

SEIKO

「うるう秒」を乗り越えるために

－基礎知識と障害事例・対策実例－

2016

セイコーソリューションズ株式会社



■「うるう秒」の基礎知識編

- ◆「うるう秒」って何？「うるう年」とは違います
- ◆NTPに関わる時間問題

■障害事例

- ◆2012年版障害事例
- ◆2015年版障害事例
- ◆2017年版対策方針

■対策実例

- ◆「うるう秒」対策・サーバー編
- ◆「うるう秒」対策・クライアント編
- ◆タイムサーバー導入のメリット

■「うるう秒」の基礎知識



◆なぜ「うるう秒」が発生するのか？

◆1秒の概念:

- ▶ 1950年台以前:地球の自転周期を一日として等分したものを1秒と定義
- ▶ 1960年台以降:セシウム原子時計を基準として、一秒を再定義


◆セシウム原子時計の発明により自転周期には毎日数ミリ秒の揺らぎがあることが発見される



◆地球の自転周期は少しずつ遅れているため、その誤差を吸収するために「うるう秒」を挿入/削除することになった

■ IERS（国際地球回転・基準系事業）：本部パリにて決定

- ◆ 日本では、IERSの発表を受けNICT（総務省 国立研究開発法人 情報通信研究機構）からアナウンス
 - ◆ 基本的には半年前にアナウンスされています。

「うるう秒」挿入のお知らせ 

～ 来年の元旦はいつもより1秒長い日となります ～

2016年7月8日
総務省
国立研究開発法人情報通信研究機構

平成29年（2017年）1月1日（日）に1年6ヶ月ぶりとなる「うるう秒」の調整が行われます。日本の標準時の維持・通報を実施している国立研究開発法人情報通信研究機構（以下「NICT」、理事長：坂内正夫）は、日本標準時に「うるう秒」の挿入を実施する予定です。

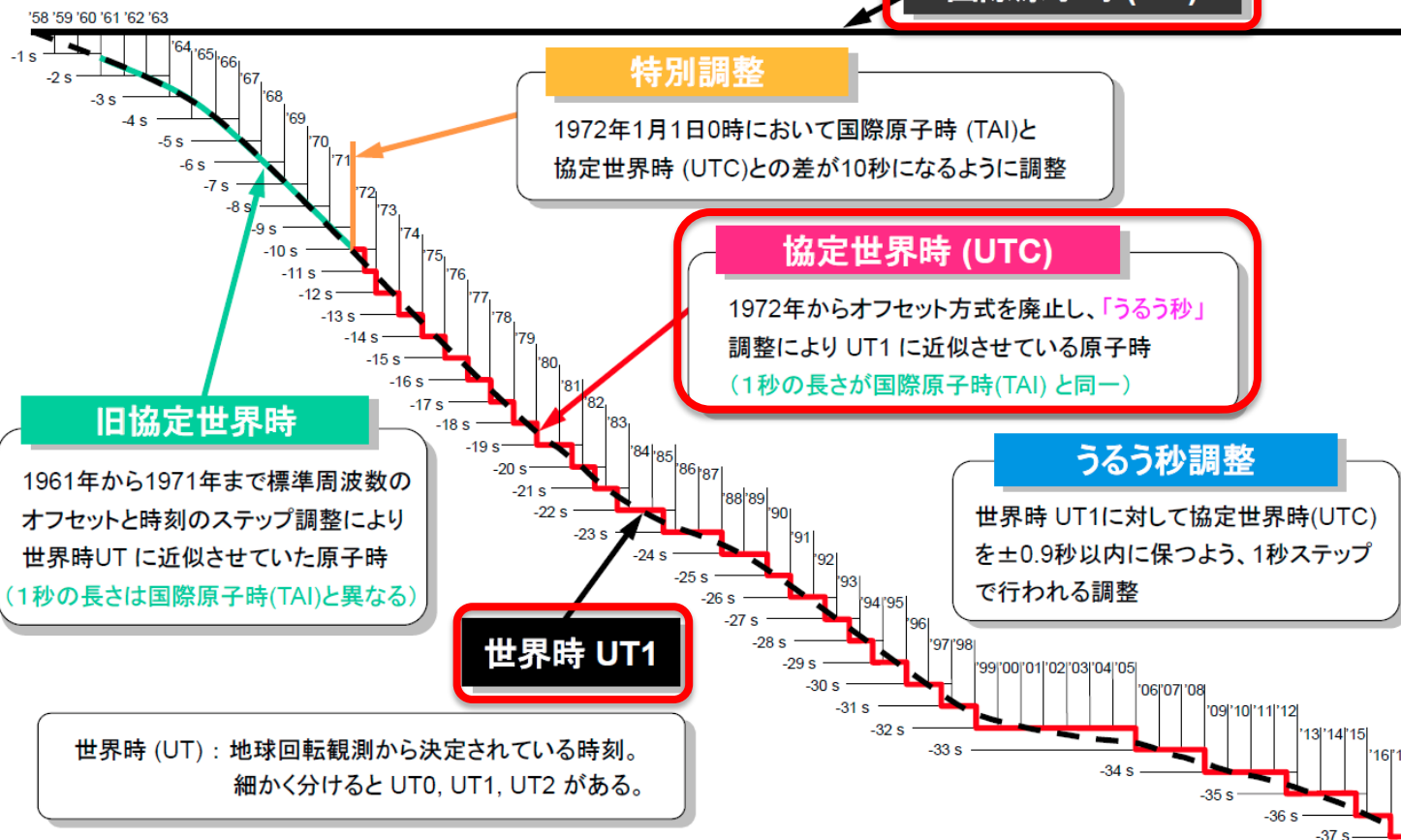
【今回のうるう秒の調整】
平成29年（2017年）1月1日（日）午前8時59分59秒と
午前9時00分00秒の間に「8時59分60秒」を挿入します。

原子時と「うるう秒」

情報通信研究機構
時空標準研究室



国際原子時 (TAI)



