Spacecraft Simulation and Visualisation with Orbiter 2006

Martin Schweiger

Centre for Medical Image Computing
Department of Computer Science
University College London, UK

martins@medphys.ucl.ac.uk

3rd International Workshop on Astrodynamics Tools and Techniques
ESTEC, Noordwijk, 4 October 2006

Contents

- Introduction
  - Orbiter overview
  - Orbiter 2006 - new features
- Simulation and physics engine
  - Dynamic state integration
  - Scenario editor
- Visualisation and graphics engine
  - Launcher and payload examples
  - Planetary surfaces
  - Interactive flightdeck simulation
- External trajectory data interface
  - Flight recording and playback
  - Playback of external data
- Orbiter demonstration

Introduction: Orbiter

- Real-time space flight simulation and visualisation on the PC
  - Under development for 6 years, latest version is 2006-P1 Edition.
  - Newtonian physics engine, numerical state integration including gravitational perturbation effects
  - Covers: atmospheric, suborbital, orbital, interplanetary flight
  - Demonstrate: launch, rendezvous/docking, re-entry, interplanetary transfers, gravity-assist, and more.
- Visualisation/demonstration tool
  - Interface to external trajectory data allows use of Orbiter as a visualisation tool, bypassing the internal physics engine
- Educational tool
  - Hands-on orbital mechanics demonstrator
- Development model:
  - Modular structure: core application provides physics and graphics engine
  - Extensive application programming interface (API) available for 3rd party addition of plugin modules (spacecraft, launch sites, celestial bodies, instrumentation, autopilots, remote control, networking, etc.)
  - An active development community has created an extensive collection of high-quality models of historic, hypothetical and fictional spacecraft.

Introduction - New features in Orbiter 2006

- Physics engine
  - Adaptive order of integration of linear and angular states (Runge-Kutta and symplectic integrators to order 8)
  - Perturbation model now includes gravity-gradient torque simulation
- User interface
  - Scenario editor for easy simulation setup
  - Instrumentation: "glass cockpit" and flight data display in external windows
- Visualisation and graphics engine
  - Support for higher-resolution planetary textures
  - Force vector visualisation
- External trajectory data interface
  - Support for simulation replay from Orbiter-recorded or external trajectory data
  - Includes animations and annotations

Topic: Physics engine

Dynamic state integration improved in Orbiter 2006

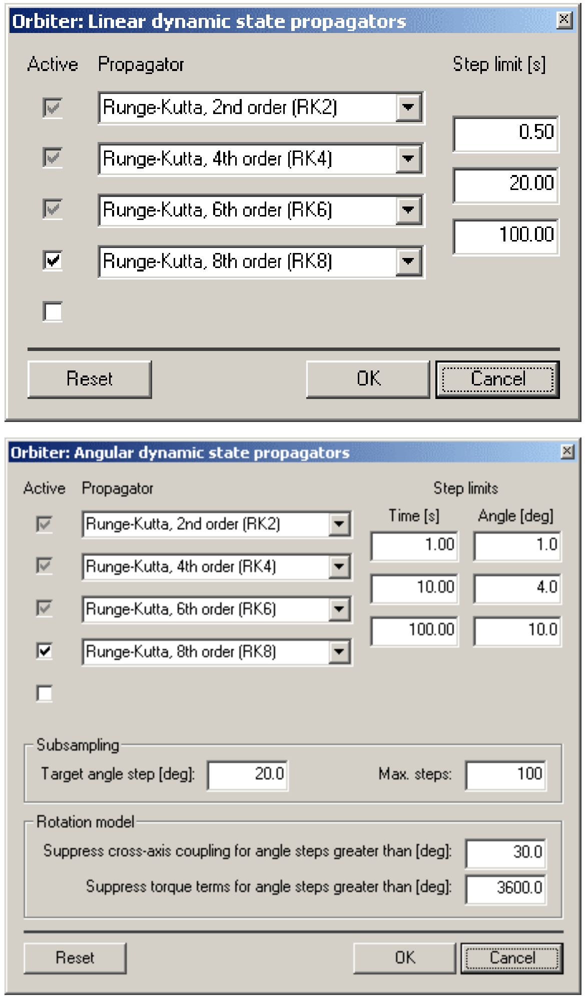
Dynamic state propagation: Integrators

Linear state propagation

- Adaptive steplength-dependent integration order provides accurate dynamic state propagation over a wide range of simulation speeds.
- Available user-definable integrators: *Runge-Kutta* and *symplectic* up to order 8
- Sub-sampling and propagation of perturbations (Encke's method) provide stability at very large time steps.

Angular state propagation

- Integration of Euler's equation of angular motion using RK integrator up to order 8.
- Adaptive and user-definable integration rules and sub-sampling depending on angular velocity



Orbiter linear and angular propagation parameter selection.

Dynamic state propagation: Integrators

Computational complexity of the integrators available in Orbiter.

