Quartz

Airbus Defence and Space flight dynamics software, based on Orekit

Orekit Day 2nd edition, Darmstadt, Germany – May 23rd, 2019

DEFENCE AND SPACE

Yannick JEANDROZ - *Quartz Java Product Owner* - yannick.jeandroz@airbus.com

AIRBUS

Space Systems – Our Activities

Telecommunication Satellites

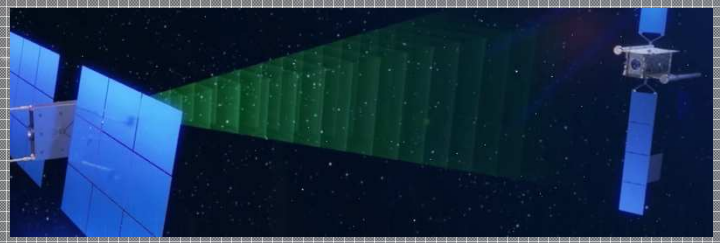
Earth Observation Systems

Space Equipment

Human Spaceflight

Space Exploration & Science

Navigation Solutions



On-Orbit Services

Ground Segments

Launchers

… and a complete range of Space-based Services in our "Communications, Intelligence and Security (CIS)" business line.

AIRBUS

Flight Dynamics Team presentation

- Flight Dynamics = related to satellite orbit (and not satellite attitude)
  - 25 people
  - Incorporated in AOCS & GNC division (TESOA)

- Covering Flight Dynamics activities:
  - From phase 0/A to phase E
  - For both Telecom and Earth Observation
  - Telecom programmes experience since 1995 (GTO and GEO mission analysis & FDS Quartz family)
    - Earth Observation programmes experience since 2000 (mission analysis), and 2005 (FDS Quartz LEO family start)

- 4 Major Activities
  - R&D activities
  - Mission Analysis (Launch and Early Operations Phase, Station-Keeping, End of Life)
  - QUARTZ Product (Airbus DS Operational Flight Dynamics Software (FDS)) + New Electrical Orbit Raising tools
  - Operations during LEOP (fully responsible of Flight Dynamics activities for LEOP preparation and execution)

Scope of the Quartz FDS

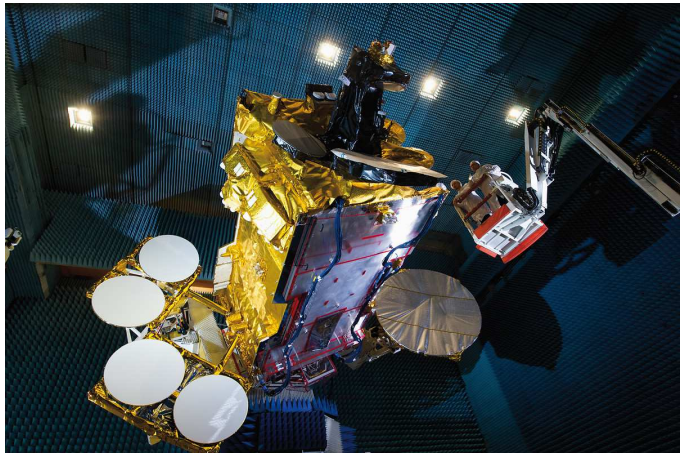
- Covers all flight dynamics needs for LEOP and Station Keeping
- Developed in-house by flight dynamics engineers with strong operational experience in GEO and LEO
- Used by Airbus and external customers
- Support for all Airbus satellite platforms

Mission profiles

- Transfer from:
  - Low Earth Orbits (LEO)
  - Geostationnary Transfer Orbits (GTO)
  - Supersynchronous Transfer Orbits (SSTO)
  - Sub-synchronous Transfer Orbits