■ 原子時計がずれているのではないか？

- ◆ 1秒を出来るだけ地球の自転周期に合わせて調整したものがセシウム原子時計の仕様です。



セシウム原子時計 (Wikipediaより参照) <https://ja.wikipedia.org/wiki/%E5%8E%9F%E5%AD%90%E6%99%82%E8%A8%88>

- ◆ 「うるう秒」が発生することは、原子時計を「時の基準」にした瞬間から予測されていました。

■ 地球の自転速度が遅くなっている？

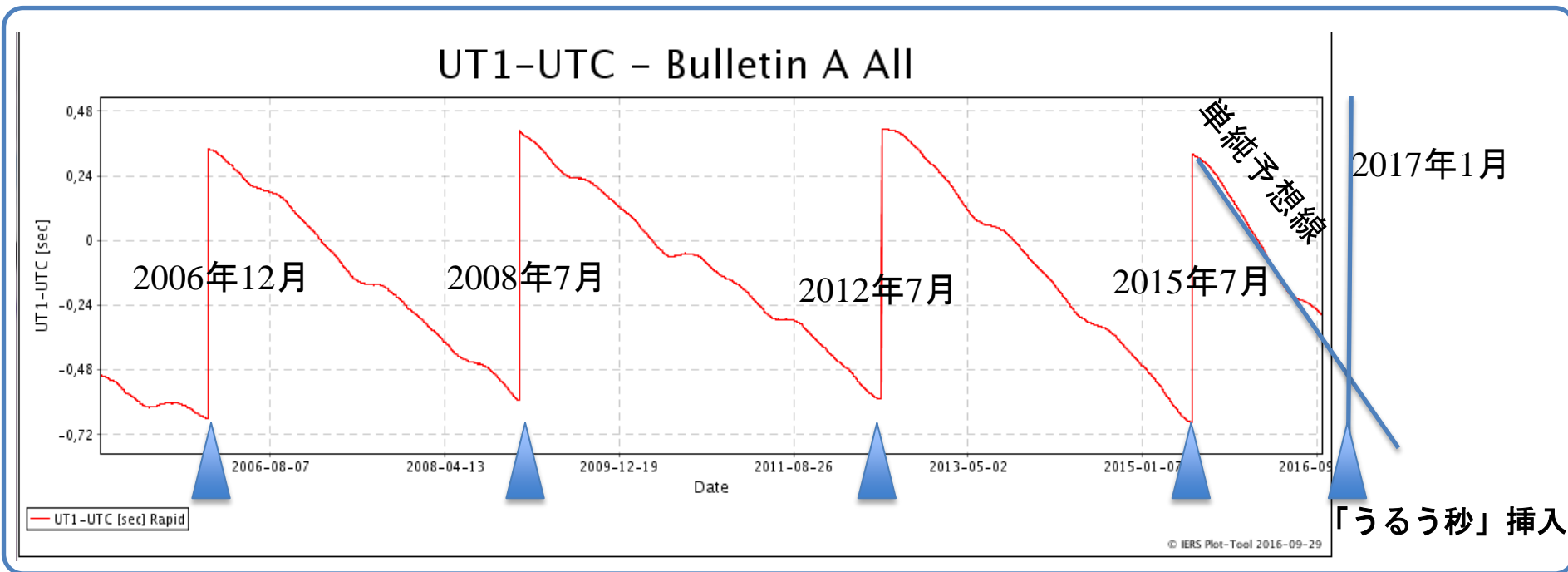
- ◆ 当初予想に比較して、遅くなっていることは事実です。
- ◆ 「うるう秒」は半年に一度、挿入or削除を実施可能です。
- ◆ しかし、今まで一度も削除されたことはありません。

◆ 自転の変動要因

- ◆ 地震等による短期的な揺れ → 早/遅
- ◆ 大気の摩擦による回転エネルギー減少 → 遅

■ IERSの世界時と協定世界時の比較

- ◆ IERSは、日々の自転情報を公開したもの
- ◆ 以下は、UT1-UTCをプロットしたもの



参照ページ: <https://www.iers.org/IERS/EN/DataProducts/EarthOrientationData/eop.html>

■ NTPにかかわる頭の痛い時間問題



■ NTP・西暦2038年問題

<http://itpro.nikkeibp.co.jp/free/NC/NEWS/20040202/139212/?mimg&rt=nocnt>

- ◆ 要約:UNIXやLinuxのシステムクロックは1970年を0秒とした経過秒数を32bitで定義。
ここでsignedでtimeを定義していた場合、2038年にオーバーフローする。
- ◆ セイコータイムサーバー現行機種は、残念ながら未対応です。

■ NTP・西暦2036年問題

https://ja.wikipedia.org/wiki/Network_Time_Protocol#2036.E5.B9.B4.E5.95.8F.E9.A1.8C

- ◆ 要約:NTPプロトコル内の秒フィールドは、32bitで定義してしまったため2036年にオーバーフローする。
- ◆ 未対応の場合、システム時計は1900年に巻き戻り、様々な障害を引き起こします。

「うるう秒」を乗り越えるために

-「うるう秒」障害事例-



■ 最新のカーネルでも「うるう秒」の影響は防げなかった

- ・ミクシィ : SNS
- Linuxカーネルのバグでサーバ通信障害で4時間停止
- ・Reddit : リンク共有サイト
- Apache Cassandraのうるう秒に関する障害発生
- ・StumbleUpon : Webレコメンデーションサービス
- ・Yelp : レストランガイド
- ・foursquare : SNSタイプの地理情報共有サービス
- ・LinkedIn : ビジネス型SNS
- ・Mozilla : ブラウザサービス

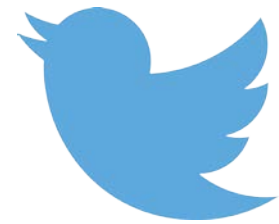
2012年障害例

- ◆ **特定バージョンのLinuxカーネルが動作しているサーバー上で、“JavaやMySQL、Apache Hadoop”など複数スレッドで動作するミドルウェアを稼働させているという環境下でサービスを提供していた場合、「うるう秒」発生時にCPUを100%占有し、結果としてサーバーがハングアップしました。**