Runge-Kutta

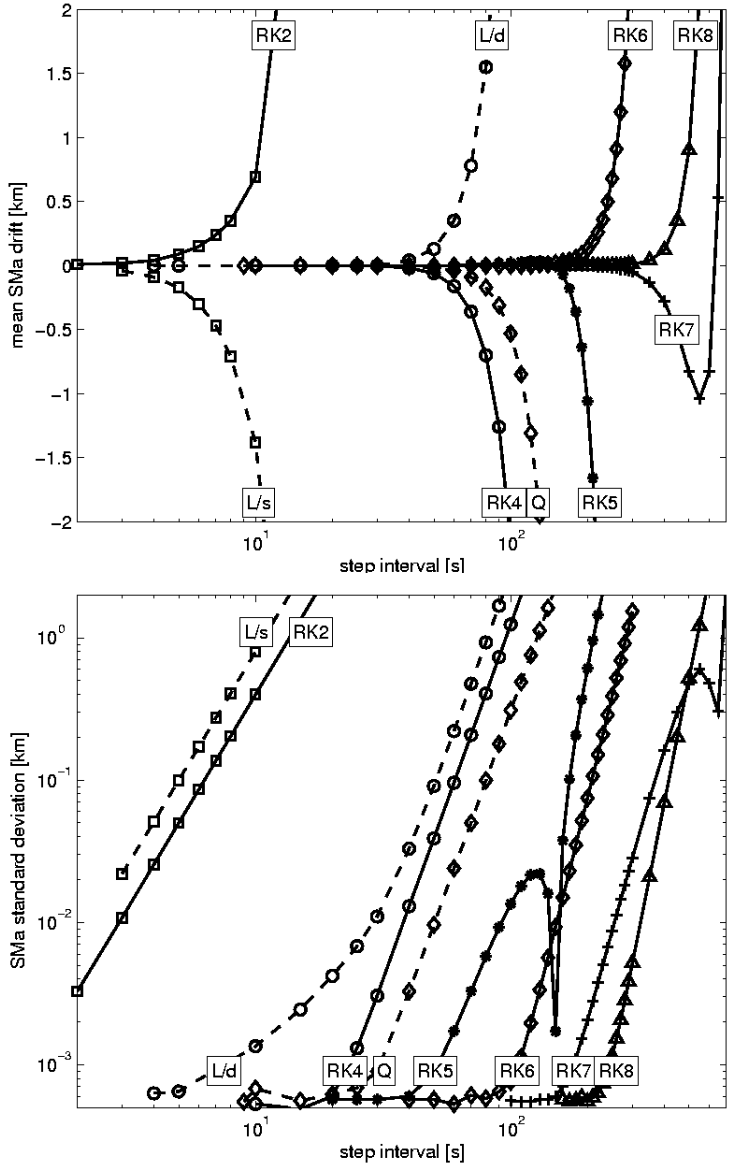
method	stages	timing [µs]
RK2	2	9.7
RK3	3	14.8
RK4	4	16.2
RK5	6	30.5
RK6	8	38.0
RK7	11	49.1
RK8	13	57.8

Symplectic

method	stages	timing [µs]
SY2	2	10.1
SY4	4	20.2
SY6	8	32.3
SY8	16	51.5

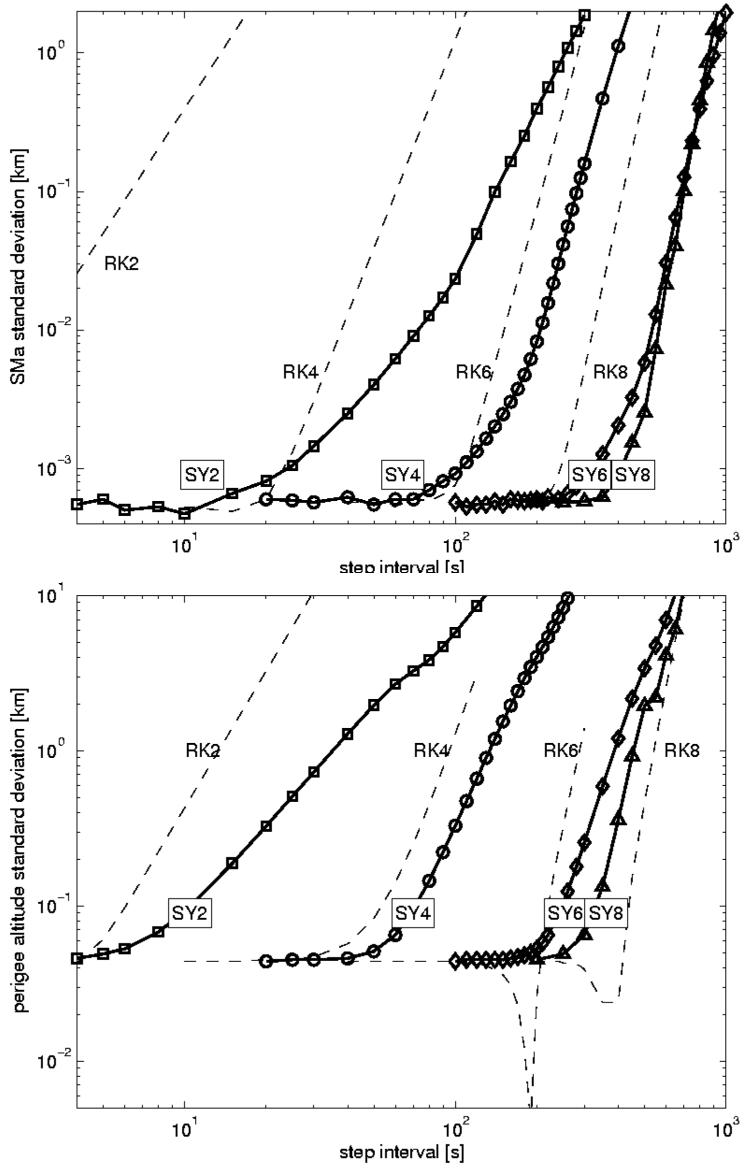
Dynamic state propagation: stability

- Long-term orbit stability with RK integrators
  - Mean drift (top) and standard deviation (bottom) of the semi-major axis for a low Earth orbit (mean altitude 217km) over a period of 10 days, as a function of sampling step length.
  - Shown are different orders of the RK family of integrators available in Orbiter.



Dynamic state propagation: stability

- Comparison between RK and symplectic integrators
  - Standard deviation in semi-major axis (top) and perigee altitude (bottom) of a low Earth orbit over a 10-day period as a function of sampling step length.
  - Shown is the family of symplectic integrators available in Orbiter.
  - For comparison, RK results are shown as dashed lines.



