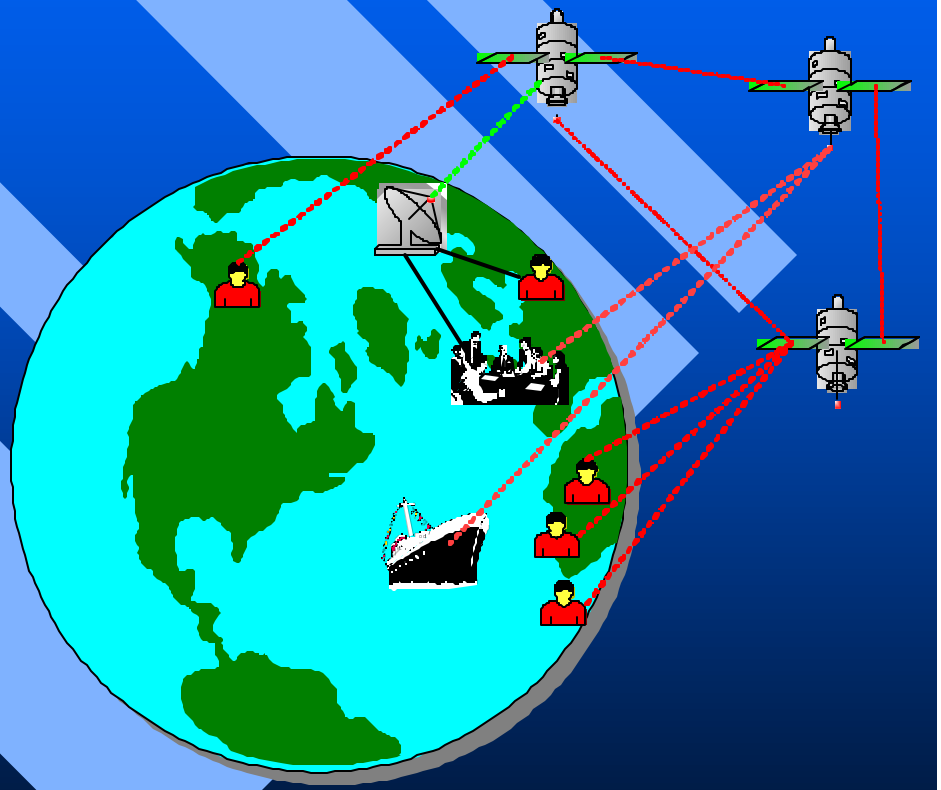
Constellation Simulation System



Presented by Fan Wu

CS525 S04

Instructor: Prof. Emmanuel Agu

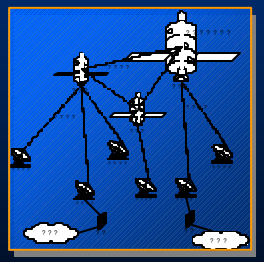
Apr. 27, 2004

Outline

- Introduction
- Major Work
- Conclusion
- System Demo

Introduction

- Introduction
- Major Work
- Conclusion
- System Demo



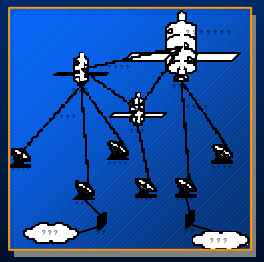
4 “W”

- **Whenever**
- **Wherever**
- **Whoever**
- **Whatever**

Satellite communication is the best way to implement such kind of personal communication