■ 2012年版 障害の特徴

- ◆ 最新のカーネルバージョンであっても「うるう秒」障害
 - ◆ Webに情報公開しているサービス事業者ばかりではないので、実際には様々なシーンで「うるう秒」障害が発生していた可能性があります。

- 2012年は、facebookやTwitterの黎明期だったこともあり、一般の人は「うるう秒」障害を実感していない（未認識）状況でした。



■ 2015年は、本格的な「うるう秒」対策元年

- ◆ 2012年の障害の経験を踏まえ、本格的な「うるう秒」対策元年となったのが2015年でした。

- 日本時間2015年7月1日08時59分60秒の「うるう秒」では、前回の教訓を活かし、最新パッチを適用することで「うるう秒」を無事乗り越える方法が広く知れ渡りました。

2012年:「うるう秒」障害元年



2015年:「うるう秒」対策元年

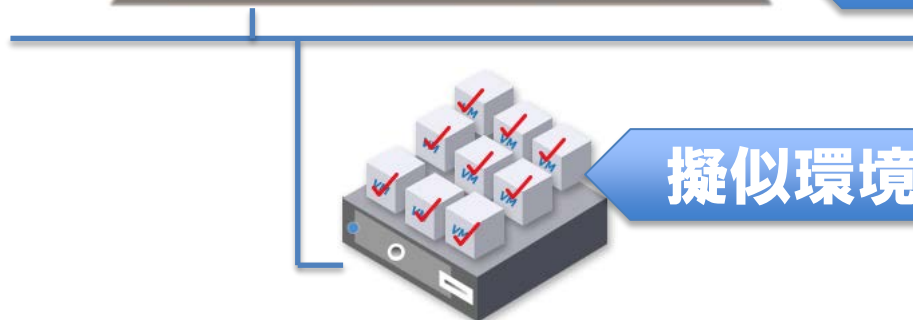
■ 擬似「うるう秒」環境テスト

- ◆ 一部のお客様からは、弊社に実際の「うるう秒」発生時と同じ環境を再現して事前テストを行いたいというご相談がありました。
- ◆ 検証作業は、半年もの時間をかけ、詳細な試験項目をこなすという入念なものでした。

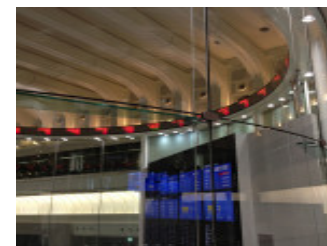
Time Server



「うるう秒」擬挿入



擬似環境



■ 2017年1月1日をいつもどおりのお正月にするために

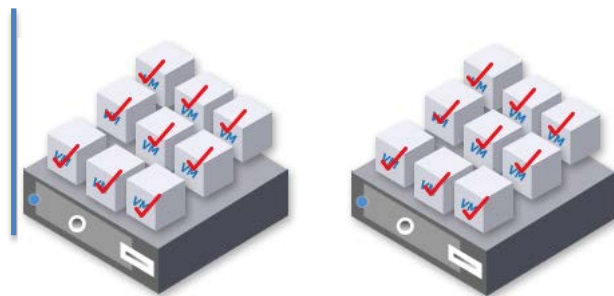
◆ 3つの対策です。

1. OSやアプリケーションを100%信用しない
事前に十分なテストを実施
2. 時刻戻りを起こさせない
タイムスタンプを遡行させるとログが壊れます
3. 08:59:60 を刻ませない
通常ありえない60秒をシステムに踏ませない

■ セイコータイムサーバーシリーズを利用することで、これらの対策を“事前に、簡単に”完了することができます。この機会に、導入をご検討されてはいかがでしょうか。

■ 完璧な「うるう秒」対策の難しさ

- ◆ 紹介した1～3の対策を確実に実施するには膨大な労力が必要です。さらに、場合によっては「うるう秒」の前日に適用してしまったセキュリティアップデートが原因で障害が発生する可能性があります。
- ◆ 仮想マシンが活躍している昨今では、各仮想マシンのOS種別やバージョン、動作しているアプリケーションまで全て把握し、全ての要因を検証して初めて「うるう秒」対策は万全だと言うことができます。



「やってられない・・・」

■ LIビットがバグを踏む例

- ◆ 多くの記事ではうるう秒の挿入によって範囲例外や、タイムアウトなどで着火しCPUが張り付く例を紹介
- ◆ 対策された最新のパッチを当てていればOK！ ではない



