Main differences between V4.10 and V4.9

PATRIUS V4.10 is a major release adding some new features and correcting some bugs. PATRIUS is now a single-module project.

New functionalities

- Ionospheric and tropospheric correction models can now describe derivable parameters
- New class PVEphemeris describing tabulated ephemerides
- PVCoordinatesProvider is now serializable
- New class AberrationCorrection for taking into account the light-time aberration correction, the stellar one (new class StellarAberration), both of them or none of them
- New class DatePolynomialChebyshevFunction representing a Chebyshev polynomial function of date
- The PyramidalField class can now determine whether a field of view is convex and return a complementary field of view, an intersection of a union of fields of view
- AbstractCelestialBody, UserCelestialBody and JPLCelestialBody can be built by means of an AbstractAttractionModel (instead of a simple GM)
- A TopocentricFrame can now be created by defining the origin of the frame with a Vector3D and its zenith
- New class CelestialBodyFrame representing a frame centered on a central body
- New interfaces CelestialBodyEphemeris and CelestialBodyEphemerisLoader to decouple the ephemeris part from the attached body one

Bugs fixes

- The methods of the class Line take now correctly into account the concept of minimum abscissa
- VisibilityFromStationDetector take now correctly into account the masking by celestial bodies
- ExtremaElevationDetector take now correctly into account the signal propagation
- StationToSatMutualVisibilityDetector and SatToSatMutualVisibilityDetector take now correctly into account the direction of the signal propagation

Main differences between V4.9.1 and V4.8

PATRIUS V4.9.1 is a major release adding some new features and correcting some bugs.

New functionalities

- ICRF is now the root frame for all frames (formerly the GCRF) (no impact for users)
- New frames have been added to CelestialBody classes splitting IAU data in constant, secular and harmonic parts (icrf, inertial equator, mean equator, true equator, constant rotating, mean rotating, true rotating).
- An object CelestialBody has now a GeometricBodyShape linked to it
- FacetCelestialBody has been renamed in FacetBodyShape. Dependency to CelestialBody has been removed. use userCelestialBody with a FacetBodyShape instead.
- PVCoordinatesProvider.getNativeFrame method has been added. it returns the "native frame" of the PVCoordinatesProvider
- Convention used for Tchebychev polynomials is now the SPICE convention
- New class NotCriterion returning the complementary of a phenomenon
- A frame can define its own referential through method Frame.setReferential()
- Intersection computation on celestial bodies now returns only the "front" intersection with line of sight. i.e. intersections behind the line of sight are not returned.
- A new detector ExtremaGenericDetector has been added. It detects the extremum (through cancellation of derivative) of un underlying detector. It can then be used recursively to detect nth order derivatives of a detector.

Bugs fixes

- The class OrCriterion has been fixed
- Frame change in class QuaternionPolynomialProfile is properly taken into account

Main differences between V4.8.1 and V4.7

PATRIUS V4.8.1 is a major release adding some new features and correcting some bugs.

New functionalities

- Solar Radiation Pressure classes (SolarRadiationPressureCircular and SolarRadiationPressureEllipsoid) have been generalized:
  - They can take into account or not eclipses
  - The occulting body can be any body
  - There can be as many occulting bodies as required
- Marini-Murray tropospheric model has been added (MariniMurrayModel class)
- Add of synodic frame through class SynodicFrame (replacement of the former class)
- Several classes have been added to handle Chebyshev polynomials:
  - PolynomialChebyshevFunction for building and manipulation of such polynomials
  - ChebyshevDecompositionEngine for getting an approximation of function as a sum of Chebyshev polynomials

Note: The parameters order of the class LinearFunction is reverse in its constructors and getter. That new behavior is more consistent with the others functions. Please be careful to reverse the parameters order in any use of this function.

Bugs fixes

- Correction of matrix multiplication for non-square matrices in some specific cases
- Correction of sub-matrix extraction in DecomposedSymmetricMatrix
- Correction of azimut computation through method computeBearing of class ProjectionEllipsoidUtils
- Correction of facets orientation produced by method toObj of class GeodeticMeshLoader
- Correction of evaluation of the distance between a sensor and a body (used in particular for detectors such as StationToSatMutualVisibility)

Main differences between V4.7 and V4.6.1

PATRIUS V4.7 is a major release adding some new features and correcting some bugs.