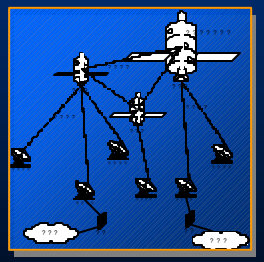
Proposed Work

- Introduction
- Major Work
- Conclusion
- System Demo



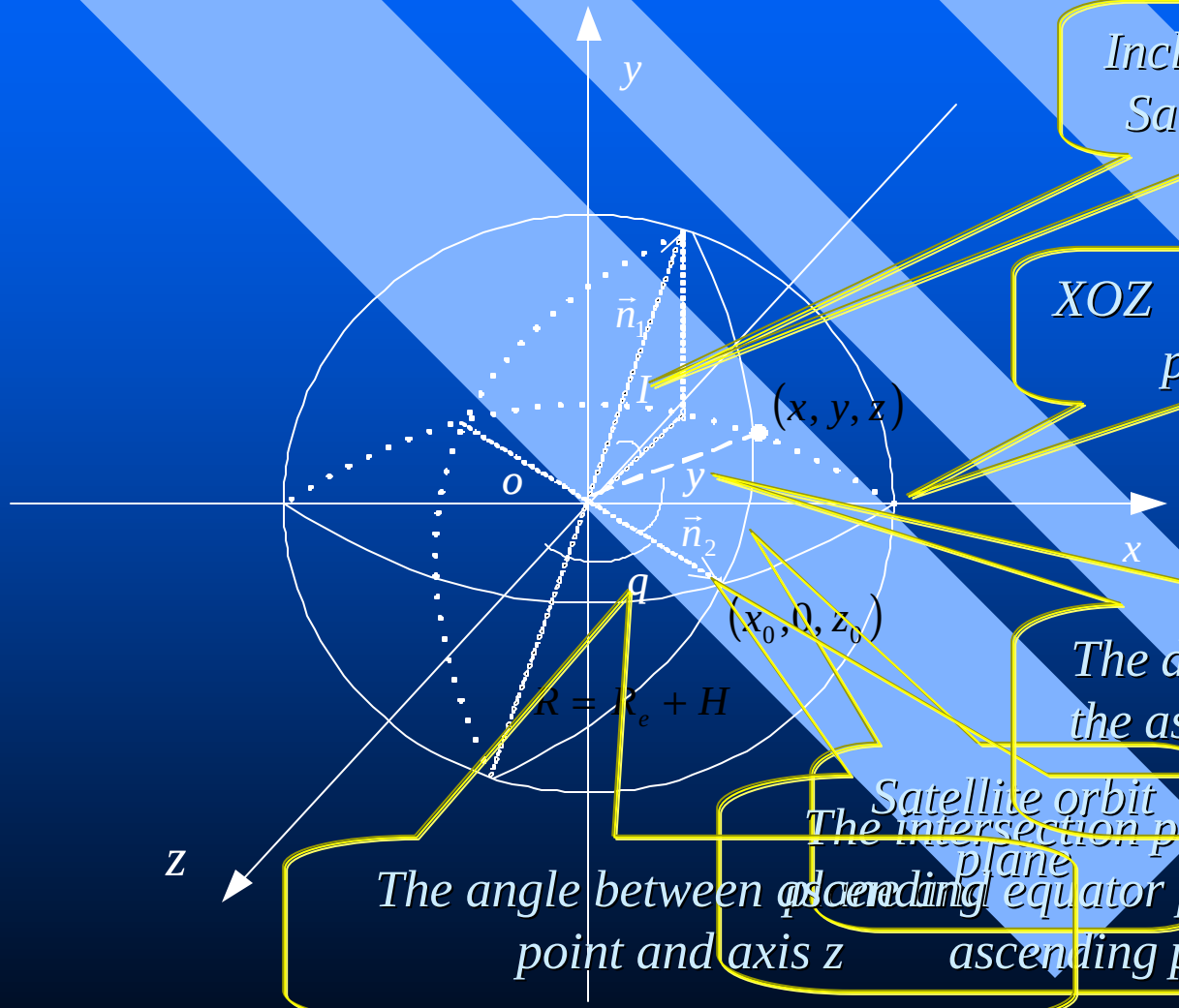
- **Develop a mathematical constellation model**
- **Implement a visual and developable constellation simulation system**
- **Load a simple routing algorithm**

Major Work



- Mathematical Modeling
  - ✓ Constellation 3D Model
  - ✓ Orbit 3D Equation
  - ✓ Satellite Moving 3D Equation
  - ✓ Hiding Equation
- Implementation
  - ✓ Programming
  - ✓ Load Shortest Routing Algorithm
  - ✓ Programming Language: Dephi