◆ 安易な例

- ◆ 「If LI != 0 exit」気持ちとしてはLIが11の場合当該パケットを無視したい

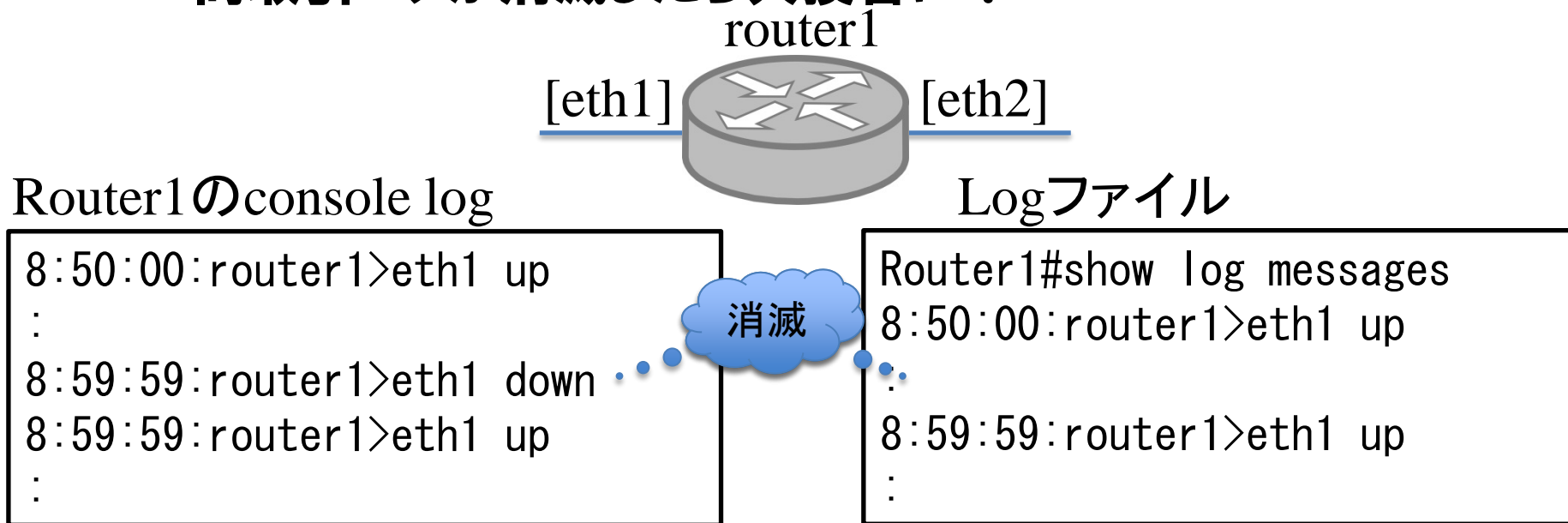
LI Leap Indicator (Leap): 2-bit integer warning of an impending Leap second to be inserted or deleted in the last minute of the current month with values defined in Figure 9.

Value	Meaning
0	no warning
1	last minute of the day has 61 seconds
2	last minute of the day has 59 seconds
3	unknown (clock unsynchronized)

Figure 9: Leap Indicator

■「うるう秒」挿入自身に問題がある

- ◆「うるう秒」挿入により、同一時刻にログが記載される
- ◆例えばルーター/スイッチの障害が上書きされ、障害原因の特定に多大な労力が必要となる
 - ◆商取引ログが消滅したら大損害に！



「うるう秒」を乗り越えるために

-「うるう秒対策」・サーバー編-



■ 分散挿入方式

数時間かけゆっくりと「うるう秒」を挿入

- ◆ SSOL製品では「アジャストモード」として実装
- ◆ LIビットは常に0
- ◆ クライアントはうるう秒非対応でOK

■ 即時調整

「うるう秒」を1秒挿入

- ◆ 「うるう秒」挿入の24時間から「うるう秒」指示子を“01”（1秒挿入）にセットしNTP配信

クライアントも協
調して1秒挿入

- ◆ 日本標準時 2017年1月1日 9:00直前に「うるう秒」として1秒を挿入
- ◆ 「うるう秒」を真面目に処理するクライアントが必須

■ 分散挿入（基本は2時間）

- ◆ 2015年7月1日の「うるう秒」挿入時、「東京証券取引所」では、2時間かけて「うるう秒」を分散挿入。
- ◆ 2012年7月1日の「うるう秒」挿入時から「Google」では、「leap smear」技術を自社サーバの同期に利用。2時間の分散挿入実施。
- ◆ 2015年7月1日の「うるう秒」挿入時から「さくらインターネット」では分散挿入を採用、こちらも2時間での分割挿入。



■ 「うるう秒」には触れない

- ◆ Nasdaq/ニューヨーク証券取引所では、「うるう秒」挿入時刻帯の取引を15分間停止。（15分以内に何か起こっても落ち着くだろう..）

■ アジャスト調整(分散挿入)がおすすめのパターン

1. 時刻が連続的にならないとシステムに問題が発生する。
2. NTPの「うるう秒」指示子が”00”以外の場合に対応できない機器がある。
3. 「うるう秒」指示子対応、非対応機器が混在している。

■ 即時調整がおすすめのパターン

1. うるう秒対応クライアント (ntpdなど) を使用している。
2. 協定世界時 (UTC) に合わせた時刻を扱いたい。
3. 1秒程度の誤差(時刻戻り)が発生しても、業務運用上問題が発生しない。

■ 弊社タイムサーバーシリーズの場合

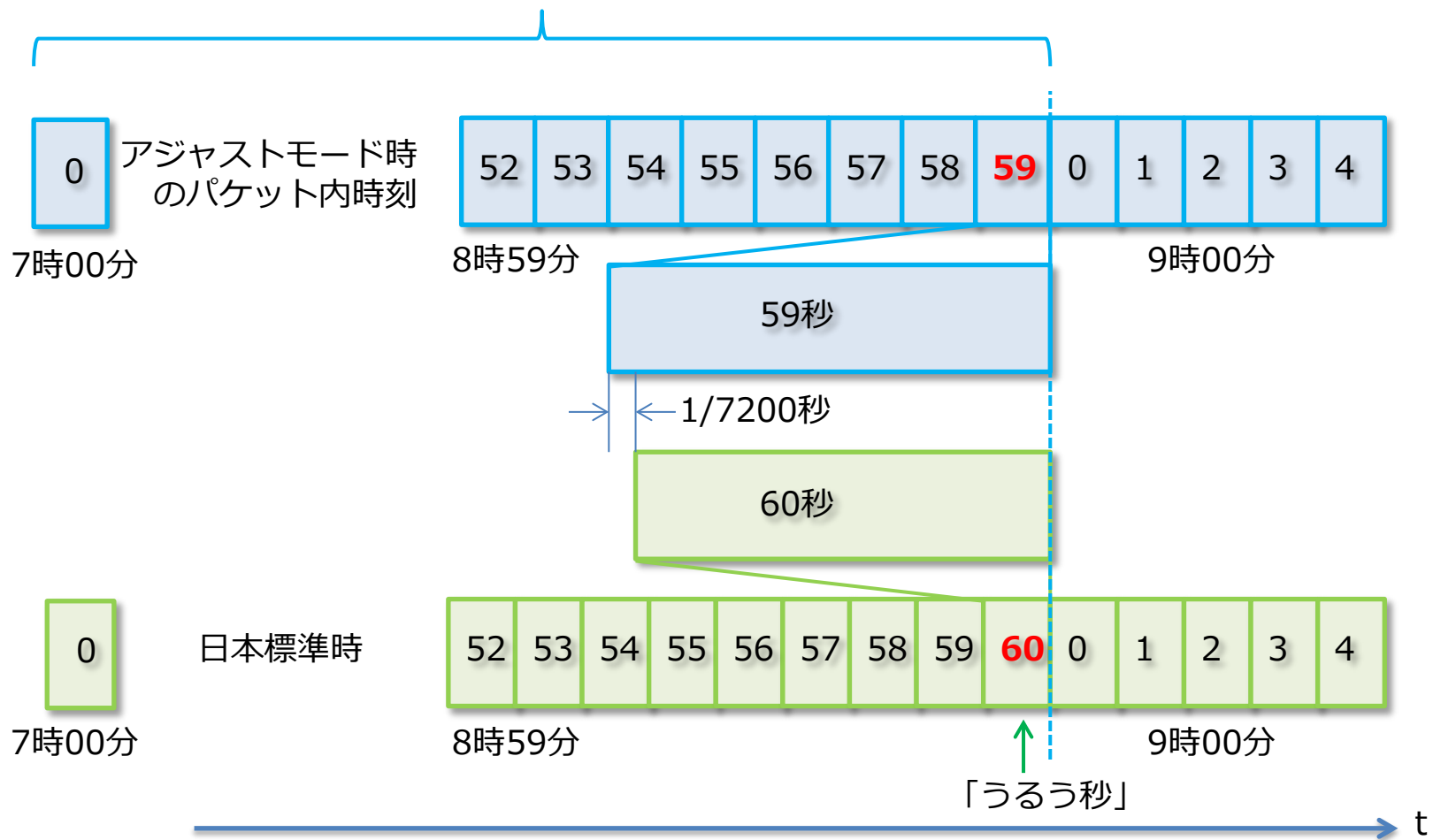
アジャストモードでは、「うるう秒」実施の約120分前より一定の割合で徐々に時間を遅らせていき、2017年1日1日9:00に調整を完了