Dynamic state propagation: Perturbations

- Secondary gravity sources
  - Dynamic inclusion of gravity sources from multiple solar system objects (allows e.g. simulation of Lagrange point orbits)

$$U(\mathbf{r}) = \sum_{n}^{N} \frac{GM_n}{|\mathbf{r} - \mathbf{r}_n|} \quad \forall n : \frac{GM_n}{|\mathbf{r} - \mathbf{r}_n|} > U_0$$

Superposition of gravitational potential contributions for given threshold U_0

- Nonspherical gravity sources
  - Spherical harmonics expansion of deformation of planetary gravitational fields due to oblateness
  - allows simulation of propagation of nodes (e.g. sun-synchronous orbits)

$$U(r,\phi) = \frac{GM}{r}\left[1 - \sum_{n=2}^{N} J_n \left(\frac{R}{r}\right)^n P_n(\sin\phi)\right]$$

Perturbations of gravitational potential U, expressed in spherical harmonics with coefficients J_n

- Gravity-gradient torque
  - torque on objects with anisotropic inertia tensors due to inhomogeneous gravitational field
  - allows simulation of resonant oscillations or tidal locking

$$\boldsymbol{\tau}_G = \frac{3GM}{|\mathbf{r}|^3}\left[(\mathbf{L}\hat{\mathbf{r}}) \times \hat{\mathbf{r}}\right]$$

Gravity-gradient induced torque $\boldsymbol{\tau}_G$ at $\mathbf{r}$, given inertia tensor $\mathbf{L}$

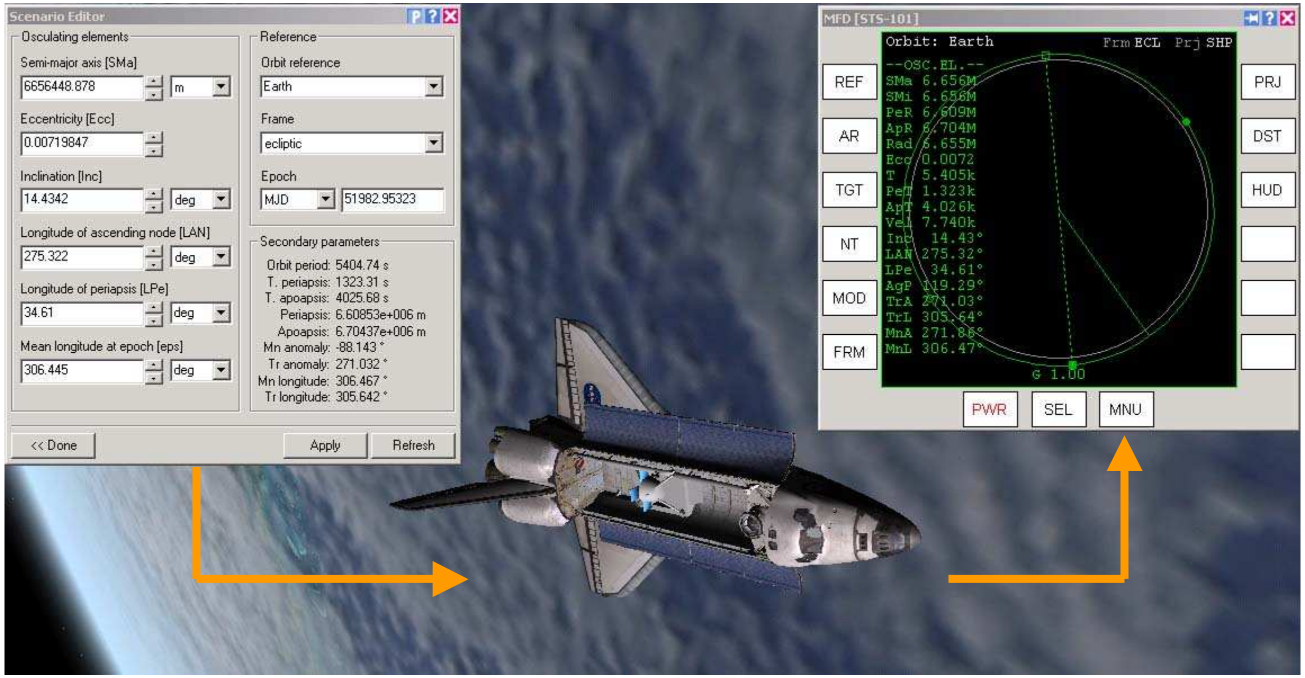
- User-defined perturbations
  - Examples: radiation pressure (orbit perturbation, solar sail simulation, etc.)