Constellation Model



Incline angle of Satellite orbit

XOZ is equator plane

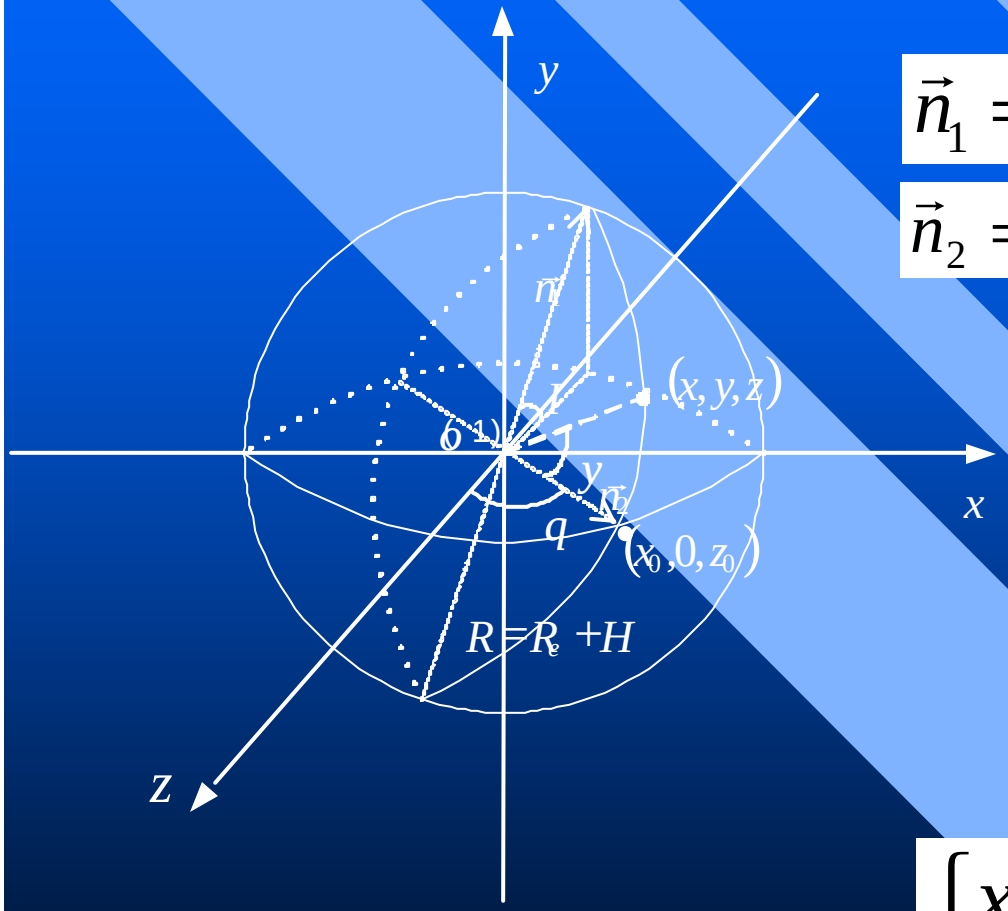
The deflection angle from the ascending points. (0^0-360^0)