“59秒2回/60秒”を挿入することなく「うるう秒」対策を行うことが可能

◆ クライアントは「うるう秒」の存在を意識する必要はない

■ アジャストモードのイメージ説明

「うるう秒」発生の2時間(7200秒)
前から分散調整



「うるう秒」を乗り越えるために

-「うるう秒」対策・クライアント編-



■ NTPサーバーは経過時間を配信

◆ ”9:00:00”とは配信しない。

←良くある誤解

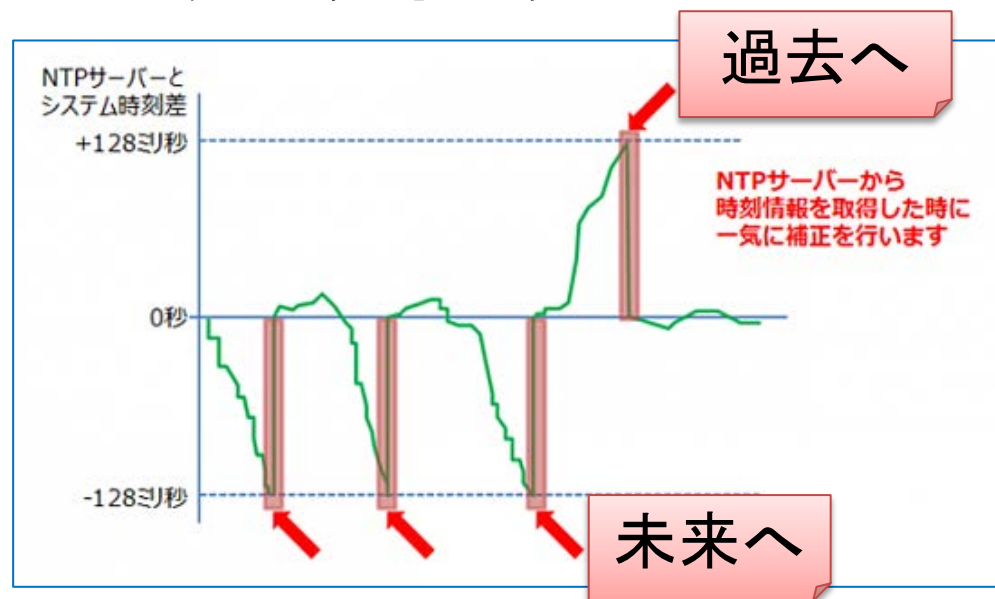
◆ 「うるう秒」は同一のNTP秒を2回配信。

2017年1月1日の「うるう秒」挿入

Date	Time	TAI Offset	NTP Leap	NTP Seconds
2017年1月1日	8:59:59	36	01	3,692,217,599
★「うるう秒」	8:59:60	36	01	3,692,217,600
	9:00:00	37	00	3,692,217,600
	9:00:01	37	00	3,692,217,601

STEPモードによる時刻調整

- ◆ NTPサーバーとクライアント時刻に128ミリ秒以上の差が生じると、NTPサーバー時刻に一気に時刻合わせを行う。
- ◆ 時刻の連続性は担保されない。
- ◆ 過去の時刻に戻る現象も発生する。