Topic: Simulation setup

Scenario editor for interactive spacecraft configuration

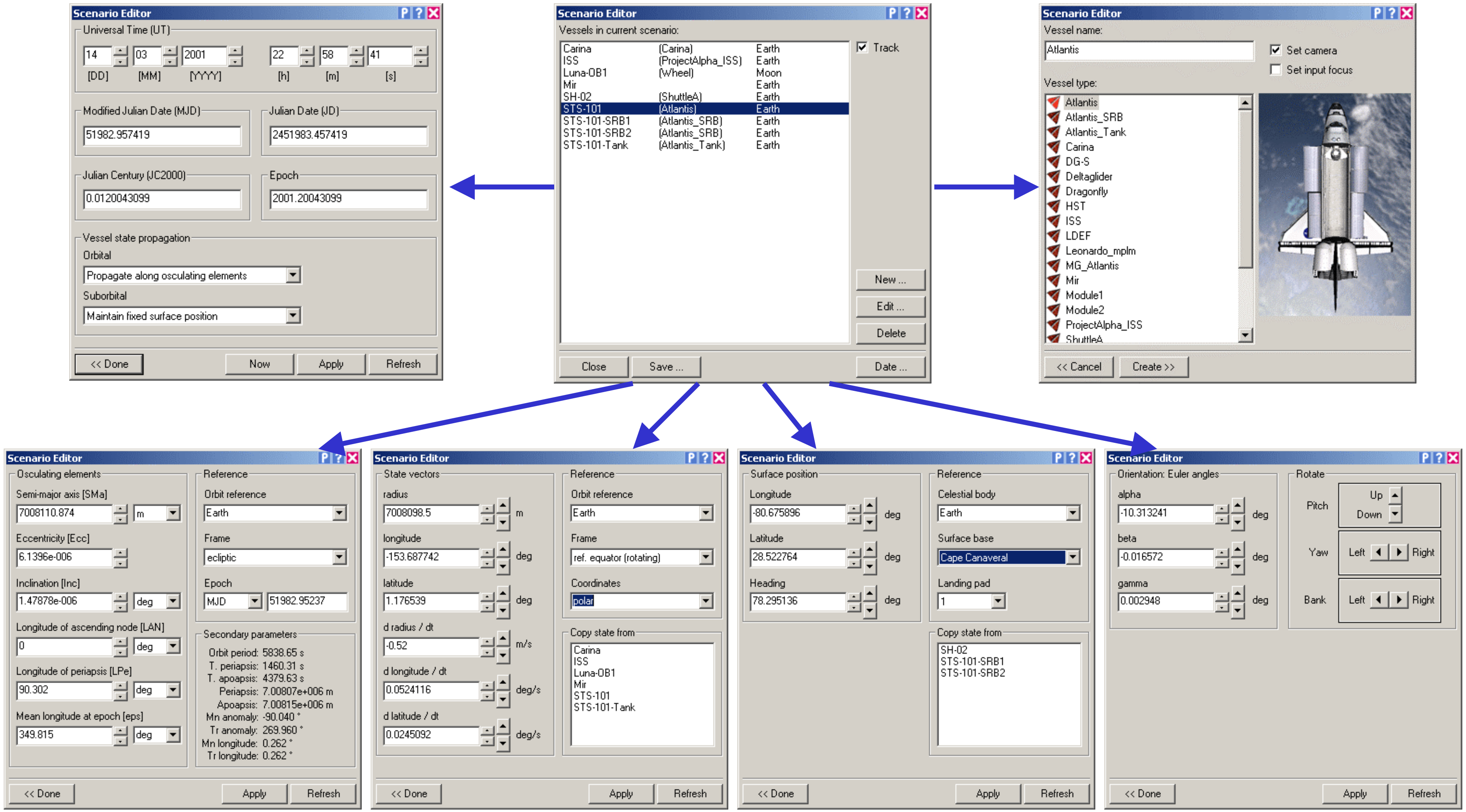
Simulation setup: Scenario editor

- Interactive configuration of spacecraft parameters
  - Orbital elements and state vectors
  - Orientation and angular velocity
  - Surface location
  - Composite structures/docking
  - Propellant status, vessel-specific parameters
  - Simulation date propagation



Simulation setup: Scenario editor

date setup — Scenario inventory — creation



...

orbital elements — state vectors — ground location — attitude

Topic: Visualisation

Spacecraft and launch site models: examples

Visualisation examples: Custom launchers

- Launchers and payload can be added to the simulation using custom meshes.
- Engine thrust, ascent behaviour, staging etc. can be defined via plugin modules.
- European launcher examples:
  - Ariane 1

Ariane 1 model by José Manuel García Estévez

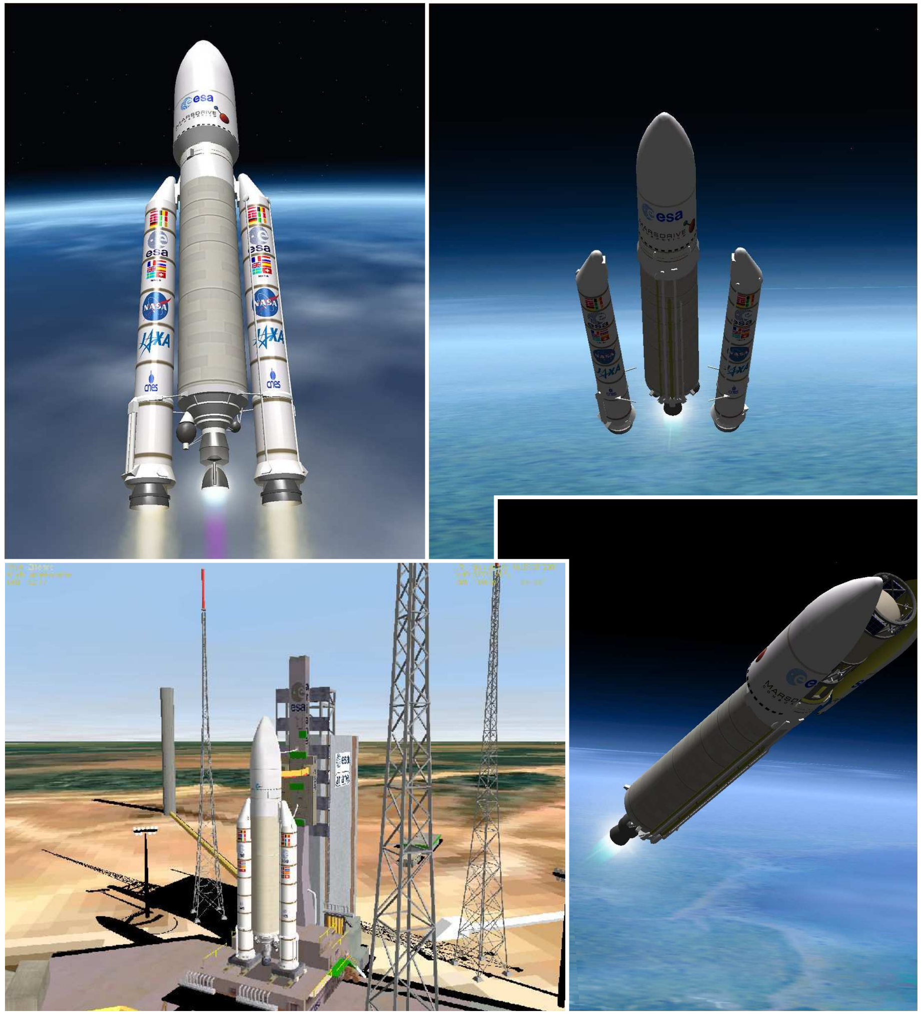
Visualisation examples: Custom launchers

- Launchers and payload can be added to the simulation using custom meshes.
- Engine thrust, ascent behaviour, staging etc. can be defined via plugin modules.
- European launcher examples:
  - Ariane 1
  - Ariane 4

Ariane 4 model by Pierre Refoubelet, Frédéric Servian, Christophe Etienne, Stéphane Colombain