Satellite orbit

The intersection point of orbit plane and equator plane named ascending point

The angle between ascending point and axis z

Orbit 3D Equation



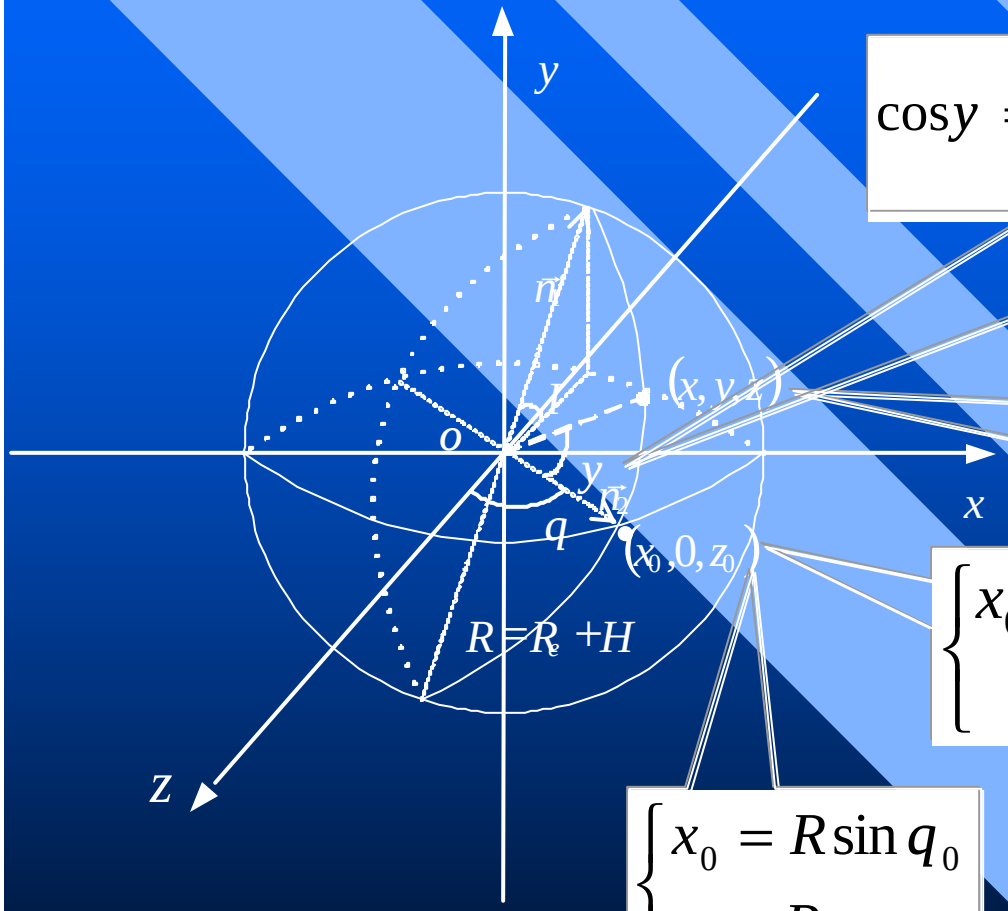
$$\vec{n}_1 = \{\cos i \cos q, \sin i, -\cos i \sin q\}$$

$$\vec{n}_2 = \{\sin q, 0, \cos q\}$$

$$\vec{n} = \vec{n}_1 \times \vec{n}_2 = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \cos i & \cos q & -\cos i \sin q \\ \sin q & 0 & \cos q \end{vmatrix} = \{\cos q, -ctgi, -\sin q\}$$

$$\begin{cases} x\cos - yctgi - z\sin q = 0 \\ x^2 + y^2 + z^2 = R^2 \end{cases}$$

Satellite Moving Equation



$$\cos y = \frac{xx_0 + zz_0}{\sqrt{(x_0^2 + z_0^2)(x^2 + y^2 + z^2)}} = \frac{xx_0 + zz_0}{R^2}$$

$$\begin{cases} x\cos - yctgi - z\sin q = 0 \\ x^2 + y^2 + z^2 = R^2 \end{cases}$$