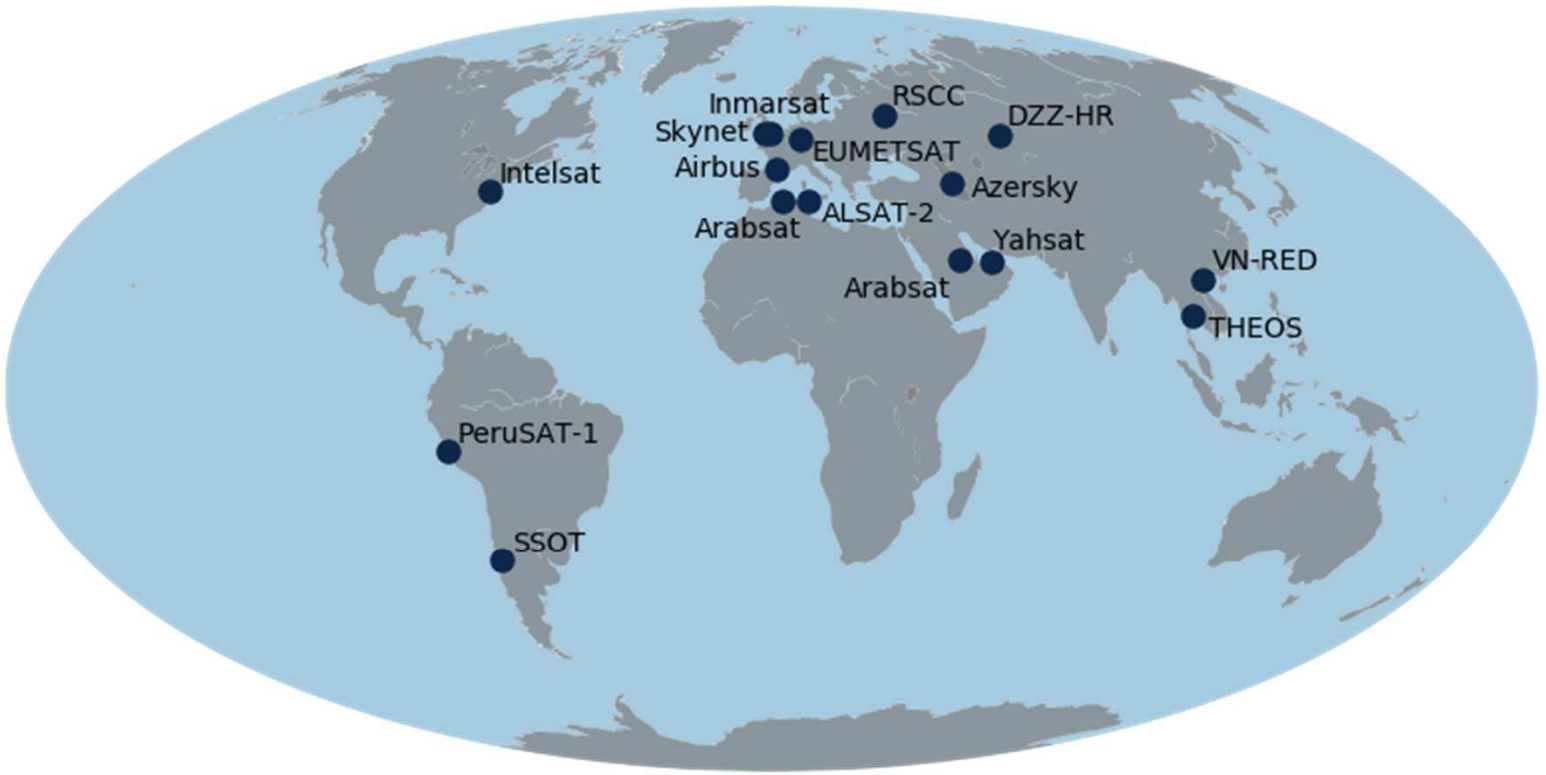
- Station-keeping for:
  - Sun-synchronous orbits (SSO)
  - Geostationnary orbits (GEO)

