2017年6月8日

時刻表示ウエッブページの確度および可用性向上についての考察

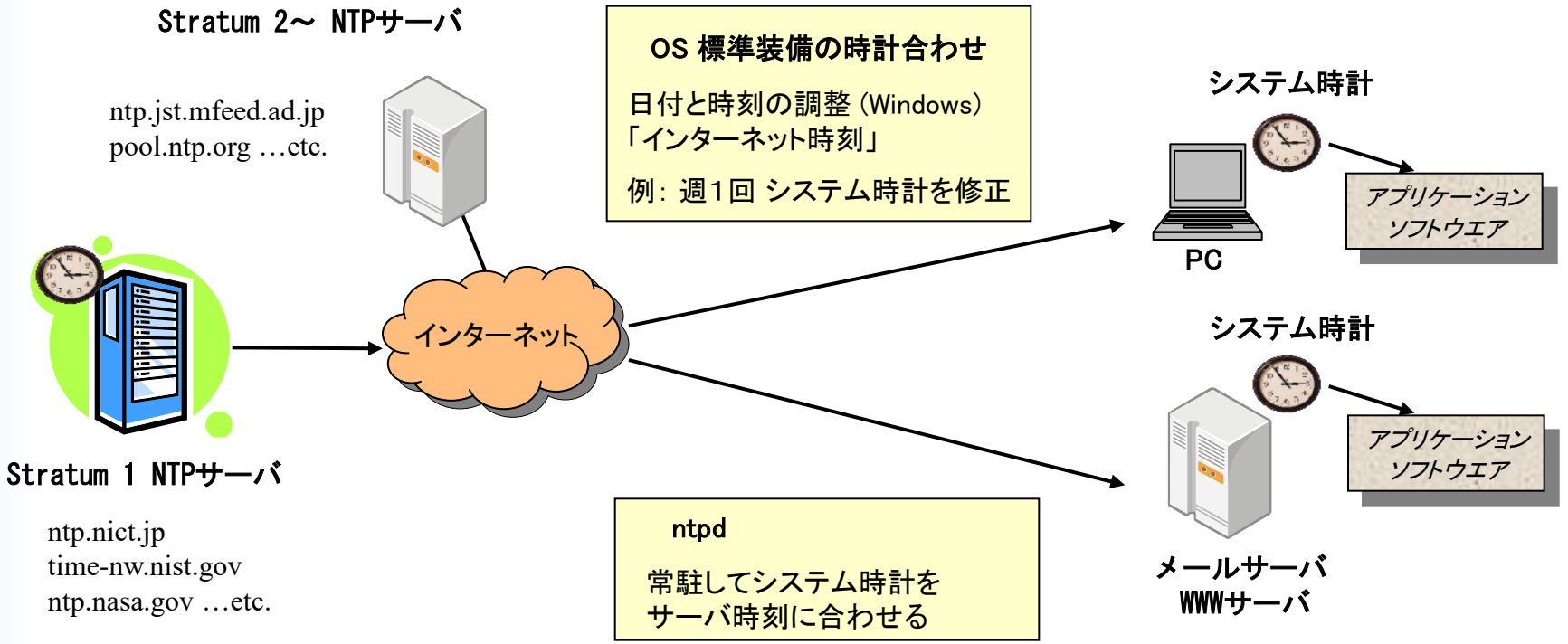
国立研究開発法人　情報通信研究機構

鳥山　裕史

NiCT National Institute of Information and Communications Technology

インターネット時刻供給

NTP が主流
- 大学、メーカ、インターネットプロバイダのサービス
- NICT Public NTP サービス （日本標準時直結）



NiCT National Institute of Information and Communications Technology

NTP 以外の時刻供給に対するニーズ

NTP 時刻供給の問題

- NTP はファイアウオールで遮断されて使えないことがある
- 時刻のトレーサビリティ確保が難しい
  - 複数のサーバ階層による時刻伝送
  - 各サーバの時刻誤差限界値が不明

新たな需要

- Web アプリケーションの利用拡大
- 精度はそこそこでも、「信頼できる時刻」の需要が増加

HTTPによる時刻供給 など

NiCT National Institute of Information and Communications Technology

HTTPによる時刻供給

クライアント側（ブラウザ上のJavaScript等）から動的に
HTTP時刻サーバにアクセスし時刻情報を取得

【メリット】
- 送出時刻・受信時刻から、取得時刻の誤差限界がわかる。
- 伝送遅延時間による時刻誤差の削減が期待できる。
- 複数サーバの利用により、時刻確度・可用性向上が期待できる。

【課題】
- セキュリティ面から、JavaScript では、ドメインを超えるアクセスが制限されている

NiCT National Institute of Information and Communications Technology

クロスドメインアクセス

△JSONP(JavaScript Object Notation with Padding)