$$\begin{cases} x_0\cos q - z_0\sin q = 0 \\ x_0^2 + y_0^2 = R^2 \end{cases}$$

$$\begin{cases} x_0 = R\sin q_0 \\ z_0 = R\cos q_0 \end{cases}$$

Satellite Moving Equation

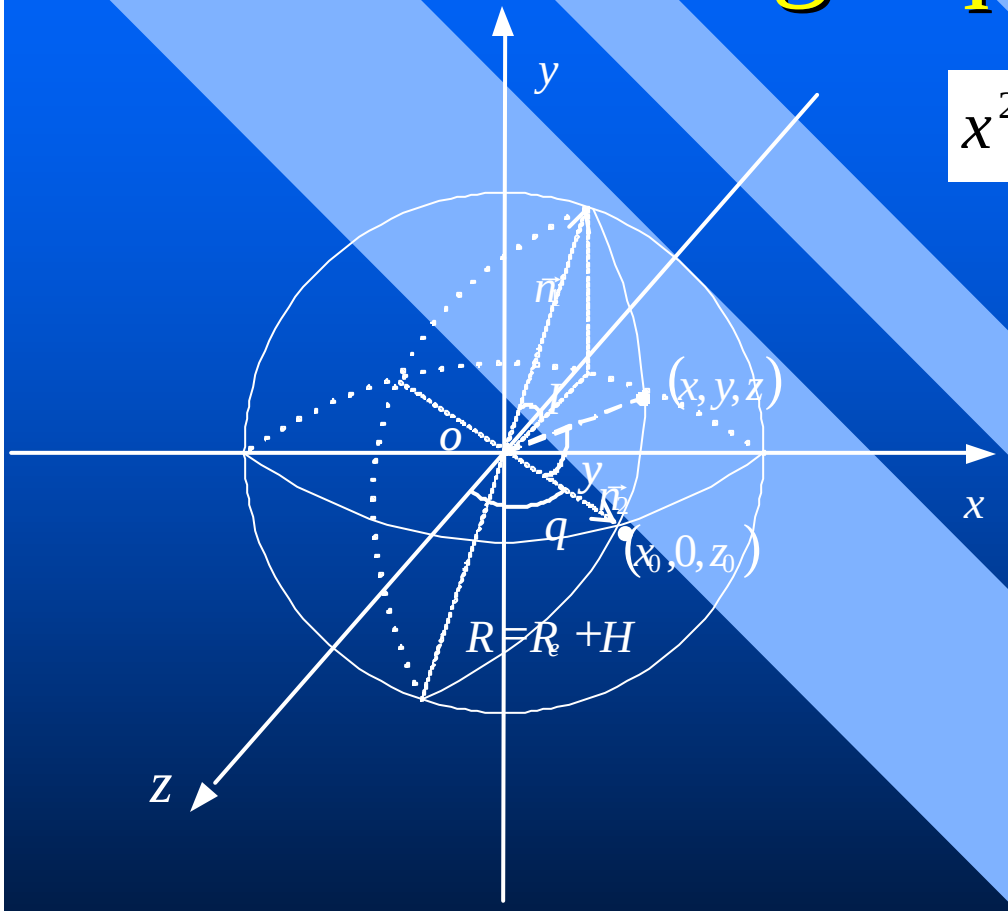
$$\begin{cases} x_0 = R\sin q_0 \\ z_0 = R\cos q_0 \end{cases}$$

$$\begin{cases} x\cos - yctgi - z\sin q = 0 \\ x^2 + y^2 + z^2 = R^2 \end{cases}$$

$$\begin{cases} y = R\sin y\ \sin i \\ x = y\cos qctgi + R\sin q\ \cos y \\ z = -y\sin qctgi + R\cos q\ \cos y \end{cases}$$

$$\cos y = \frac{xx_0 + zz_0}{\sqrt{(x_0^2 + z_0^2)(x^2 + y^2 + z^2)}} = \frac{xx_0 + zz_0}{R^2}$$

Hiding Equation



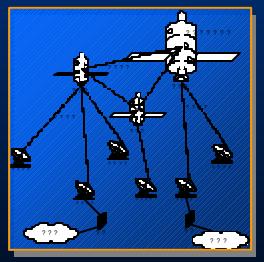
$$x^2 + y^2 = R_e^{\,2} \qquad x^2 + y^2 + Z^2 = R^2$$

$$Z^* = -\sqrt{R^2 - R_e^{\,2}} = -\sqrt{H(R+H)}$$

$$Z < Z^* = -\sqrt{H(R+H)}$$

Mathematical Modeling

- Introduction
- Major Work
- Conclusion
- System Demo



Orbit 3D Equation

$$\begin{cases} x\cos - yctgi - z\sin q = 0 \\ x^2 + y^2 + z^2 = R^2 \end{cases}$$

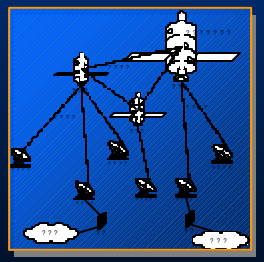
Satellite Moving Equation

$$\begin{cases} y = R\sin y \ \sin i \\ x = y\cos qctgi + R\sin q \ \cos y \\ z = -y\sin qctgi + R\cos q \ \cos y \end{cases}$$