- Propulsion types:
  - Chemical
  - Electrical



Main features

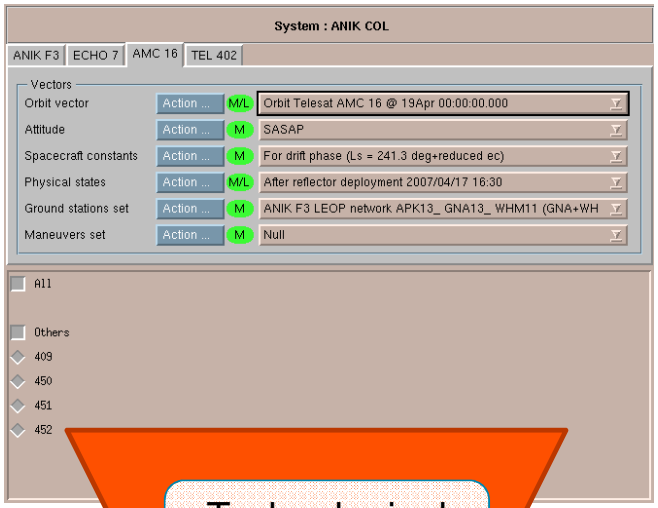
Many customers worldwide

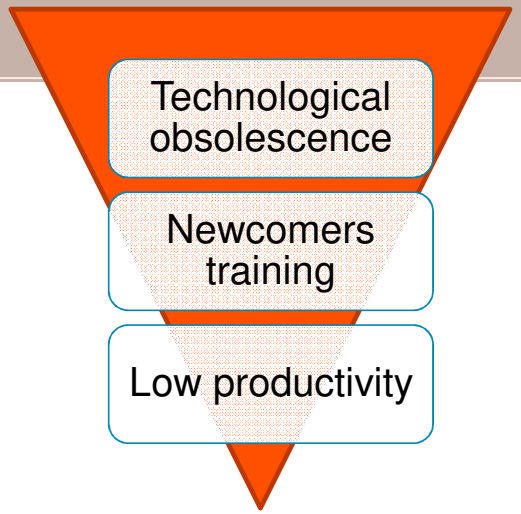


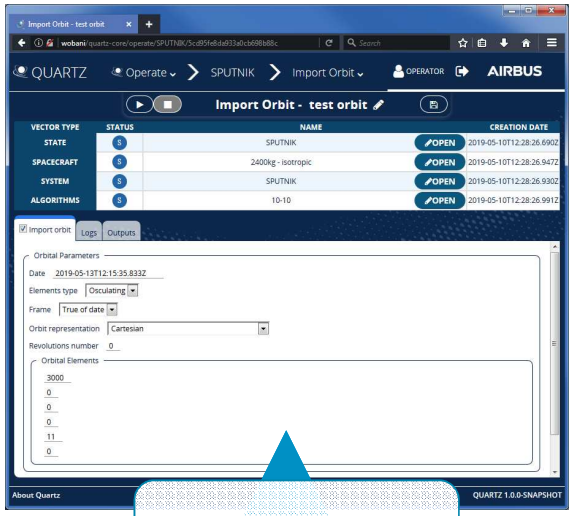
AIRBUS

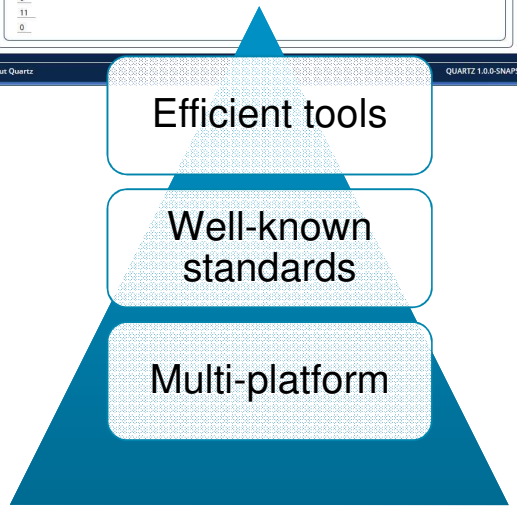
Transition from Fortran to Java

Quartz product family









Orekit, a key component for the Java rewrite of Quartz

- Orekit covers all pure flight dynamics requirements
- Our team can focus on operational and mission analysis algorithms
  - This is our core business
  - Greater value for our customers