Visualisation examples: Custom launchers

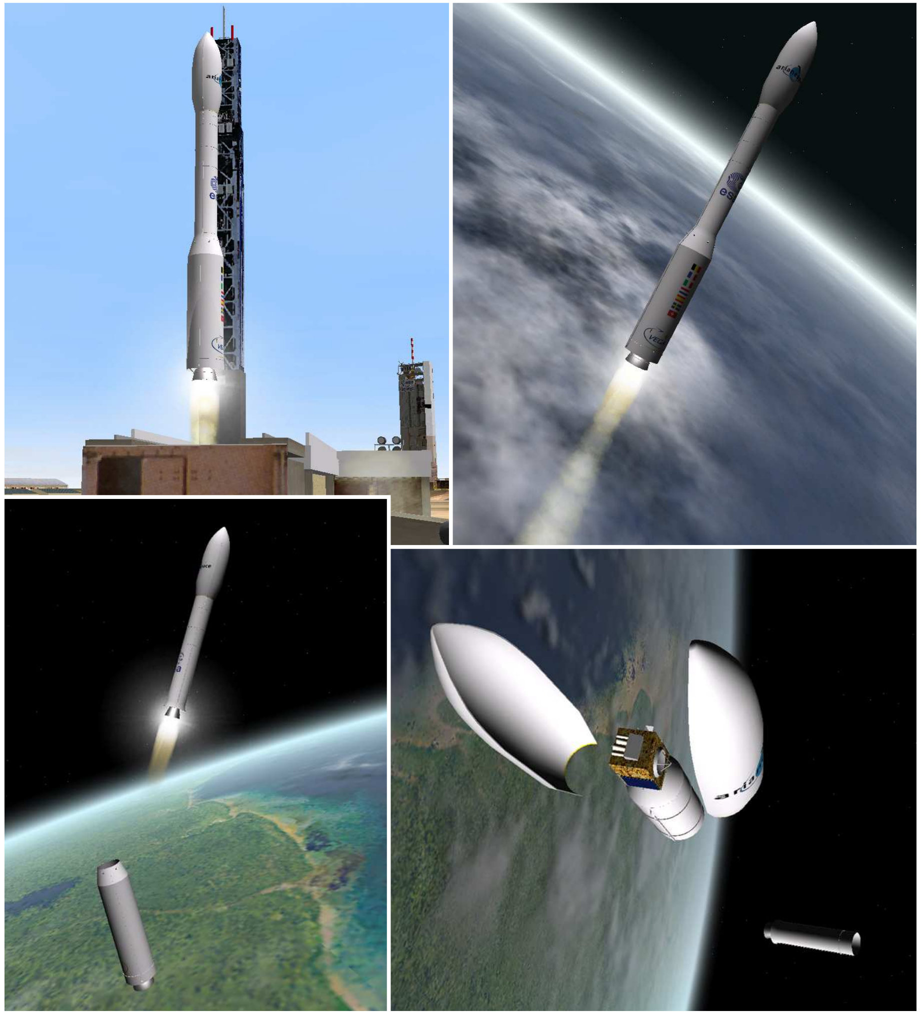
- Launchers and payload can be added to the simulation using custom meshes.
- Engine thrust, ascent behaviour, staging etc. can be defined via plugin modules.
- European launcher examples:
  - Ariane 1
  - Ariane 4
  - Ariane 5



Ariane 5 model by Thomas Ruth, with modifications by Andy McSorley

Visualisation examples: Custom launchers

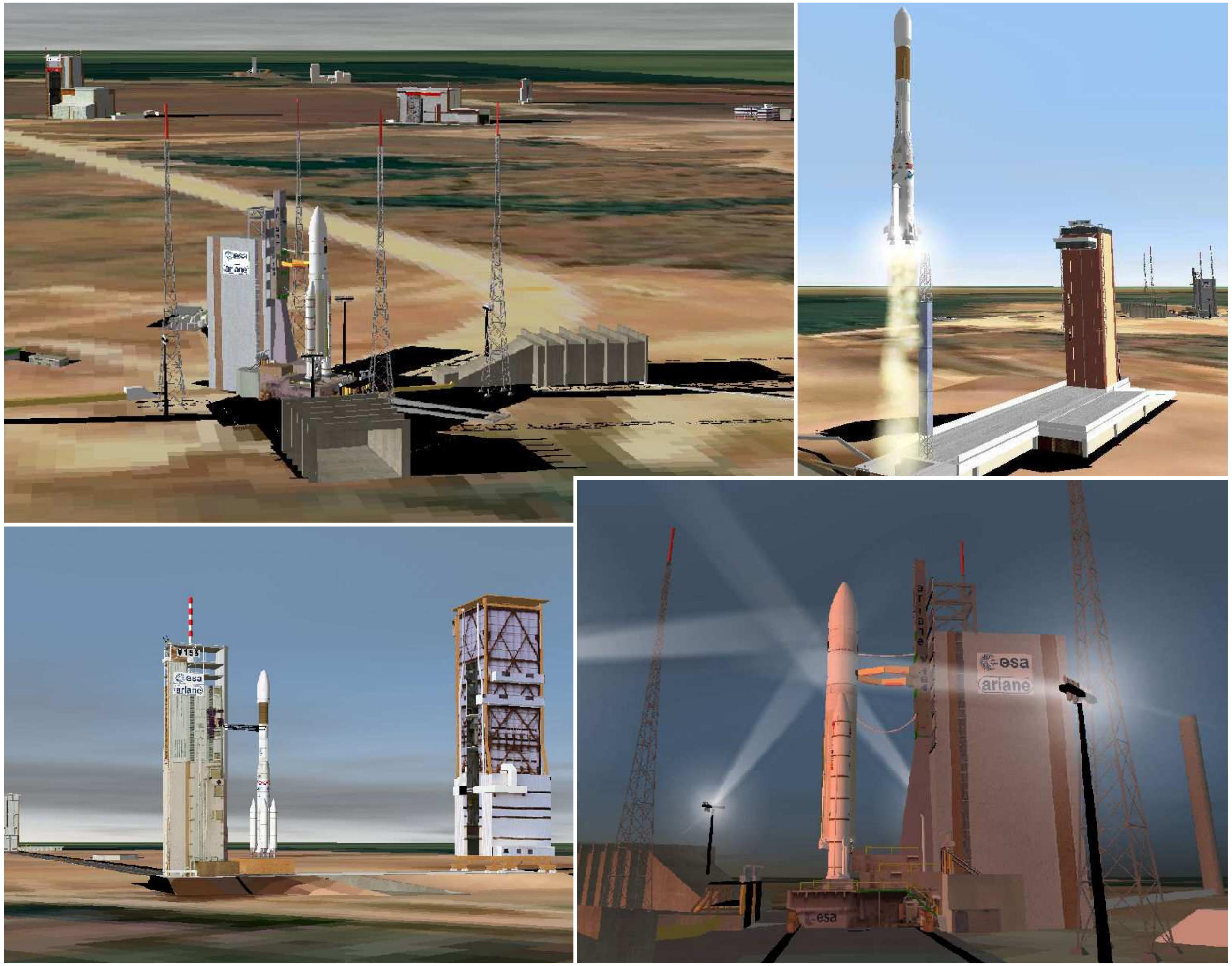
- Launchers and payload can be added to the simulation using custom meshes.
- Engine thrust, ascent behaviour, staging etc. can be defined via plugin modules.
- European launcher examples:
  - Ariane 1
  - Ariane 4
  - Ariane 5
  - VEGA