New functionalities

- Revision of attitude sequences, now to be used through StrictAttitudeLegsSequence
- Add grid attraction model to be used with small bodies such as Phobos
- Add computation of derivative of IAU pole, used when computing transformation from/to body inertial/rotating frames.
- Add harmonics of third body potential in ThirdBodyAttraction force model.
- Add handling of spherical coordinates SphericalCoordinates
- Generalisation of TDB time scale, which can now be changed by users

Bugs fixes

- Correction of numerical propagation in order to reach exactly the required end date (rare case)
- Correction of a thread-safety problem when using analytical 2D propagation
- Correction of pole tides displacement computation (colatitude instead of latitude)

Main differences between V4.6.1 and V4.5.1

PATRIUS V4.6.1 is a major release adding some new features and correcting some bugs.

New functionalities

- Add Mean <=> Osculating 2D analytical conversion through classes Analytical2DPropagator and Analytical2DOrbitModel.
- Add model DTM-2012 through class DTM2012
- Add celestial body defined by facets such as Phobos: FacetCelestialBody. This class efficiently performs some standard computations on the body (intersection, eclipse computation, etc.)
- Add STELA Moon-Sun model up to order 8
- Add Synodic frame through class SynodicFrame
- Add 2nd order Cowell integrator through class CowellIntegrator. This integrator can be used with PATRIUS numerical propagator and possesses same features as PATRIUS 1st order integrators
- Add pole tides force through class PoleTides

Bugs fixes

- Correction of SP3 reader (velocity only)
- Correction of TLE reading when B* is of the form 1E-x

Main differences between V4.5.1 and V4.4

PATRIUS V4.5.1 is a major release adding some new features and correcting some bugs.

New functionalities

- Add Mean <=> Osculating Liu conversion
- Add fast sin and cos computation in one single time
- Add handling of partial derivatives equations in multi-satellites propagation
- Improve computation times (orbit propagation mainly) when using Assemblies, drag force and rediffused PRS
- Add handling of signal propagation delay in event detections

Bugs fixes

- Correction of Meeus board model
- Correction of yaw compensation attitude law when used with a non-geocentric frame
- Correction of MSIS model (temperature below 72.5km)
- Correction of TLE handling of classification char
- Correction of redifused radiative model when used in conjonction with a non-spherical satellite
- Correction of MassModel mass update when used with several maneuvers

Main differences between V4.4 and V4.3

PATRIUS V4.4 is a major release adding some new features and correcting some bugs.

New functionalities

- Add possibility to provide mobile parts to Assembly
- Add maneuvers on orbital parameters (da, de, di)
- Add DV computation during propagation

Bugs fixes

- Correction of event detection on 1ULP propagation duration: it is now more robust
- Correction of shiftedBy method of AbsoluteDate: it is now more accurate
- Correction of azimut mask handling if 0 not included in mask

Main differences between V4.3 and V4.2

PATRIUS V4.3 is a major release adding some new features and correcting some bugs.

New functionalities

- Use of Java 8
- New low-level math frameworks: Math, StrictMath and the fastest library available for each math function
- JAFAMA dependency is now 2.3.1
- Add a possibility to by-pass min step criterion on adaptive stepsize integrators such as DOP853
- Body packages has been refactored (ExtendedOneAxisEllipsoid)

Bugs fixes

- Correction of "Remove detector" feature when propagating several times with the same propagator
- Correction of third body ephemeris through the proper handling of IAU data (TDB scale)
- Correction of event detection: it is now more robust
- Optimisation of SpacecraftState handling of attitude for increased performances
- Optimisation of KinematicsToolkit for increased performance
- Correction of angle normalization function

Main differences between PATRIUS 4.2 and 4.1.1

PATRIUS V4.2 is a major release adding some new features and correcting some bugs.

New functionalities

- Mecanism to handle and define low-level math frameworks
- JAFAMA 2.1.0 can now be used instead of FastMath for increased speed
- Add a MultiAttitudeProvider interface for handling of correlated attitudes when propagating several satellites
- Generalization and revision of attitude sequences
- Creation of a new package "orientation" for orientation providers similar to attitudes providers
- Add some new features to generic intervals and dates intervals such as intersection, merge, etc.

Bugs fixes

- Corrections to ensure propagation is properly performed exactly up to required user date
- Correction in class AzimuthElevationField to properly manage double-points
- Corrections in definition of celestial frames: the ICRF frame is now colinear to GCRF frame and all celestial bodies parent frame is now the ICRF.
- Correction of events detection: it is now more robust
- Correction of angular velocity of TNW frame with respect to other frame
- Correction of mass equations integration
- Correction of "Remove detector" feature when propagating several times with the same propagator.

Main differences between PATRIUS 4.1.1 and 4.1

Bugs fixes

- Corrections on the Vehicle class to be consistent with GENOPUS V2.0.
- Corrections on the ForceModelsData class to be consistent with GENOPUS V2.0.
- Adding some getters to be consistent with GENOPUS V2.0.

