Network Earth Rotation Service

Leonid Petrov *ADNET Systems Inc/NASA GSFC*
Christian Bizouard *Paris Observatory, France*

Problem statement:

Earth rotation business has four pillars:

- **Theory**
- **Observations**
- **Data analyses**
- **Delivery to customers**

Delivery to customers is the weakest point.

Past and present:

1895–1990s	Printed bulletins are sent by surface mail
1990s–2016	Bulletins are put on the ftp

Modern requirements:

- UTC service: NTP protocol. Seamless client/server interaction.
- Internet services as a tool for distribution of Earth science data: data + access software = long URL
- Internet of things: interaction of Internet services.

NTP client provides UTC **function:** `time_t time(time_t *t)` and command line interface **command:** date

Can we make a similar tool that provides UT1?
Can we extend it to the Earth Rotation matrix?

Goal:

To make rotating the Earth so simple, that even kids can do that

To provide rotation matrix for the past, present, and near future.

Objectives:

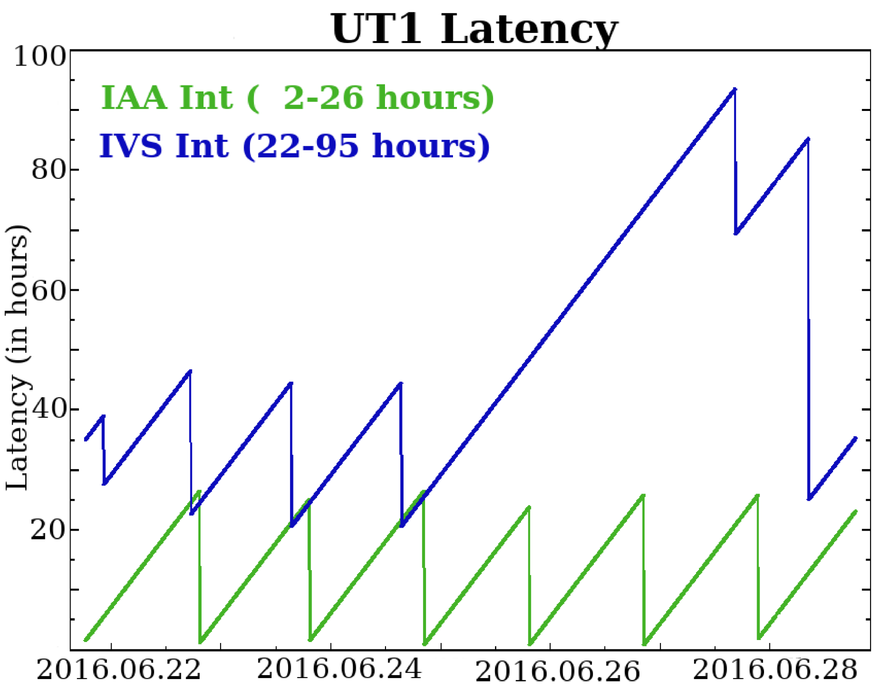
NTP:	UTC	time_t time (time_t t);
NERS:	UT1	ners_get_eop (time_t t . . .);
	EOPs	Internet service