Orekit was a significant contributor to the decision of switching Quartz to Java

Legal aspects

Orekit is released under **Apache 2.0 licence**

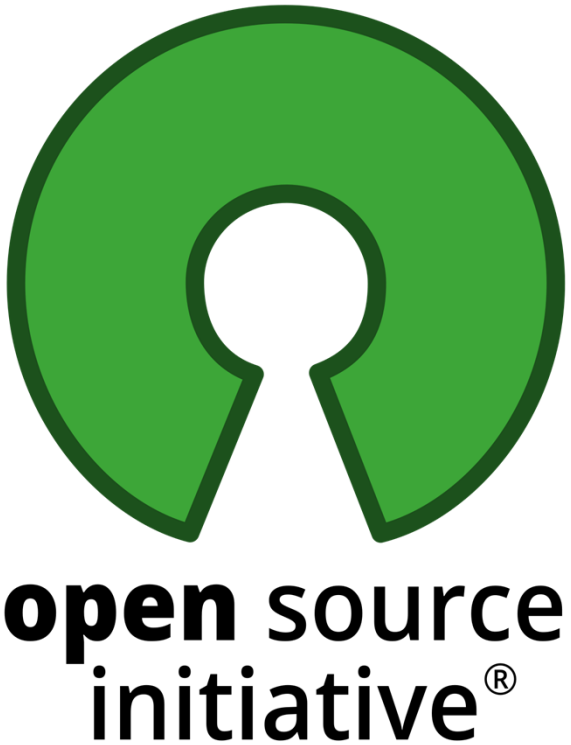
- Orekit is open-source
- Selling a product based on Orekit is allowed
- The product sold can be closed-source.

Compliant with Quartz business model.

THE APACHE® SOFTWARE FOUNDATION apache.org

The value of open-source

- The code can be reviewed
- The software will be maintained as long as a development community finds it useful
- Contributing is possible, instead of building workarounds
  - Airbus DS has made a quality-of-life contribution in 2018: making OrekitException an unchecked exception
  - Another contribution is currently being prepared: variable-thrust maneuvers



AIRBUS

Orekit strong points

Airbus has developed an internal mission analysis tool based on Orekit a few years ago. This has allowed us to get real-world feedback before committing more resources to an Orekit-based version of Quartz.

- Professional training courses allow to quickly become productive with Orekit
- Orekit software/documentation quality is at least on-par with commercial software
- Bug fixing is the fastest we have ever seen, commercial products included
- Support is very reactive on the forums, both by developers and community members
- Orekit's community seems to be growing in numbers, so it should hopefully remain active for the years to come
- Project infrastructure is actively maintained: deployment of a forum and new development tools in 2018

Orekit potential improvements

Using Orekit for an operational Flight Dynamics Software proved to be more demanding than for a regular mission analysis tool.
Note : in Quartz, we use Threads to handle multiple activities simultaneously with minimal overhead.

- Not designed for multi-threading with heterogeneous data contexts
  - Orekit uses quite a few static variables to improve performance, assuming that those variables will have the same value for all threads (physical models). This is not always true for Quartz, where simultaneous computations could require different data.

- Internal state that can influence subsequent computations
  - To improve performance of some heavy computations (most notably frame conversions) some results are cached for potential reuse when another computation with nearly-equal inputs is required.
  - This is a bit painful for automated testing: the order in which the tests are executed has a (minor) impact on the results

In an attempt to solve both issues, I am experimenting with the Java class *InheritableThreadLocal* which is basically a static variable with a per-thread value. This is promising, but a significant amount of work would be required to change all non-final static variables into *InheritableThreadLocal*.

