

S.Loyer⁽¹⁾, S.Banville⁽²⁾, F.Perosanz⁽³⁾, F.Mercier⁽³⁾

⁽¹⁾ CLS, Ramonville St Agne, France

⁽²⁾ NRCAN, Ottawa, Canada

⁽³⁾ CNES, Toulouse France

Introduction:

This document is a proposal for exchanging attitude data information within IGS based on the ORBEX format¹ as discussed initially in Vienna at the EGU “IGS Multi-GNSS Working Group Splinter Meeting” (April 2017)^{2,3} and decided at the IGS AC meeting of Potsdam (April 2019)⁴.

History of changes in this document:

1. June 2017: Initial version
2. April 2019 (S. Loyer): Corrected version with clarifications on the use of the ORBEX format, precisions on the frames related to the quaternion transformation and modified Example of appendix 2.

Clarifications on ORBEX format:

This proposal is limited to attitude data exchange considered as additional information relative to satellite CoM terrestrial positions and satellite clocks already provided in .sp3 and .clk files. We use the ORBEX format version 0.09 with the following guidelines:

1. The “Header Lines” and “FILE/DESCRIPTION block” are mandatory and follow the description of ORBEX 0.09 (p 9).
2. The SATELLITE/ID_AND_DESCRIPTION Block is "Mandatory" and must list the satellites appearing in the file (with the 3 characters of the satellites IDs). Other information (Satellite Description) can be omitted.
3. The time system to be used is the GPS time as in the sp3 and the clk files (Then the TIME_SYSTEM label of the FILE/DESCRIPTION block should be “GPS”)
4. The stepsize shall be fixed and preferably coherent with the clocks one used in .clk files. Epochs containing no information at all must be given even if the data block is not populated.
5. We recommend limiting the content of the EPHEMERIS/DATA block to attitude records (ATT) using the fixed format given in ORBEX 0.09 (p 32). This will avoid redundancy with the content of sp3/clk files but

¹ The complete description of the ORBEX file format is available at:
<ftp://ftp.ngs.noaa.gov/pub/ORBEX/ORBEX009.pdf>

² IGS Multi-GNSS Working Group Splinter Meeting EGU, April 2017, Vienna - Summary and Conclusions.

³ IGS Multi-GNSS Working Group Splinter Meeting EGU, April 2017, O.Montenbruck, P.Steigenberger - Handouts materials.

⁴ Workshop 2019 recommendations: https://s3-ap-southeast-2.amazonaws.com/igs-acc-web/igs-acc-website/workshop2019/Workshop_Findings.pdf

other records like PCS, VCS, POS, CLK, etc. are allowed (if specified in the LIST_OF_REC_TYPES lines of the FILE/DESCRIPTION block).

6. Unknown or invalid ATT data for any satellite can be omitted. ORBEX (section 2.3) allows a variable number of satellites at each epoch.
7. We follow the recommendation of the Vienna MGEX meeting (2017) to give the attitude quaternions between ITRF and Body-Frame of the satellite. The four parts of the quaternion (q_0 being the scalar part of the Quaternion) will provide the transformation from the terrestrial frame to the spacecraft body frame. ORBEX follow the quaternion notation (q_0, q_1, q_2, q_3) outlined in [Kuipers 1999]⁵ and [Montenbruck 2000]⁶ and the ECEF frame involved in the transformation must be specified in the COORD_SYSTEM LABEL field of the FILE/DESCRIPTION block. The transformation can also be described with comments lines at the start of EPHEMERIS/DATA block. An example of file, limited to attitude data, is given in Appendix 2.

Name of the files:

Names of ORBEX ATT files will follow the general rules in place within IGS with extension .obx or .OBX (e.g. GRG0MGXFIN_20190020000_01D_30s_ATT.OBX or gbm20343.obx).

Detailed definitions on quaternions:

In order to avoid any misunderstanding about the quaternion representation of the attitude of the satellite, we define here the conventions to be used and the practical use of the quaternion values appearing in the ATT record.

If we note q the quaternion, where $q=(q_0, q_1, q_2, q_3)$, q_0 is the scalar part of the quaternion and (q_1, q_2, q_3) is the vectorial part of the quaternion.

A 3D vector $\vec{X}(X_1, X_2, X_3)$ is classically equivalent to the quaternion $qX=(0, X_1, X_2, X_3)=(0, \vec{X})$ with a vanishing scalar part.