Main differences between PATRIUS 4.1 and 4.0

New functionalities

- New methods for Cartesian/spherical transformation Jacobian matrixes computation
- New RK6 interpolator (more precise one as its order is 2; error in position is about some centimeters on a LEO orbit rather than few hundreds of meters as previously).
- New class to collect environment models data (ForceModelsData).
- Access to potential coefficients.
- Rediffused PRS modified when reduction of the amount of sources (see JavaDoc).
- Glint position vector available (GlintApproximatePointingDirection class).
- Adding several classes and methods (as getters or clone/copy) from GENOPUS.
- New IntervalleOccurrenceDetector class

Bugs fixes

- Correction when using Eckstein-Hechler propagator constructor with an attitude law.
- Matrixes multiplications in the CovarianceUtils class are now well coded.
- TankProperty and PropulsiveProperty did not need the name of the part in the constructor.
- IsisSunPointing class: in the getAttitude() method, the xSun vector is normalized.
- Performance problems for multithreading due to a bad use of synchronization.
- Correction of the getDate() method in the GalileoAlmanacParameters class, considering GST epoch as 1999/08/22 0h0mn0s UTC.
- Correction of the equivalent maximal inertia in the AbstractIsisSpinBiasSlew class taking into account all the wheels.
- Anomaly when reading some TLE.
- Useless computations in the Rotation class constructor.
- Improvement of the HelmholtzPolynomial.computeHelmholtzPolynomial() method: Balmino model is 30% more rapid.
- SI value for hydrogen mass
- Error messages for MultiNumericalPropagator and NumericalPropagator classes have been modified.

API incompatibilities

- TankProperty and PropulsiveProperty constructors

Regressions

- When using RK6 integrator

Main differences between PATRIUS 4.0 and 3.4.1

Major release

PATRIUS V4.0 is a major release and has a simplified architecture: its former dependencies (Commons-Math, OREKIT and the addons) have been merged into a single standalone library (patrius_V4.0.jar).

Users only need to download one library for compilation purpose.

To help migrate sources based on PATRIUS 3.x, a script is available (search & replace all the import) and make it quite easy to update to PATRIUS 4.0. The script will transform all Java files found under the current directory.

New functionalities

- **PATRIUS is now a Java 7 project : all users need a JRE (or JDK) 7 (or higher).**
- ConstantThrustManeuver & VariableThrustManeuver have been merged in ContinuousThrustManeuver. Constructors use "PropulsiveProperty" object to set all parameters. Besides, the behavior and the numerical results are the same as before.

Bugs fixes

- AtmosphereData uses molar mass in kg/mol (and not in g/mol). GlobalAeroModel results are affected.
- Assembly can now correctly handle mobile parts
- STELA propagator can use osculating parameters for its initial orbit.
- There was a bug in ExtremaElevationDetector (max/min elevation was wrong)
- Some optimizations in :
  - AbsoluteDate : around 5% gain in a lot of algorithms
  - HelmoltPolynomial
  - GTOD frame : a big computation time issue has been fixed. All the frames of this branch now have the same performances as the IERS one.
- Normalized Earth potential coefficients can be used for order&degrees above 86 (numerical quality bug fix)

API incompatibilities

- EarthGravitationalModelFactory methods need a new parameter to allow missing coefficients.
- FramesConfigurationBuilder : the new method setEOP2000History() replaces setEOPHistory()
- TOD, VEIS, ITRF equinox based frames now need to be configured with EOP History, through a FramesConfiguration

Regressions

- GlobalAeroModel : some regression (about 10 cm after a 6h LEO propagation)
- Use of MultiNumericalPropagator & Assembly : small regression have been observed (~ 1e-14m)
- Cn/CtCookModel : these models have been fixed (former results were wrong)
- ContinuousThrustManeuver : some numerical regression (ie : 1e-7m in position) can be observed while propagating through a maneuver
- Earth potential : for high order/degree, there are some numerical regression (1e-7m after 24h numerical propagation of LEO orbit) but a better precision (see § bug fixes).