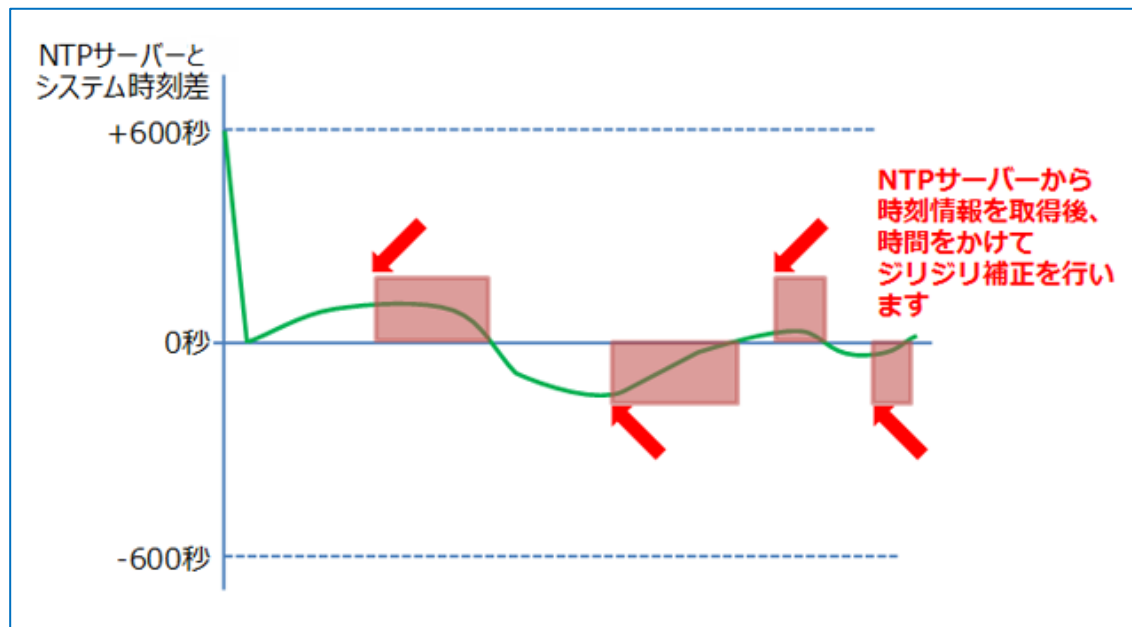
■ 「うるう秒」が発生すると話はさらに難解に
下記は、STEPモードでの「うるう秒」発生時の時刻遷移

- タイムサーバを導入せず公開NTPに同期している例
- 59秒を2度刻む動作を確認
(もちろん60秒を刻むディストリビューションもあります)

```
08:59:59.000000 ← 一度目の59秒
:
08:59:59.999999
08:59:59.000000 ← 二度目の59秒
:
09:00:00.000000
```

■ SLEWモードによる時刻調整

- ◆ NTPサーバーから取得した時刻情報で、クライアント時刻をジリジリと徐々に補正。
- ◆ 補正量は、1秒間に最大0.5ミリ秒。 ↓よくある誤解
- ◆ 600秒以上の差がある場合、一気に時刻合わせ。



- SLEWモードでは、「うるう秒」発生後の数十分間、日本標準時との時刻差が生じる。

08:59:59.000000

:

08:59:59.999999

09:00:00.000000

:

09:33:33.000000

← 「うるう秒」が挿入されない

← 秒間0.5ミリ秒ずつ標準時との差を縮める

← 2000秒 \div 33.33分かけて標準時に追いつく

- 同じ「うるう秒」処理でもモードによって大きく異なる振る舞いをします。
加えて、Linux OSのntpdのデフォルト動作はSTEPモードであることも注意が必要です。
- Centos (Linux) でのslewモード設定方法はこちらです。

```
1. ntpdを停止する
   # service ntpd stop
2. kernelの保持時刻・周波数オフセットをクリアする
   # ntpdate -s 0 -f 0
3. ntpdをslewモードで起動するようセット
   # vi /etc/sysconfig/ntpdate
   - OPTIONS="u ntp:ntp -p /var/run/ntpdate.pid -g "
   + OPTIONS="-x -u ntp:ntp -p /var/run/ntpdate.pid -g" -xオプションを追記
4. ntpd を起動
   # service ntpd start
```