Hiding Equation

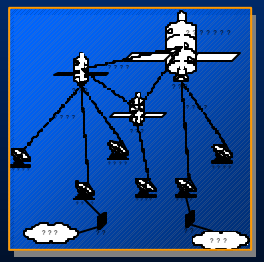
$$Z < Z^* = -\sqrt{H(R+H)}$$

Basic Idea



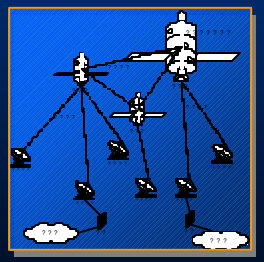
1. Based on the equations of orbit and satellite, with time going on, calculate the coordinates of each satellite.
2. Call Tconstellation.Refresh function at a certain time interval to simulate constellation.
3. Using the coordinate information provided by the simulation to constellation to load routing algorithm

Basic Objects



1. TSatellite Object
2. TTrack Object
3. TConstellation Object

Basic Objects



1. TSatellite Object

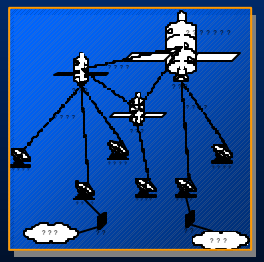
```
Function Visible:Boolean; Hiding or not;
Procedure Create;         Create satellite model;
Procedure Show;           Draw the satellites
Procedure Refresh;        Draw the satellites at the position next time;
```

2. TTrack Object

3. TConstellation Object

Basic Objects

- Introduction
- Major Work
- Conclusion
- System Demo



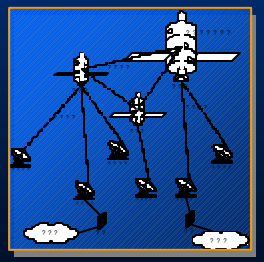
1. TSatellite Object

2. TTrack Object

```
procedure Create;          Create Track model;
procedure DrawTrack;       Draw the tracks
procedure Refresh;         Draw the tracks at the position next time;
```

