issue a reset to the Tango motion tracking system

"Reset" - the display will stop responding to motion while the reset is underway

STOP
Set velocity to zero and cancel all autopilot plans. This does not affect the simulation clock

## Set Commands

Always in the form of "set x to value" or "set x value" where x is a known object that is settable and the value is numeric.

Values that can currently be set are
time - the simulation clock speed, between 1 and 1000
speed - spacecraft engine speed
elevation - orbital elevation, when in orbit.

SET TIME TO 100 or SET TIME 100 to set normal simulation clock
SET SPEED TO 1 or SET SPEED 1 to set the smallest speed you can, which is often still too fast
SET ELEVATION TO 2 to set the orbital distance

Note that elevation and speed are actually expressed in normalized units of measures scaled against the simulation model, i.e. knowing what the actual values should be isn't childs play. SET is excellent for adjusting the simulation clock, after that its a tool for experts.

# High Level Project View

## Alpha

The goal of the Alpha phase is platform and simulation stability. With respect to the platform, this encompasses Tango and all of the Google APIs that are used to support operation, such as the Speech Recognition API. With respect to the simulation, this encompasses orbital mechanics, the command interface that lets the user navigate the spacecraft, and the processing of the solar data to generate a more interesting simulation that still conveys the key realities, to some degree.

## Continuous Recognition

Next generation mobile applications that deliver rich interactive experiences on position and attitude aware devices should not have traditional UIs as their primary interface. This application is an experiment in making an application which is controlled principally by voice commands, as recognized by the Google speech recognition services for Android.

Currently the system operates in continuous recognition. Between this, the 3D, and everything else, run time from full charge to a depleted battery is barely more than an hour.

Will be switching to punctuated recognition - the local device will perform simple match of a key phrase, which will then trigger voice recognition

User: "R2"
Sim: "Ready"
User: "Fly me to the moon"

## Solar System Simulation

The backbone of this application is a simulation of our solar system. Significant liberties have been taken with scientific fact, for it is a scientific fact that our solar system is mostly empty, and any exploration of it requires mostly spending time in the journey from point A to point B.
Real values are used to define the planets and moons, and then a scaling algorithm is run to deal with two messy details. The first is to get the simulation into a world space where the graphics hardware has half a chance of dealing with it, which primarily involves non-linear adjustments to distances to ameliorate Z battling. A secondary algorithm is used to maintain relationships at a local level, i.e. a planet and its moons, even though distances are being shifted around.
The end result is a system where the massive is even more massive and the distances are compressed. Therefore, the Sun is gigantic from any perspective, even from the orbit of Jupiter, but you can get a feel for the difference between a trip from Mars to Jupiter and a trip from Neptune to Jupiter.

## Alpha Exit Criteria

1) Platform is stable, i.e. people can treat it as "it just works"
2) Orbital mechanics all in place - this means unpleasant little bits like the orbital injection/ejection curves.
3) Command infrastructure in place. Whatever you need to tell the spacecraft to do, you can. At all the right levels
4) Simulation Veracity - the right knobs and dials are in place to allow a more or less realistic model to be automatically created from the basics of our solar system.

## Beta

First came the enemies, and then came the weapons systems.
The beta phase involves the integration of an optional module with no educational value but a lot more entertainment value. Development of this will also serve as a gap analysis for the baseline system, which is expected to grow to cover identified deficiencies.

## Two Simulation Lines

A) 2001 Storylines - find the monoliths, one by one - if all goes well, the last one will convert Jupiter to a Sun - runs on top of the baseline sim

B) Spacecrafts and Lazers, oh my - the ship is augmented with weapons systems, and some opponents are provided

Cardboard Branch
The app will be branched to support Cardboard

Release
QED

Release Plan
Various elements of the system will be brought online during each of the release phases

VR Application