Vega model by José Manuel García Estévez

Visualisation examples: Ground structures

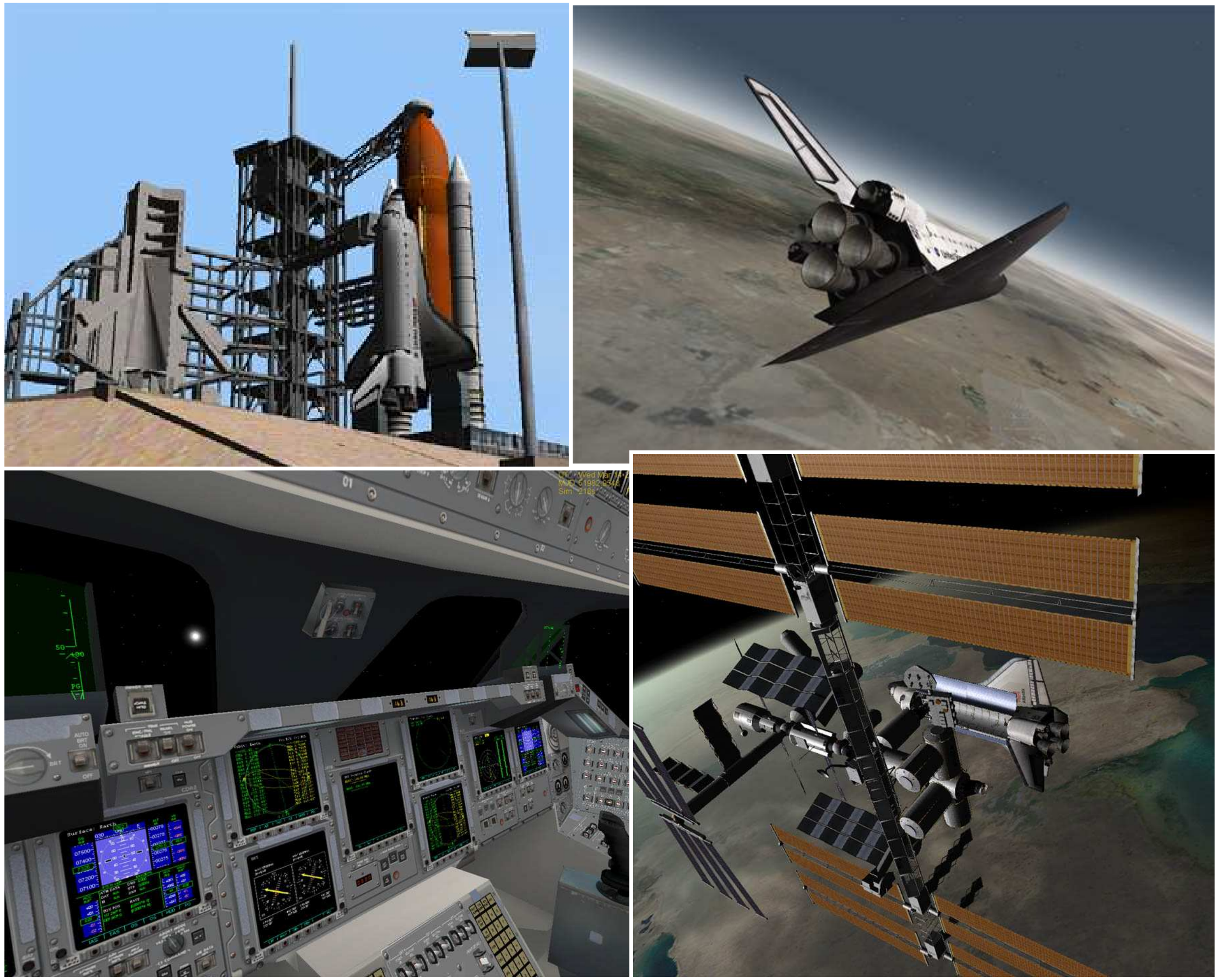
- Custom ground structures for launch sites can be added to the simulation.
- Example: Kourou
  - ELA1
  - ELA2
  - ELA3



Kourou site by Pierre Refoubelet, Frédéric Servian, Christophe Etienne, Stéphane Colombain

Visualisation examples: Space Shuttle

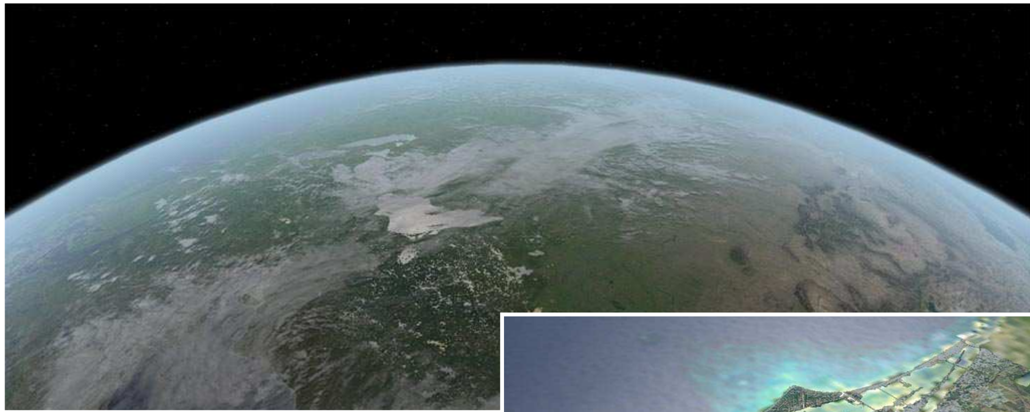
- Manned spacecraft: Modelling of flight deck interior ("virtual cockpit")
- Interactive manipulation of flight controls/ instrumentation
- Example:
  - Space Shuttle Atlantis