3. TConstellation Object

Basic Objects



1. TSatellite Object

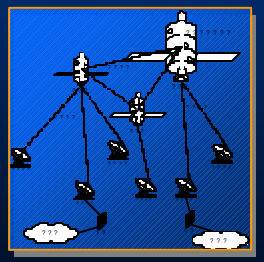
2. TTrack Object

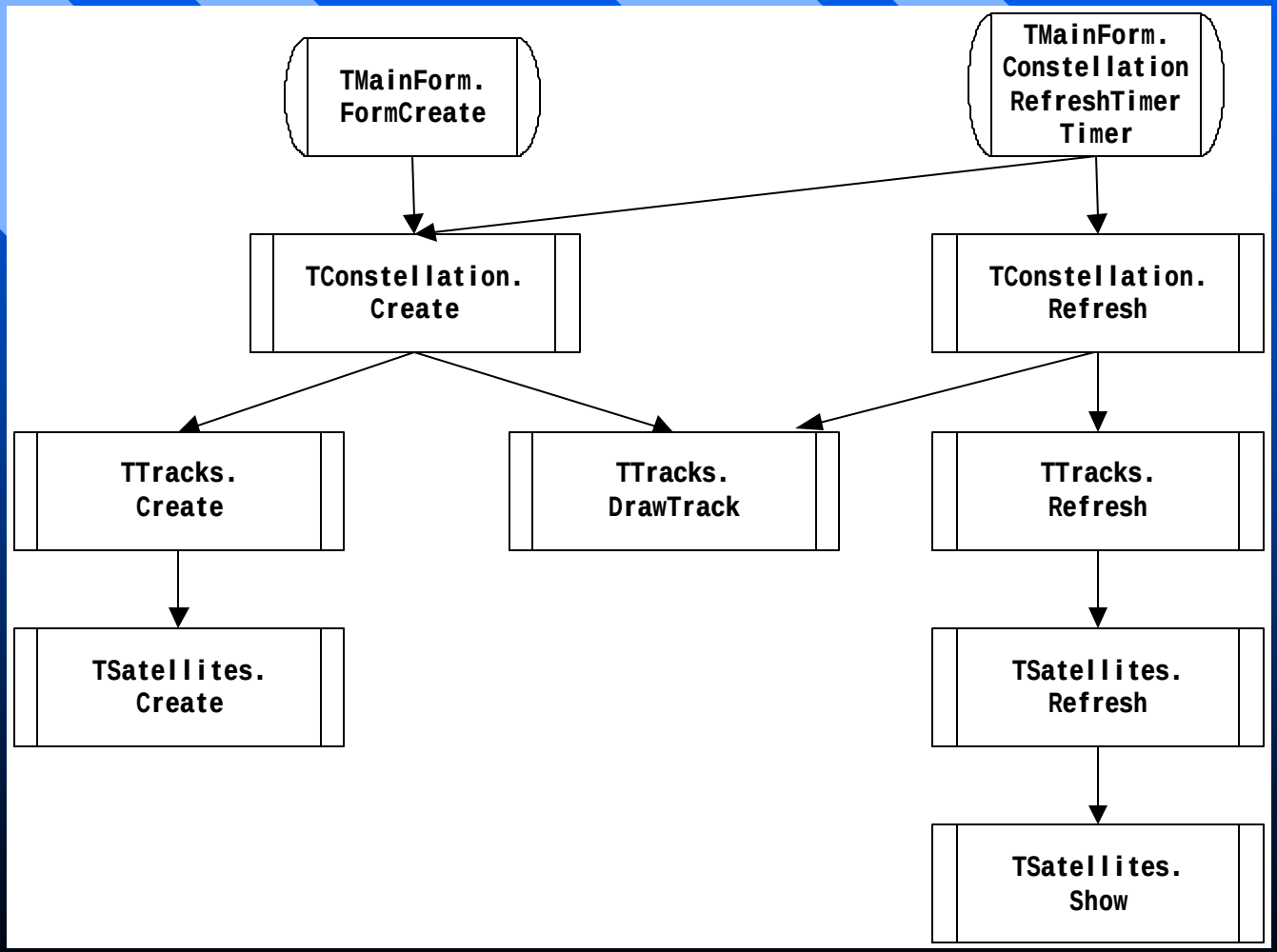
3. TConstellation Object

```
procedure Create;          Create Constellation model;
procedure Refresh;         Draw the Constellation at the position next
                           time;
```

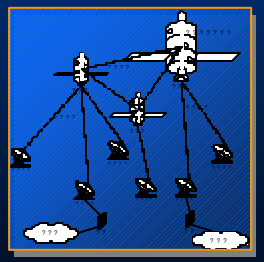
Relationship Between Functions

- Introduction
- Major Work
- Conclusion
- System Demo





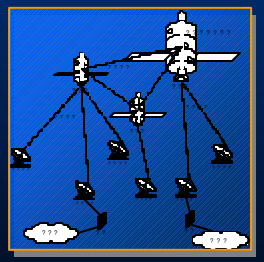
- Introduction
- Major Work
- Conclusion
- System Demo



Major Accomplishment

- Develop a mathematical constellation model
- Implement a visual and developable constellation simulation system
- Provide user a tool to design constellations
- Load and simulate a simple routing algorithm