- 動的情報（時刻情報）を埋め込んだJavaScriptソースを読込み
- クライアントは、コールバック関数中で時刻情報を利用
- クロスドメイン禁止の制限を受けない（抜け穴的）
- 外部からのJavaScriptソースを実行するため、セキュリティ面の懸念がある

◎XMLHttpRequest Level2 （AJAX）

- レスポンスヘッダでの指定により、クロスドメインのアクセス可

  → Access-Control-Allow-Origin:*

- 最新版であれば、ほとんどのブラウザで対応済み
- データ形式は、JSON が扱いやすい

NiCT National Institute of Information and Communications Technology

HTTPによる時刻供給

クライアント

JSON形式
時刻サーバ

送信時刻 T0

http://ntp-a1.nict.go.jp/
cgi-bin/json?1231234567.012

(T0+T2)/2

推定時刻差⊿T

T1

受信時刻 T2

{ “T0”: 1231234567.012,
“T1”: 1231234567.123 }

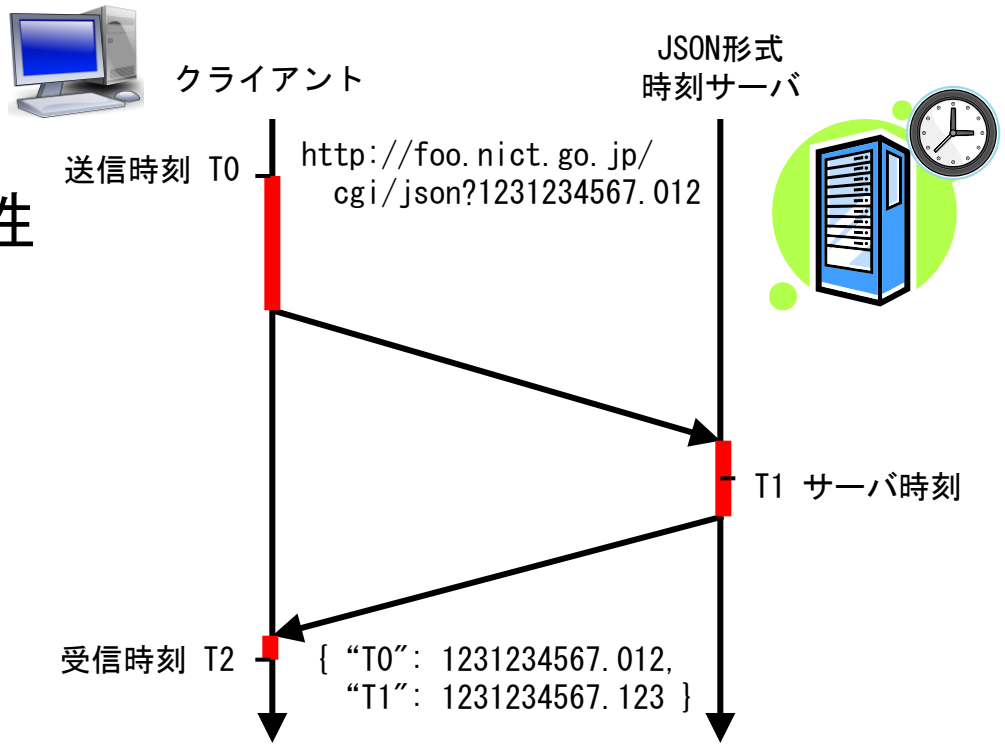
推定時刻差⊿T：
⊿T=(T0+T2)/2-T1

往復の遅延時間が同一と仮定した場合の、サーバ時計とクライアント時計の時刻差計算値

NiCT National Institute of Information and Communications Technology

推定時刻差⊿Tの誤差要因

- 往復の遅延時間差
  - エラー再送
  - 経路・処理時間の非対称性
- 時刻取得関数の粒度
  - getTime() : 16ms ?
  - (IE / Windows)
- 内部処理時間の非対称性
  - DNS ルックアップ
  - SSL 鍵交換（https の場合）



NiCT National Institute of Information and Communications Technology

時刻の誤差限界(1サーバ)

信じられるもの

◆ $t0 \le t1 \le t2$ ・・(1)