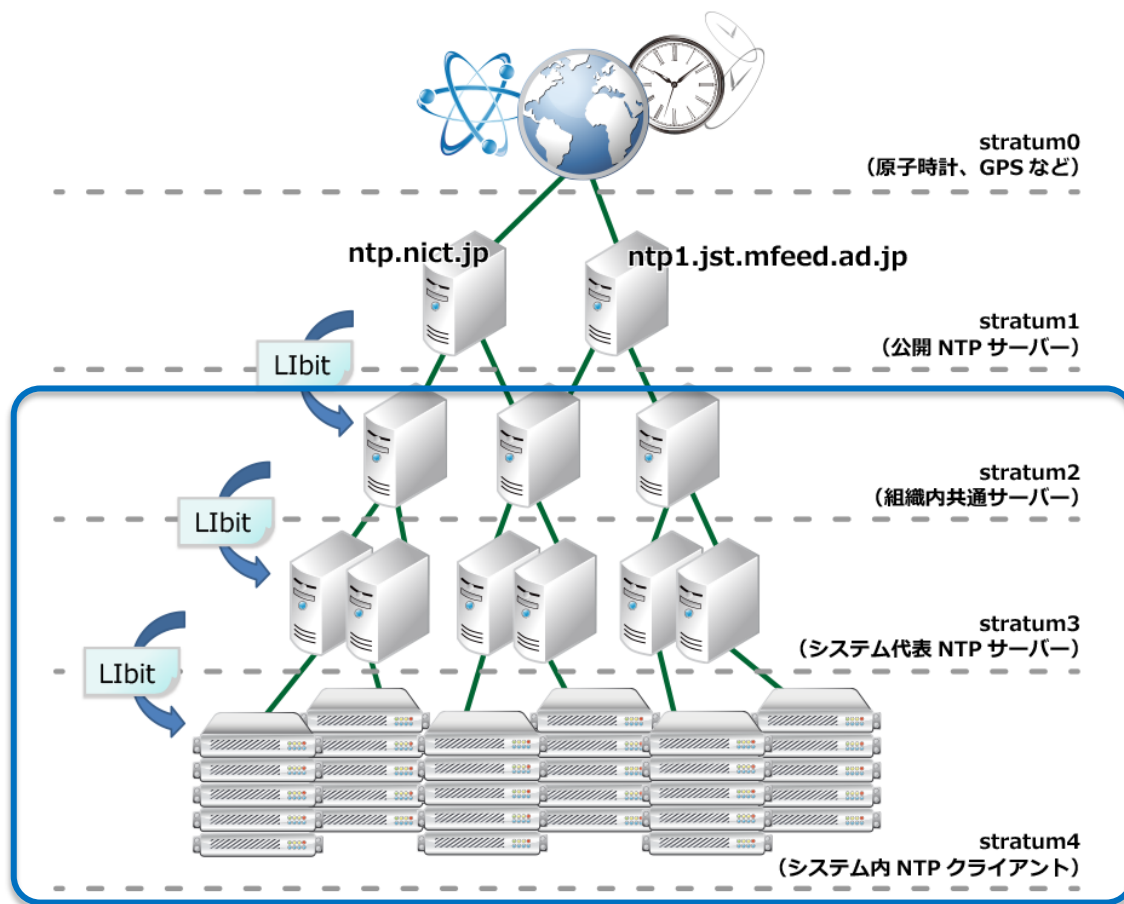
「うるう秒」を乗り越えるために

—タイムサーバー導入のメリット—



■ タイムサーバーを利用しない場合の時刻同期構成

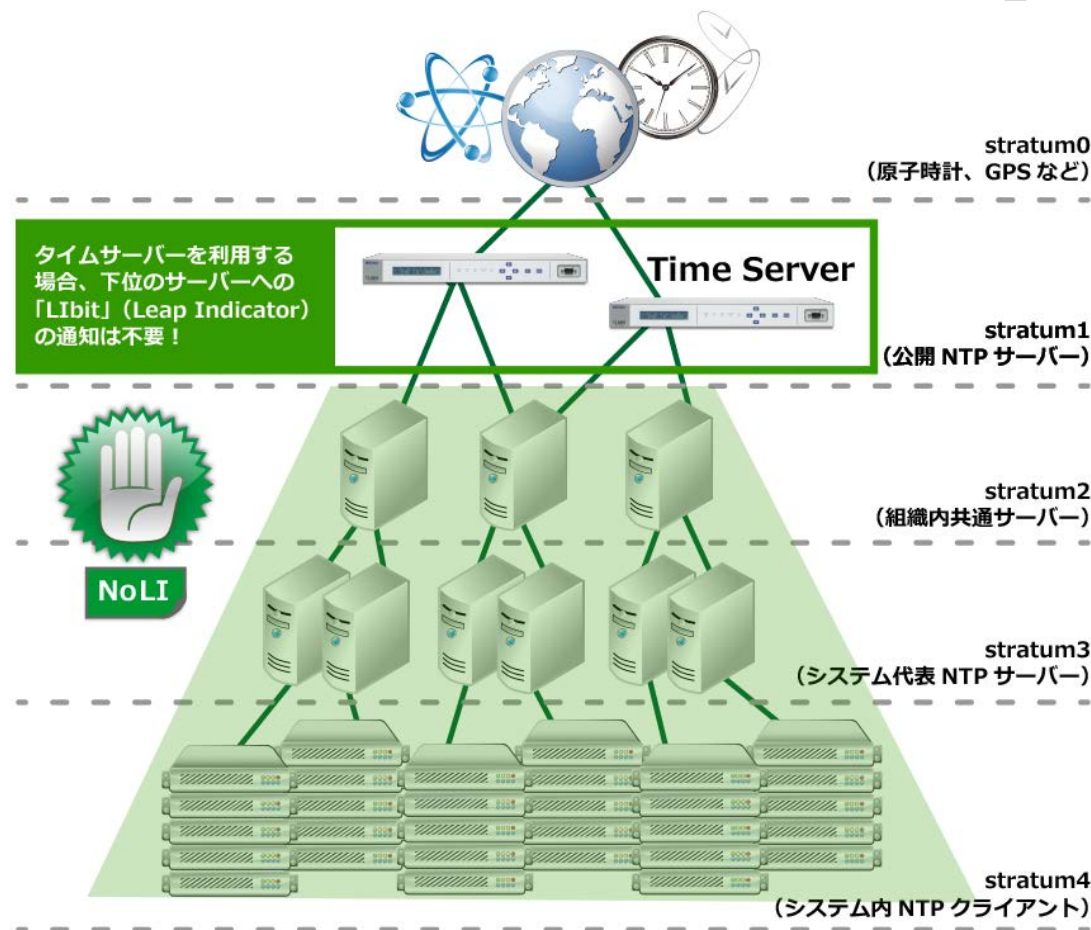
◆ 全てのサーバー/クライアントで「うるう秒」対策が必要



「うるう秒」対策範囲

■ セイコータイムサーバーを利用した場合の時刻同期構成

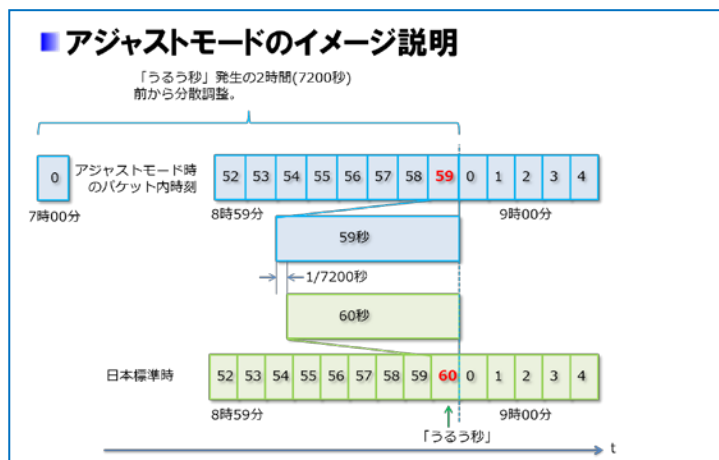
◆ アジャストモードで運用：クライアントは「うるう秒」知らず



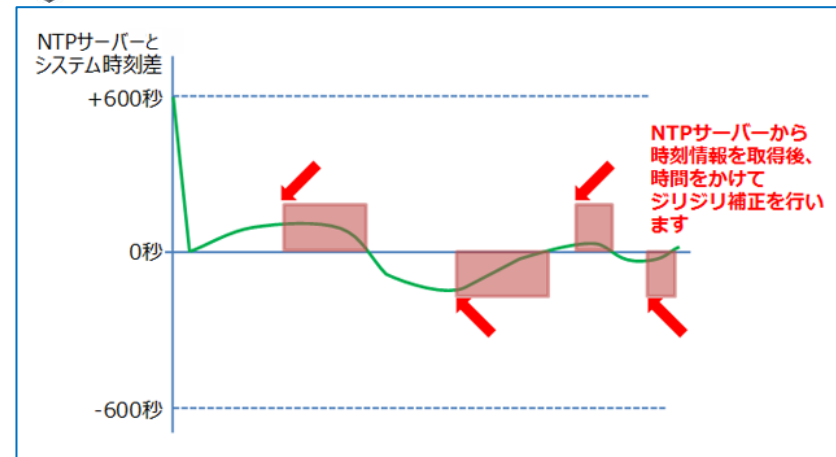
■ **タイムサーバーをアジャストモードに設定した場合、うるう秒の挿入2時間前から1秒につき1/7200秒を分散挿入します。**

◆ **アジャストモードなら、クライアントは安心して「うるう秒」を迎えることができるのではないのでしょうか。**

(決して即時調整モードを否定しているわけではありません)



&

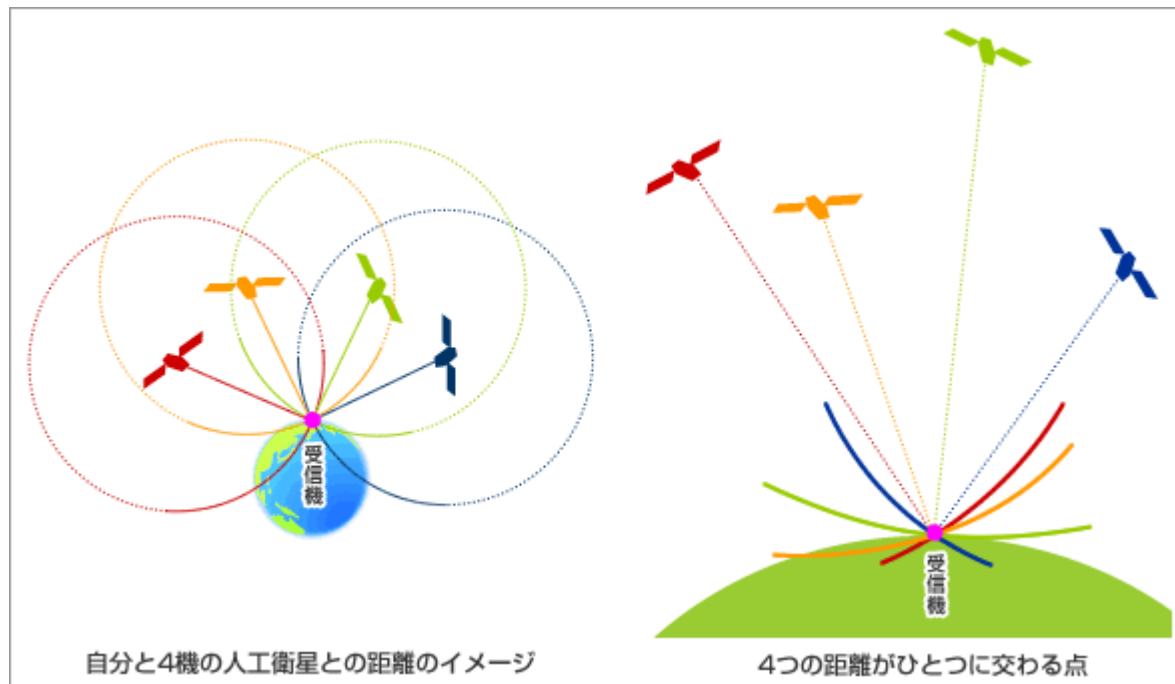


時刻源別の「うるう秒」対応

時刻源	機種	対応状況
GPS衛星	TS-2210GPS TS-2550GPS TS-2850	「うるう秒」自動設定 「うるう秒」手動設定の両方で対応可能
テレホンJJY	TS-2210TJJY TS-2550TJJY TS-2850	
光テレホンJJY	TS-2210TJJY	
長波JJY	TS-2210長波JJY	
FM放送	TS-2210FM	

■ GPSの位置情報取得にも「正確な時刻」が必要

1. 各GPS衛星から常時地上に発信される衛星自身の正確な位置情報を受信する。