A quaternion of rotation q is given by the quadruplet (q_0, q_1, q_2, q_3) with a norm equal to 1; this implies the following relationship (square of the norm equal to 1): $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$

The ATT record has to contain the 4 values describing the quaternion of rotation for the current satellite and the current date; q_0, q_1, q_2, q_3 appear in this order in the ATT record line. The given quaternion describes the transformation between the Terrestrial Frame and the Satellite Body-Frame such that the coordinates of a vector (x_1, x_2, x_3) in the Satellite Body-Frame and (X_1, X_2, X_3) in the Terrestrial frame are related to each other by the following relationship:

$$(0, x_1, x_2, x_3) = q \cdot (0, X_1, X_2, X_3) \cdot \bar{q} \quad (1)$$

5 Kuipers, J. B. 1999: Quaternions and Rotation Sequences: A primer with Applications to Orbits, Aerospace and Virtual Reality, Princeton University Press, Princeton, New Jersey. 47

6 Montenbruck, O., 2000: Quaternion Representation of BIRD Orientation and Reference System Transformations, DLR-GSOC TN00-03, www.weblab.dlr.de/rbrt/pdf/TN_0003.pdf

\bar{q} being the transposed quaternion such as $\bar{q} = (q_0, -q_1, -q_2, -q_3)$ and the product ‘.’ of two quaternions $r=(r_0, \vec{R})$ and $s=(s_0, \vec{S})$ being defined by:

$$r.s = (r_0 s_0 - \vec{R} \cdot \vec{S}, r_0 \vec{S} + s_0 \vec{R} + \vec{R} \times \vec{S})$$

The use of (1) allows for an easy computation of the $\text{Phase_Center})_{\text{earth}}$ of the satellite expressed in the terrestrial frame using the CoM position of the satellite in the terrestrial frame (read in the sp3 files) and the \overrightarrow{PCO} (Phase Center Offsets) vector expressed in the Body-Frame (read in the ANTEX file):

$$\text{Phase_Center})_{\text{earth}} = \text{CoM})_{\text{earth}} + \bar{q} \cdot (\theta, \overrightarrow{PCO}) \cdot q$$

No convention of sign is to be applied when writing the quaternion components in the ATT line, and one can put either q or $-q$ since they represent the same rotation.

Link between matrix and quaternion:

To make the coordinate transformation between the two frames, one can use either formula (1) or use the equivalent matrix transformation:

$$\vec{x} = M \vec{X}$$

$$\text{with } M = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix}$$

The above transformation (from quaternion to matrix) is straightforward to compute but the inverse one (from matrix to quaternion) can be numerically problematic in some cases; we provide an example of a Fortran program that avoids numerical problems in Appendix 1.

Appendix 1: Example Fortran routine to compute the transformation from matrix to quaternion:

```
subroutine mat2quater(xmat,quater)
!
! -----
!
! This subroutine is an example of a possible algorithm to avoid
! numerical instabilities when computing a quaternion of rotation
! from the elements of the corresponding matrix of rotation.
! The matrix given in entry have to be a rotation matrix (i.e.
! having a determinant equal to 1). This is not checked here!
!
! INPUT  : xmat (3,3) : matrix of rotation
! OUTPUT : quater(4)  : corresponding quaternion of rotation
!                   (with a positive scalar part)
!       quater=(q0,q1,q2,q3)      q0: scalar part
!                               q1,q2,q3: vectorial part
!
! example:
!
!       | 0.000000000000000000 -0.50000000000000001 -0.8660254037844386 |
! xmat= | 0.9238795325112867 -0.3314135740355917  0.1913417161825449 |
!       | -0.3826834323650897 -0.8001031451912655  0.4619397662556435 |
! gives:
!
!   quater = ( 0.5316310262343734      ! q0
!             -0.4662278970042302      ! q1
!             -0.2272920256568435      ! q2
!             0.6695807158758448 )     ! q3
!
! -----
implicit none
double precision , dimension(3,3) :: xmat
double precision , dimension(4)   :: quater , qsquare
double precision :: xtr, xs , max , denominator
integer :: imax , i
!
! computation of the squares of the quadriplet:
xtr = xmat(1,1)+xmat(2,2)+xmat(3,3)
qsquare(1) = (xtr + 1.d0)/4.d0
xs = 0.5d0 - qsquare(1)
qsquare(2) = xs + xmat(1,1)/2.d0
qsquare(3) = xs + xmat(2,2)/2.d0
qsquare(4) = xs + xmat(3,3)/2.d0
!
! we cannot take as it is the square root of these squares since this
! gives numerical errors for nearly vanishing values.
!
! selection of the max of the values (to be used like pivot):
!
max = -1.d0
do i=1,4
  if (qsquare(i) > max ) then
    max = qsquare(i)
    imax = i
  end if
end do
quater(imax) = sqrt(qsquare(imax))
denominator=4.d0*quater(imax)
!
! quaternion values now computed using extra diagonal terms
! of the matrix
!
if (imax == 1) then
  quater(2)=(xmat(3,2)-xmat(2,3) )/denominator
  quater(3)=(xmat(1,3)-xmat(3,1) )/denominator
  quater(4)=(xmat(2,1)-xmat(1,2) )/denominator
else if (imax == 2) then
```

```

    quater(1)= (xmat(3,2)-xmat(2,3))/denominator
    quater(3)= (xmat(1,2)+xmat(2,1))/denominator
    quater(4)= (xmat(1,3)+xmat(3,1))/denominator
else if (imax == 3) then
    quater(1)= (xmat(1,3)-xmat(3,1))/denominator
    quater(2)= (xmat(1,2)+xmat(2,1))/denominator
    quater(4)= (xmat(2,3)+xmat(3,2))/denominator
else if (imax == 4) then
    quater(1)= (xmat(2,1)-xmat(1,2))/denominator
    quater(2)= (xmat(1,3)+xmat(3,1))/denominator
    quater(3)= (xmat(2,3)+xmat(3,2))/denominator
end if
!
! as quater and -quater are equivalent this subroutine always return
! the one of the two having a positive scalar part.
! warning: this procedure does not imply that two successive call
! will give two quaternions that can be interpolated.
!
if (quater(1)<0.d0) quater(1:4)=-quater(1:4)
!
return
end