信じられると仮定するもの

◆ サーバの時計は正確である

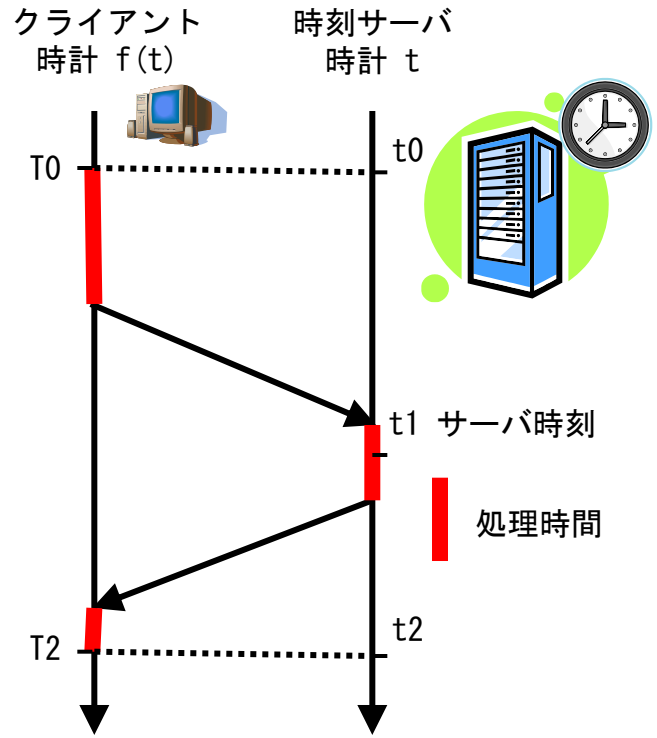
◆ $t0 \le t \le t2$ の期間では、クライアントPCのクロック誤差は無視できる

$f(t) = t + c$ (cは一定値) ・・(2)

◆ 時刻取得関数の粒度などに起因する誤差は、一定値 E より小さい

$|e0| < E, |e2| < E$ ・・(3)

(誤差が大きいOSでも、16ms 程度)



記録される情報:
送信タイムスタンプ: T0 = f(t0) + e0
サーバタイムスタンプ: T1 = t1
受信タイムスタンプ: T2 = f(t2) + e2
但し e0, e2 は、時刻取得関数の粒度などに起因する誤差

NiCT National Institute of Information and Communications Technology

時刻の誤差限界(1サーバ)

誤差の限界

(1),(2) より、

$$c \geq (T0 - e0) - T1 \quad \cdots (4)$$

$$c \leq (T2 - e2) - T1 \quad \cdots (5)$$

c の推定値を $(T0+T2)/2 - T1$ とし、
(4),(5) の両辺から引くと、

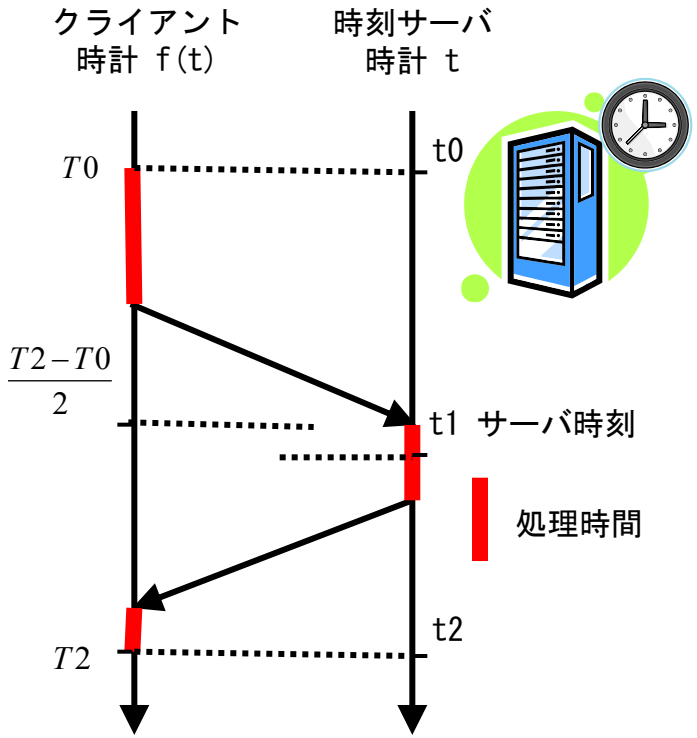
$$c - (\frac{T0+T2}{2} - T1) \geq -\frac{T2-T0}{2} - e0 \quad \cdots (6)$$

$$c - (\frac{T0+T2}{2} - T1) \leq \frac{T2-T0}{2} - e2 \quad \cdots (7)$$

(3),(6),(7) より、

$$\left| c - (\frac{T0+T2}{2} - T1) \right| \leq \frac{T2-T0}{2} + E \quad \cdots (8)$$

この右辺が誤差の最大値となる



記録される情報：
送信タイムスタンプ： T0 = f(t0) + e0
サーバタイムスタンプ：T1 = t1
受信タイムスタンプ： T2 = f(t2) + e2
但し e0, e2 は、時刻取得関数の粒度などに起因する誤差

NiCT National Institute of Information and Communications Technology

時刻の誤差限界(複数サーバアクセス)

信じられるもの

◆ $t0 \le t1 \le t2, t3 \le t4 \le t5$ ・・(1)