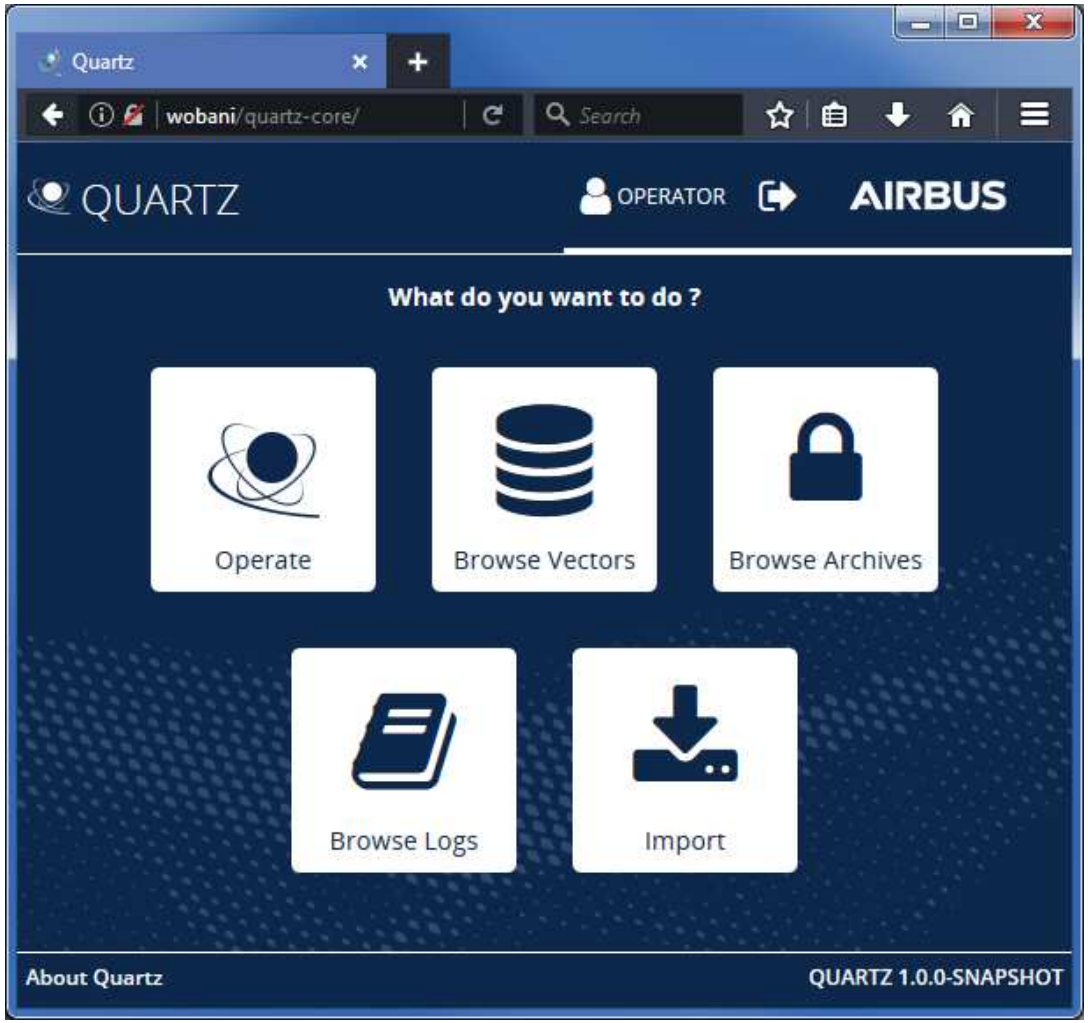
Current status of Quartz Java

Development team

- Development started 2.5 years ago.
- About 5 developers worked on the project on average (started with 2, peaked at 10).
  - Mix of profiles: Flight Dynamics skills and Computer Science skills
  - Developers are also users of the software
- SCRUM agile method to coordinate developments
- Modern development environment and techniques:
  - Continuous integration using Jenkins
  - Code management with Gitlab

First major version being released

- 9 intermediate development versions have been already deployed successfully in an Airbus control center
- Quartz v1.0.0 is currently being delivered
- Quartz Java has become the baseline FDS for all new platforms. Fortran generation is still being used and maintained for current programs.
- First use in operations will occur next year.