Main differences between PATRIUS 3.4.1 and 3.3

New functionalities

- Some improvement of ISIS Spin Bias Slew : user can now provide a final law, final date and a target initial law.
- New tank and propulsive properties
- New Vehicle class, that simplifies Assembly creation
- AeroModel can compute partial derivatives atmospheric drag vs position

Bugs fixes

- Synchronization of the Assembly mass during numerical propagation
- Bug fix in the event detection algorithm, for synchronous events
- Some optimization in CartesianOrbit, EOP2000HistoryConstantOutsideInterval and event detection.
- Two solar activity readers can be used at the same time
- Bug fix in local time

API incompatibilities

- Classes LocalTimeDetector , SolarTimeDetector and LocalTime : these classes and their methods have been renamed with “Angle” suffix (except computeEquationOfTime) and now return local hour angle in [-Pi ; Pi [
- SpinBiasSlew classes always take a “TypeOfDate” parameter, to choose between INITIAL and FINAL date to compute slews.
- The AERO_SPHERE property has been renamed in AERO_CROSS_SECTION
- DragCoefficientProvider interface : the method getCoefficients() now take an AtmosphereData parameter for temperature instead of a single double value
- The EarthGravitationalModelFactory now take the name of the file to read.
- MassEquation does not use any more a MassProperty
- AeroSphereProperty does not use any more the atmosScaleHeight parameter.

Regressions

- Some numerical regression when using a GlobalAeroModel
- Some small numerical regression when using the CookWallGasTemperature class
- NumericalPropagator now handles correctly interpolated mass when using various mass equations
- IAUPoleFactory corrections, with some impact for Pluto
- Some regression when adding a SymmetricMatrix with a non-symmetric one (due to a bug fix)
- Rotation class : some numerical regression when using getAngles(RotationOrder) for one axis rotation and symmetric sequences (ex : XYX, XZX, etc.).
- AeroModel : some improvement when computing partial derivatives of drag acceleration vs position
- Event detection algorithm may give slightly different results, in quite complex cases (ex : synchronous events), but will still respect the convergence threshold.

Known bugs or limitations

- When using the mass of an Assembly in a StepHandler, user must get it through the interpolator and not through the Assembly (mass may not be synchronized).
- Event detection : if there is a phenomena with a duration smaller than maxCheck, and and other event that reset state during the phenomena, then the beginning of phenomena will be detected at the resetState date.

Main differences between PATRIUS 3.3 and 3.2

New functionalities

- Attitude Law “ISIS Sun Pointing”
- ISIS Spin Bias Slew (AnalyticalIsisSpinBiasSlew and NumericalIsisSpinBianSlew)
- Model for atmospheric drag (GlobalAeroModel), that interpolates coefficient depending on two angles (azimut, elevation of the “flux” – atmosphere relative velocity – in the spacecraft frame)
- “Cook” Model for atmospheric drag coefficients (CnCookModel & CtCookModel), to be used with the new drag model
- MeeusSun can use “on board” Sun direction (a simplified Sun direction, that does not rely on binary ephemeris)
- MSIS2000 & DTM2000 atmosphere model can provide extended atmosphere data, such as partial densities of the main components of the atmosphere

Bugs fixes

- RFLinkBudget computation
- Performance fix for multi-threaded propagations
- Correction in the TwoSpinBiasSlew algorithm
- Optimization of drag & sun radiation pressure algorithms
- Optimization in event detection algorithm

API incompatibilities

- Classes SensorModel and SensorProperty use LocalRadiusProvider instead of double, to set the mainTargetRadius and inhibitionTargetRadiuses
- Class NumericalPropagator : the addMassProviderEquation has been renamed in setMassProviderEquation
- Class LocalTime : computeMeanLocalTime always compute local time in TIRF and does not need sideral time as a parameter any more
- The InertialProvider class does not exist anymore : the ConstantAttitudeLaw should be use instead
- RotationInterpolator & SlerpInterpolator interfaces have been suppressed : it’s recommended to use HermiteInterpolator
- TabulatedAttitude class : one constructor has been deleted
- LofOffset : two constructors have been deleted
- Attitude : estimateSpin() method does not exist any more. Users should call AngularCoordinates.estimateRate() instead.
- GroundPointingWrapper : the getUnderlyingAttitudeLaw should be used instead of getGroundPointingLaw
- SpinStabilized : the getUnderlyingAttitudeLaw should be used instead of getNonRotatingLaw

Regressions

- Distance between a line and an ellipsoid : minor numerical regression
- Numerical propagation with partial derivatives : some minor regressions may be observed if user was using more tighten tolerances.
- PatriusSolarRadiationPressure : corrections imply 1e-12 relative error after a day of propagation
- Numerical propagation with continuous maneuver & drag : correction have been made in the way mass was handled. This can cause small differences : 1e-12 relative error after a day of propagation on our test cases (only when continuous maneuver are used !)
- Partial derivatives of Sun radiation pressure according to k0 : old algorithm was wrong and some differences will be observed
- RF Link Budget : the old algorithm was detecting events even when the station was “behind” the antenna.

Known bugs or limitations

- Nothing worth to mention.