2. GPS衛星から発信される正確な時刻情報を受信する。
3. 各GPSからの時刻情報の遅延から、正確な位置情報を算出する。



JAXA「今いる場所・時間がわかる測位とは？」より抜粋

http://www.jaxa.jp/countdown/fl8/overview/gps_j.html

■ GPSからの受信情報を表示

- ◆ タイムサーバーの設置位置を確認。
- ◆ 位置固定モードに遷移し、正確な時刻情報を抽出。
 - ◆ TS-2910シリーズでGPS情報を表示。
 - ◆ 「うるう秒」挿入情報あり。

タイムサーバーで受信したGNSSからのGPS情報

(c)TS-2910# show gnss

Config Information

~~~~~

Latitude : N 35.65549 #タイムサーバーの位置情報

Longitude : E 140.03993

Altitude (m) : 27.5

Synchronized State : 3D #位置情報確定を示す情報

Leap Time : 2017/01/01 09:00:00 #次の「うるう秒」情報

~~~~~ #1秒挿入を示しています

セイコーソリューションズ株式会社

ネットワークソリューション統括部 NS営業部

| | |
|--------|---|
| TEL | 043-273-3184 |
| E-Mail | ts-sales@seiko-sol.co.jp |
| URL | http://www.seiko-sol.co.jp/ |

- 本資料の著作権は弊社に帰属します。
- 本資料の一部または全部を弊社に無断で転載、複製、改変など行うことは禁じられています。
- 本資料の内容は断りなく変更することがあります。
- 本資料に記載されている会社名、製品名などは、各社の商標または登録商標です。