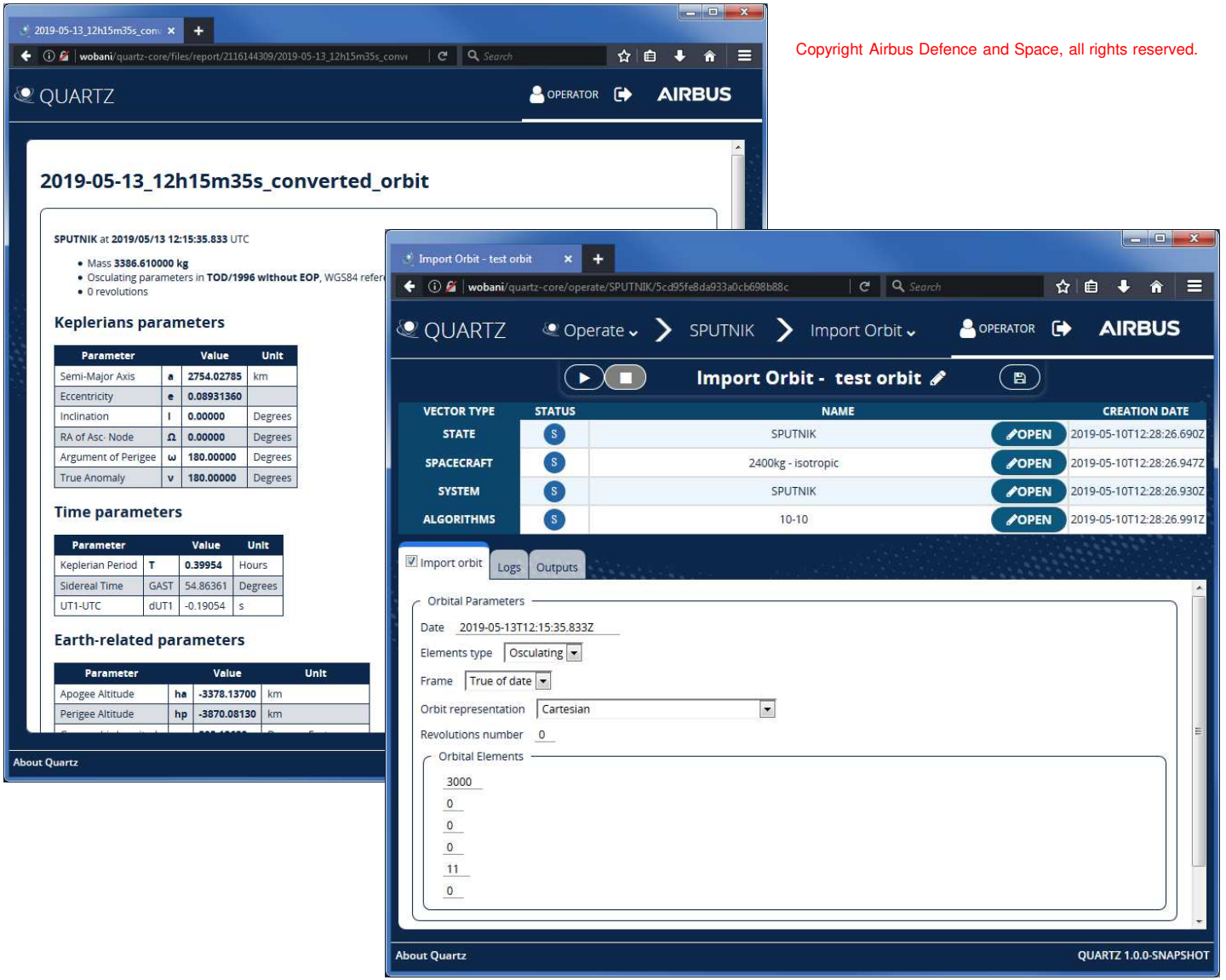
GUI overview

- Web frontend

ANGULAR

- Workstation needs only a browser (no internet access required)
- Well-suited for multi-screen
- Ability to bypass the GUI and automate any activity via REST API instead

mongoDB



AIRBUS

Final word

- After 2.5 years of development, we now have a first major version of Quartz Java.
- First use in operations will occur next year
- Quartz Fortran is still being used, but will slowly enter into a maintenance phase
- Orekit was a major building block for Quartz Java

Thank you

Contacts:

- Yannick Jeandroz, Quartz Java Product Owner: yannick.jeandroz@airbus.com
- Benoît Blais, Flight Dynamics Team Leader: benoit.blais@airbus.com

AIRBUS