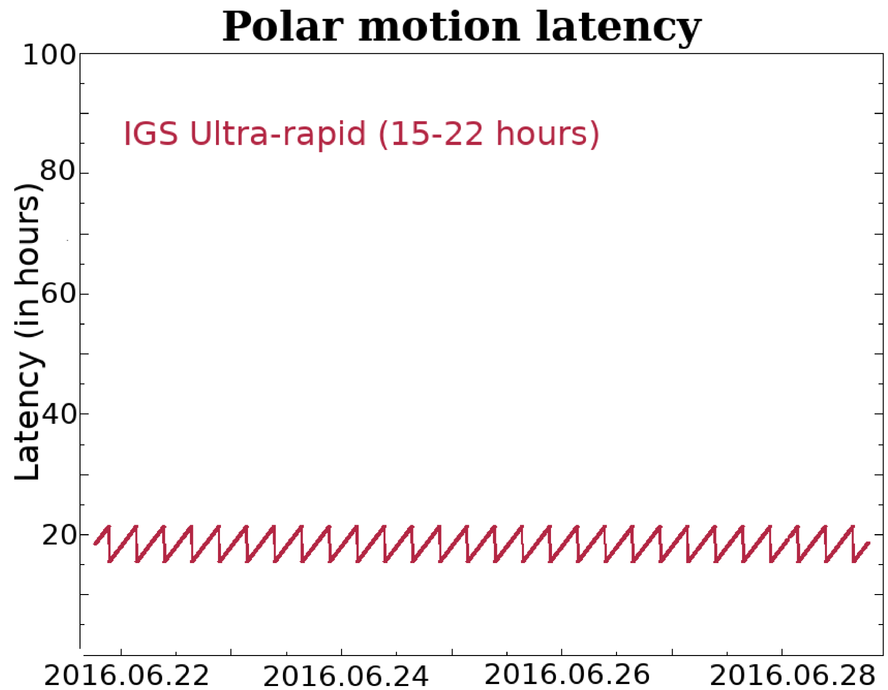
Three problems:

1. Short-term EOP forecast (1–5 days);
2. Long-term EOP prediction (5–300 days);
3. Delivery.

Short-term prediction is needed for real-time applications.

Step 1: EOP prediction





Current status: UT1 and polar motion have 1 day latency at the worst case

EOP forecast is required only for short interval (1–3 days)

Long-term prediction is needed only as a hot backup

EOP prediction:

1. Compute the AAM time series
   - Extract state of the atmosphere from NASA GEOS-FP model
   - use GEOS-FP assimilation and forecast up to 72^h
   - expand state of the atmosphere to 4D B-spline, upgrid to 1×1 km, and integrate;

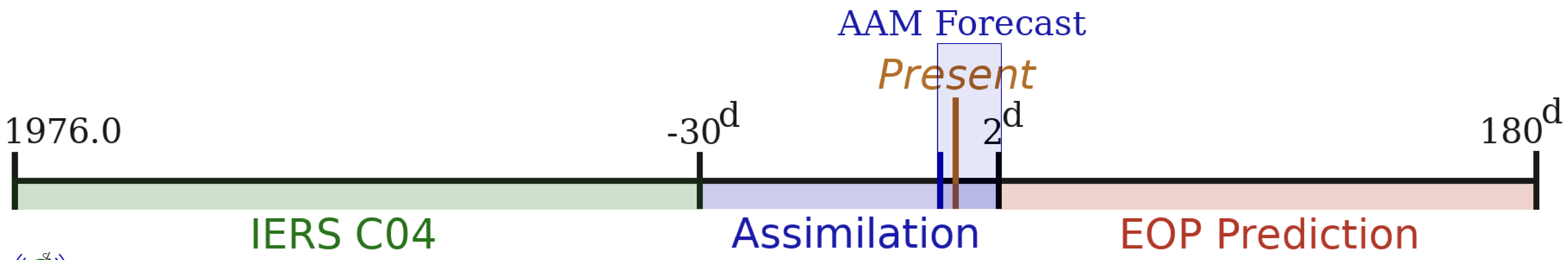
2. Expand the UT1 and polar motion into B-spline basis; assimilate using EOP estimates and AAM

3. Constraint to C04 series

4. Quarterly VLBI solutions for harmonic expansion of EOP variations

NERS EOP message:

1. B-spline coefficients for IERS C04
2. B-spline coefficients for assimilation + forecast
3. B-spline coefficients for long-term prediction
4. Coefficients of the recent harmonic expansion
5. Auxiliary information



NERS machinery:

- Ingest of NASA GMAO numerical weather models ($every\ hour$)
  - Extraction of datasets
  - Computation of the state of the atmosphere
  - Expansion of the state of the atmosphere in 3D B-splines
  - Computation of the AAM
- Ingest of EOP time series ($every\ hour$)
  - Reformatting the EOP time series
  - Computation of the EOP forecast up to 46–60^h in the future
- Running NERS client ($continuously$)
  - Refreshing EOP forecast (every 20 minutes)
  - Serving requests for EOP

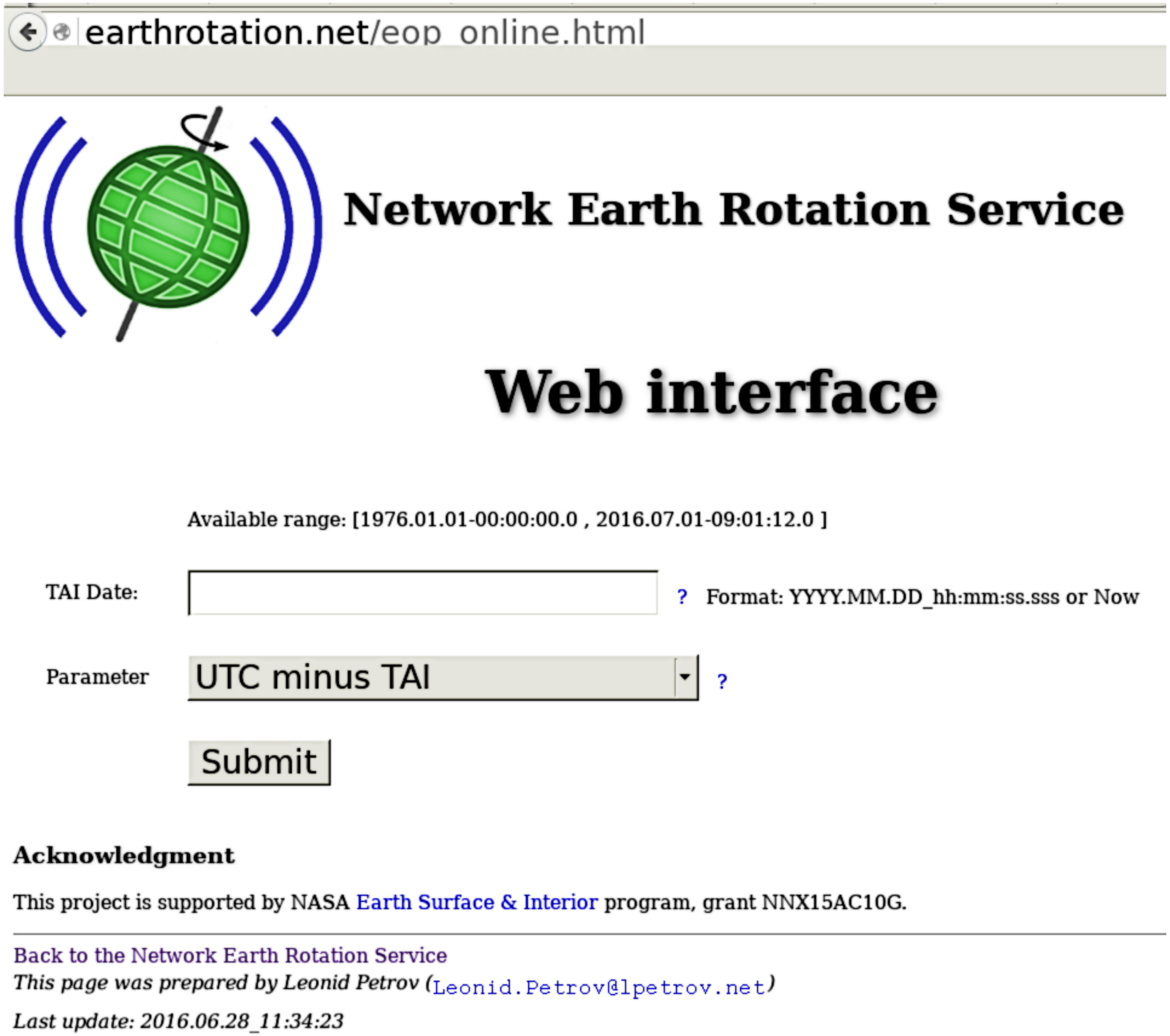
NERS User interface:

There are three levels of interface

- **Level 1**:

Advantage:
You see what you get.