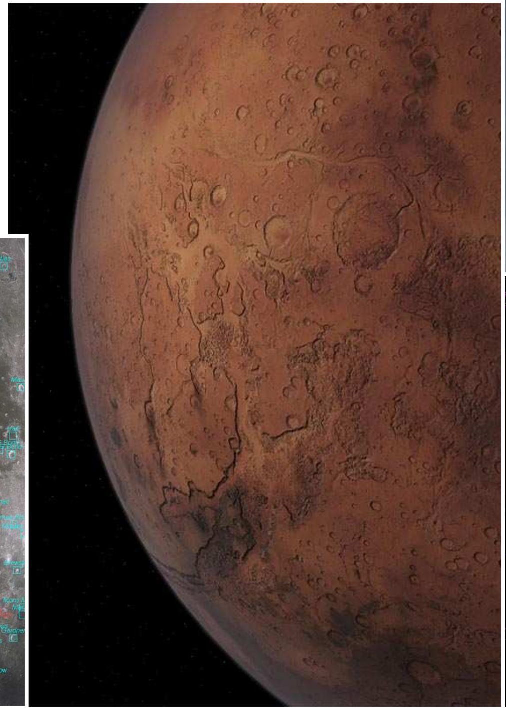


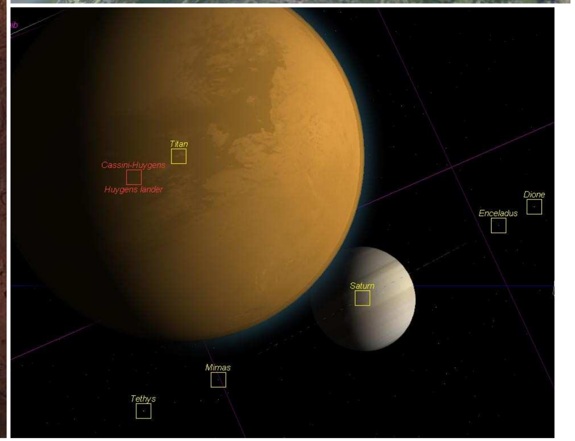
Atlantis model by Michael Grosberg, with extensions by Don Gallagher

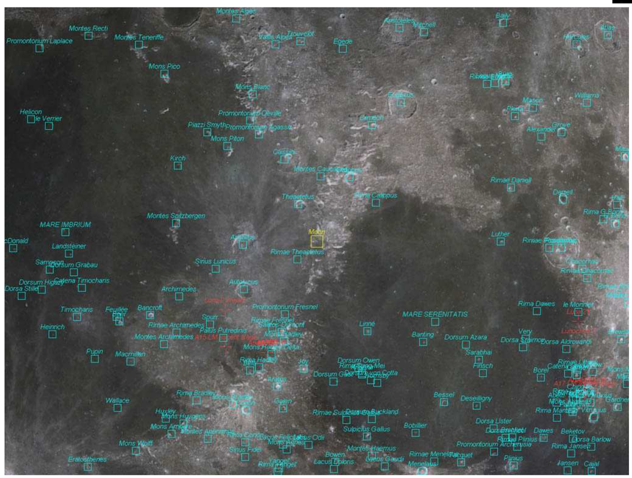
Visualisation examples: Planetary surfaces

- Celestial body surfaces:
- adaptive resolution as a function of apparent size up to 32k x 16k (equiv. 1.2km for Earth)
- support for local high-resolution textures (e.g. launch sites)
- support for specular reflections from water surfaces, cloud layers, atmospheric haze and city lights.
- support for celestial and surface labels and markers







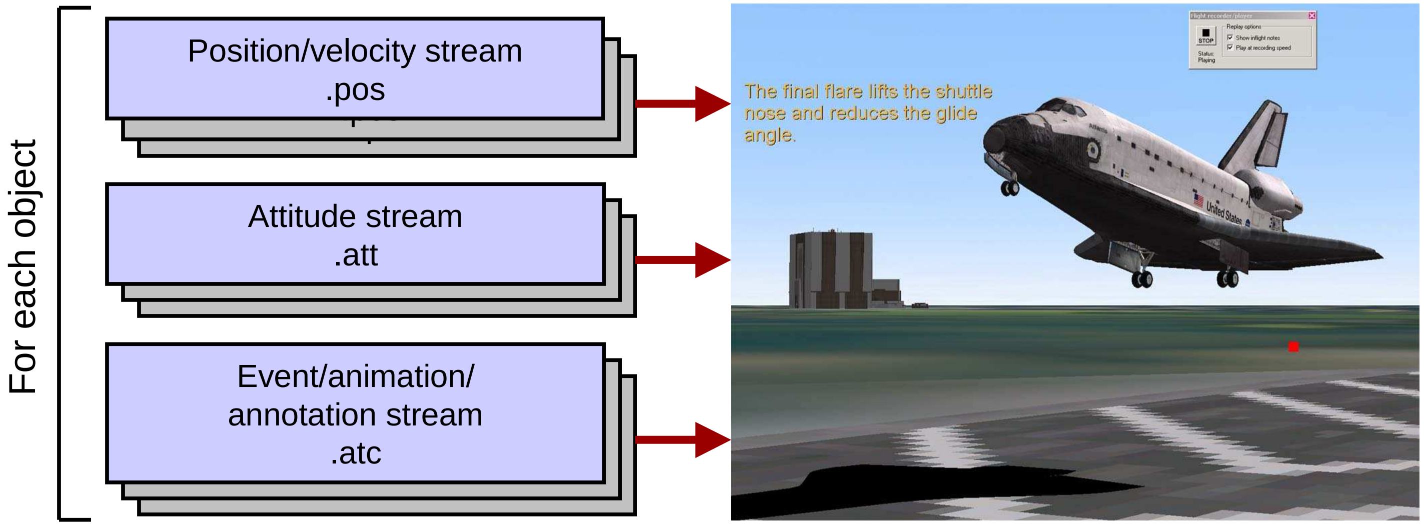


Topic: Flight recording and playback

Visualisation of externally provided trajectory data

Playback from external trajectory data

- **Data format**
  - Sampled position and velocity data (ecliptic or equatorial reference)
  - Sampled attitude data (ecliptic or local horizon reference)
  - Articulation data (engine and animation events, staging, booster separation, onscreen annotations, playback speed, etc.)