- Introduction
- Major Work
- Conclusion
- System Demo

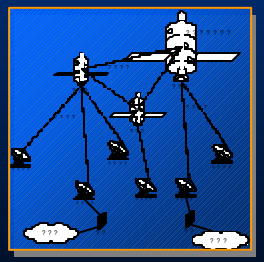


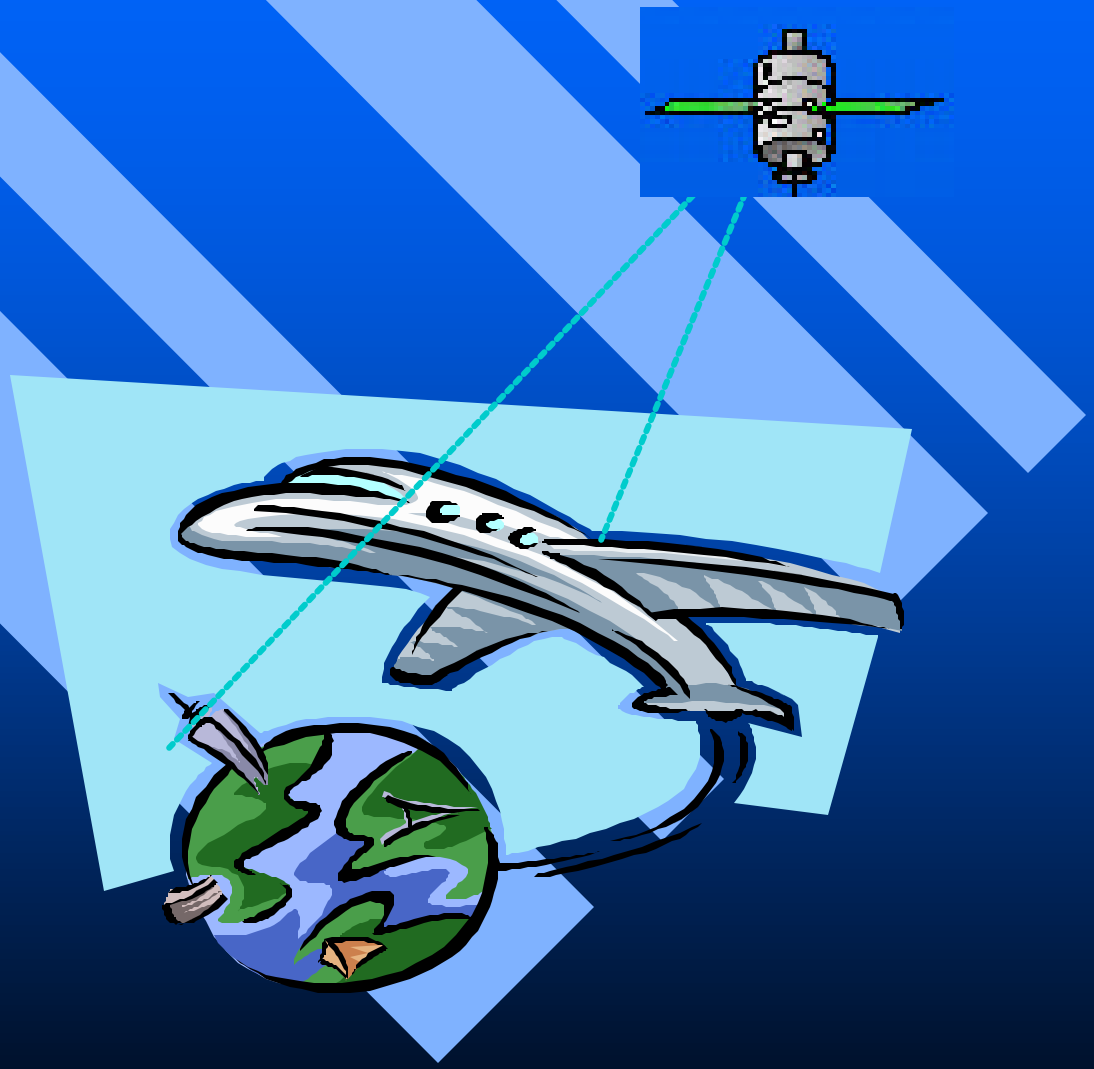
Work To be done …

- Load more routing algorithm
- Improve the performance of the system
- Research a way to cooperate with the exising software

System Demo

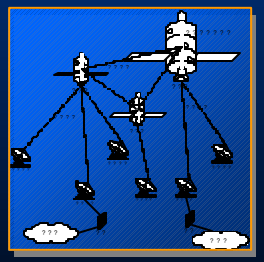
- Introduction
- Major Work
- Conclusion
- System Demo





Constellation Simulation System

- Introduction
- Major Work
- Conclusion
- System Demo



Questions ?