Disadvantage:
Usually, you need to export the EOPs into a code that performs some computation.



NERS User interface:

- **Level 2**: Internet service

An Internet service is a request to a HTTP server with parameters. The server returns the answer upon receiving the request.

Example:

wget -q -O -
'http://**earthrotation.net**/**cgi-bin/eop_online.py**?**req_date**=**now**&**param**=**utcmtai**'

Server:	**earthrotation.net**		
Service:	**cgi-bin/eop_online.py**		
Parameter	**req_date**	value:	**now**
Parameter	**param**	value:	**utcmtai**

Advantage: you can include calls to an Internet service in your program.

Disadvantage: Internet service is rather slow. If a given server is down, your application will fail.

NERS User interface:

- **Level 3**: client library
  - Download and install ners client library
  - Insert in your code three calls:

```
ners_init    ( config_file, ners, time_tai_start, time_tai_stop, iuer)
ners_get_eop ( ners, time_tai, cpar, m_par, l_par, pars, iuer )
ners_quit    ( code, ners )
```

`cpar` may be either a EOP, or rotation matrix and its time derivatives.

`time_tai` may be in a range from 1976.01.01 through 48–60 hours in the future.

The client library downloads the EOP forecasts, keeps the local copy, maintains a buffer in memory, checks the forecast age, and performs calculations.

Advantage: very fast, efficient, and robust.
Disadvantage: requires a little bit programming.

Command line interface:

```
/tmp> date
Tue Jun 28 23:28:04 EDT 2016
/tmp> ut1mtai
-36.210602
/tmp> ners_eop -p xpol -t 2016.07.01_00:43:18.192
TAI: 2016.07.01-00:43:18.191 EOP:  7.425805576848D-07
/tmp> ners_eop -p mat

0.86133528689840511 -0.50803500927422374
-1.3974603327308885E-003 0.50803442782134745
0.86133641927649118 -7.7004947183124614E-004
1.5948955696387420E-003 -4.6687177796133145E-005
0.99999872706340454

/tmp>
```

NERS accuracy:

Comparison of past EOP forecasts to the current moment against modern C04 over the 1.5 year interval [2016.05.12, 2017.11.26]:

RMS of the differences of old NERS forecasts minus new C04:

Axes 1,2	(polar motion)	**1.1 nrad**	(7 mm)
Axis 3	(UT1 angle)	**2.5 nrad**	(16 mm)

This is the NERS accuracy for a current moment prediction

Accuracy in this context is an agreement with C04 based on 24 hour VLBI and final GNSS solutions.

Summary:

- The NERS is a step forward towards
  1. short-term EOP forecast;
  2. dissemination results in a modern form

- The NERS covers
  - Past: [1976.0, 30^d] — identical to IERS C04
  - Present: [30^d, -2^d] — best prediction of IERS C04
  - Future: [-2^d, -180^d] — coarse prediction

- The NERS is a matured service

- The NERS project is supported by NASA and Paris Observatory

Pilot project:

The NERS became operational 1.5 years ago

Time came to let the NERS in the wild

Goals of the pilot:

- collect user comments
- make adjustments in accordance with user comments
- prepare outreach material
- prepare a progress report to the next IERS DB meeting
- the IERS DB decides either to make it official product or extend as a pilot phase for a next year
- finally make it the official IERS product in addition to other products after considering all comments and making requested changes