- **Replay mode**
  - Mixture of playback-controlled and active spacecraft is possible
  - User- or datastream-controlled playback speed
  - User-controlled camera

Playback from external trajectory data

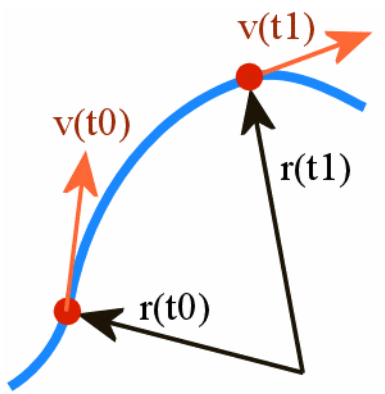
- Data interpolation
  - C2-continuous interpolation: piecewise linear acceleration
  - Given state samples $r_0=r(t_0)$, $r_1=r(t_1)$ and $v_0=v(t_0)$, $v_1=v(t_1)$ at consecutive sampling times t_0, t_1, the acceleration satisfies

$$a(t) = a_0 + b\Delta t, \quad t_0 \le t \le t_1, \quad \Delta t = t - t_0$$

  - Equations of motion: Integration of state vectors leads to

$$v(t) = \int_0^{\Delta t} a(t')dt' = v_0 + a_0\Delta t + \frac{1}{2}b\Delta t^2$$

$$r(t) = \int_0^{\Delta t} v(t')dt' = r_0 + v_0\Delta t + \frac{1}{2}a_0\Delta t^2 + \frac{1}{6}b\Delta t^3$$

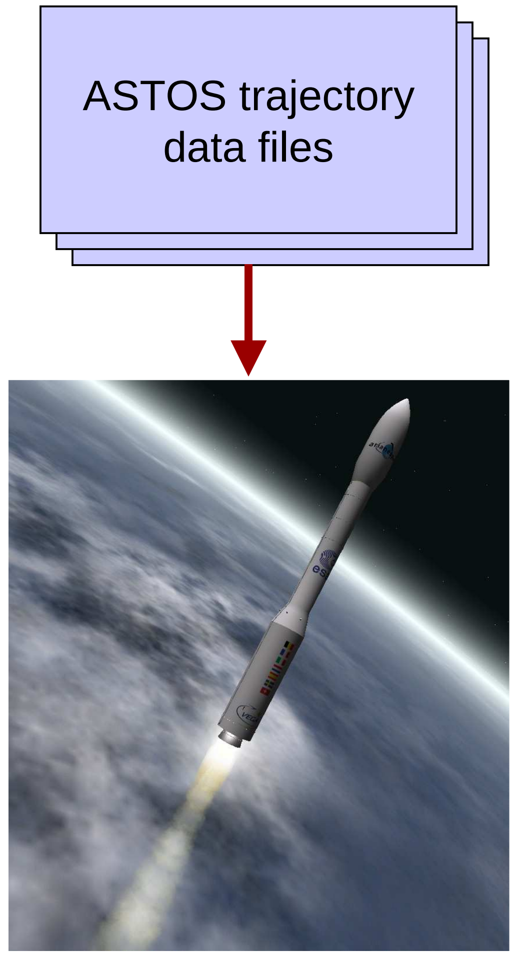


  - resulting in parameters

$$\left.\begin{aligned} a_0 &= \frac{2[3(r_1 - r_0) - \Delta T(2v_0 + v_1)]}{\Delta T^2} \\ b &= \frac{6[2(r_0 - r_1) + \Delta T(v_0 + v_1)]}{\Delta T^3} \end{aligned}\right\} \quad \Delta T = t_1 - t_0$$

Playback from external trajectory data

- Example: Interface to ASTOS trajectory data
  - The Orbiter playback interface was designed to accept data from the ASTOS aerospace trajectory optimisation software.
  - The ASTOS position/velocity and attitude data samples can be used as playback input streams for Orbiter.
  - Additional spacecraft-specific events (stageing, animations) and onscreen annotations can be added via the articulation stream to create complete launch demonstrations.
  - This allows to use Orbiter as a visualisation tool or demonstrator for ASTOS trajectory data.
  - Example: VEGA launch vehicle: launch, orbital insertion and payload deployment.



Summary

- Orbiter is a modular customisable real-time simulation and visualisation tool for spacecraft operation.
- Programming interface supports data exchange between Orbiter core and 3rd party addon modules.
- Versatile: simulation of historic missions or hypothetical concepts; "virtual prototyping"
- Built-in physics engine: dynamic propagation of linear and angular state vectors over a wide range of sampling intervals, including various perturbation sources.
- User interface: fast setup of spacecraft parameters via scenario editor; real-time simulation of flight instrumentation, immersive simulation of manned missions: "virtual cockpits".
- Support for mission playback from recorded or externally provided trajectory data, for demonstration and visualisation.

Resources and acknowledgements

- Orbiter main site and addon repositories:
  - orbit.medphys.ucl.ac.uk (Orbiter main site and core download)
  - www.orbithangar.com (Orbiter addon repository)
  - www.avsim.com (includes Orbiter addon repository)
  - users.swing.be/vinka (spacecraft wrapper dll for rapid prototyping)
- Educational resources:
  - "Go Play In Space" e-book by Bruce Irving, available at: www.orbiter.migman.com/orbiter.htm
  - Resources for educators, maintained by Jean-Marc Perreault: www.orbiterschool.com

I would like to thank the following authors for contributing addon models to Orbiter presented here:

Pierre Refoubelet, Frédéric Servian, Christophe Etienne, Stéphane Colombain (Ariane 1+4 models and Kourou site)

Thomas Ruth and Andy McSorley (Ariane 5 model)

José Manuel García Estévez, supported by Hispaseti.org and Astoseti.org (Vega model)

Michael Grosberg and Don Gallagher (Atlantis model)

Seth Hollingsead, Rolf Keibel and others (planetary textures)

Some images were taken from Bruce Irving's web site.

Thanks also to all other authors and contributors, in particular the beta test team for their input.