```

Appendix 2: Example of ORBEX file limited to attitude data

```

%=ORBEX 0.09
%%
+FILE/DESCRIPTION
DESCRIPTION      Attitude quaternions for grg/grm products
CREATED_BY       CNES-CLS IGS-AC
CREATION_DATE    2019 05 02 12 34 54
INPUT_DATA       u+U
CONTACT          igs-ac@cls.fr
TIME_SYSTEM      GPS
START_TIME       2018 10 21 00 00 0.000000000000
END_TIME         2018 10 21 00 01 0.000000000000
EPOCH_INTERVAL   30.000
COORD_SYSTEM     IGS14
FRAME_TYPE       ECEF
LIST_OF_REC_TYPES ATT
-FILE/DESCRIPTION
+SATELLITE/ID_AND_DESCRIPTION
E01
E02
E03
R01
R02
R03
G01
G02
G03
-SATELLITE/ID_AND_DESCRIPTION
+EPHEMERIS/DATA
*ATT RECORDS: TRANSFORMATION FROM TERRESTRIAL FRAME COORDINATES (T) TO SAT. BODY FRAME ONES (B) SUCH AS
*
      (0,B) = q.(0,T).trans(q)
*REC ID_      N ___q0_(scalar)_____ q1_x_____ q2_y_____ q3_z_____
## 2018 10 21 00 00 0.000000000000 09
ATT E01      4 0.2796988739859625 0.0767732228075297 0.9535493300680007 -0.0813516273813716
ATT E02      4 -0.0763832709942057 0.2798108239960775 0.0805438903508286 0.9536163696235584
ATT E03      4 0.5051654088446309 -0.0123667305508339 -0.8608432558942428 0.0600321785688232
ATT R01      4 0.3929179843903584 0.1771727389613362 -0.7035390216886975 0.5650293116935323
ATT R02      4 0.1005541208221312 0.1166037923840662 -0.3880332242101842 0.9086928200973037
ATT R03      4 -0.0278325195218448 -0.1619139938829182 -0.1519125633400278 0.9746444390347746
ATT G01      4 0.4062502437634697 -0.0970122592581474 -0.8694030375664159 -0.2639843163258768
ATT G02      4 -0.1496681237335243 -0.2530795077026955 -0.7371638931910669 0.6083910009972509
ATT G03      4 0.3203878356598950 -0.0321052432048551 -0.9467398141198986 0.0021476685370511
## 2018 10 21 00 00 30.000000000000 09
ATT E01      4 0.2794666584952466 0.0788926857131641 0.9532771962325394 -0.0832881628654021
ATT E02      4 -0.0785052158963970 0.2795801729714974 0.0824831455053343 0.9533458914086921
ATT E03      4 0.5041284994648602 -0.0141082233717651 -0.8615253227849633 0.0585622084698466
ATT R01      4 0.3936182419914179 0.1754501671718069 -0.7054536941674464 0.5626873055227684
ATT R02      4 0.1007699680301561 0.1175295119858701 -0.3895677507545111 0.9078927221470803
ATT R03      4 -0.0285733933451835 -0.1603295276881839 -0.1532496228213486 0.9746756162175904
ATT G01      4 0.4065921190368229 -0.0986582811099466 -0.8684206983132919 -0.2660730783950743
ATT G02      4 -0.1496939903353548 -0.2526908562350616 -0.7384408338917909 0.6069960257479998
ATT G03      4 0.3197072840913043 -0.0334593005685304 -0.9469253855505140 0.0002047057334085
## 2018 10 21 00 01 0.000000000000 09
ATT E01      4 0.2792315418951427 0.0810077144798338 0.9529976390208150 -0.0852232141283546
ATT E02      4 -0.0806227445862077 0.2793466020796836 0.0844209198940245 0.9530679709475965
ATT E03      4 0.5030876207653896 -0.0158419216213776 -0.8622015285941634 0.0570999425730834
ATT R01      4 0.3943114626522111 0.1737119189156034 -0.7073624054387999 0.5603401351139925
ATT R02      4 0.1009931412040600 0.1184458879020473 -0.3911061152064730 0.9070870761492159
ATT R03      4 -0.0293088796883079 -0.1587475977271824 -0.1545879463757490 0.9747013679187043
ATT G01      4 0.4069287788973764 -0.1003018648871554 -0.8674315645499522 -0.2681622375124261
ATT G02      4 -0.1497097542061216 -0.2523063427000890 -0.7397124347680939 0.6056021902026991
ATT G03      4 0.3190254690192300 -0.0348094961213492 -0.9471050800871627 -0.0017367699695737
-EPHEMERIS/DATA
%END_ORBEX

```