信じられると仮定するもの

◆ すべてのサーバ時計は正確である

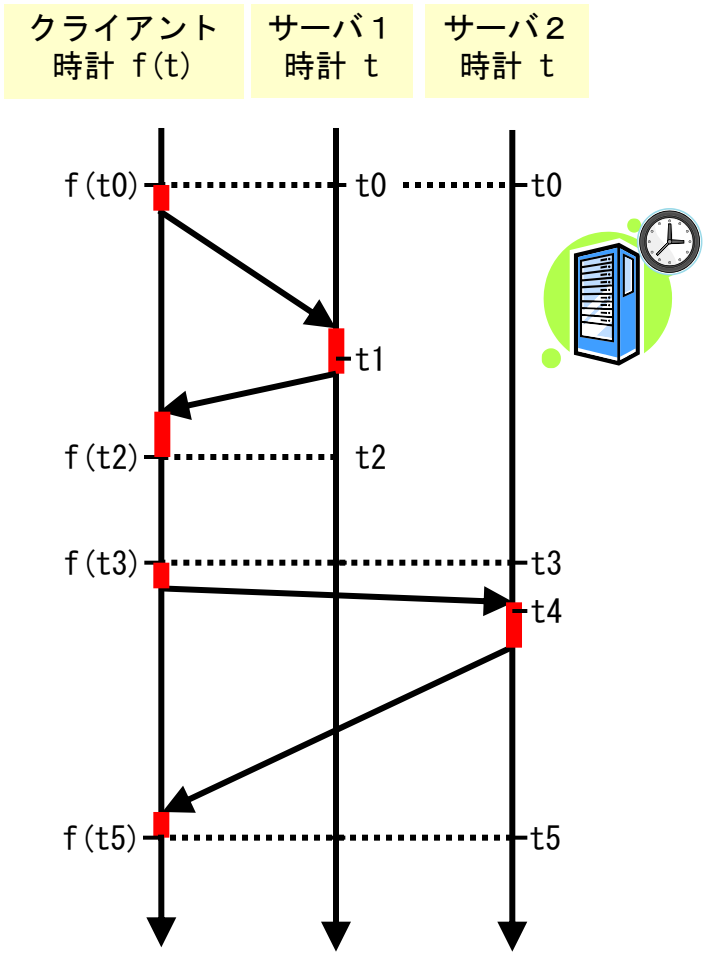
◆ $t0 \le t \le t5$ の期間では、クライアントPCのクロック誤差は無視できる

$f(t) = t + c$ (cは一定値) ・・(2)

◆ 時刻取得関数の粒度などに起因する誤差は、一定値 E より小さい

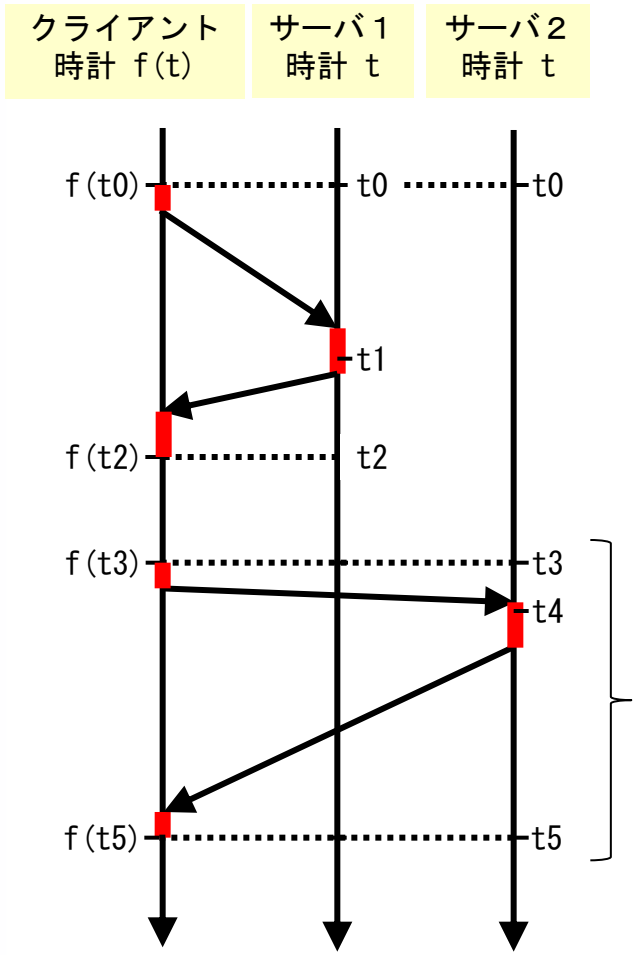
$|e0| < E, |e2| < E$ ・・(3)

(誤差が大きいOSでも、16ms 程度)

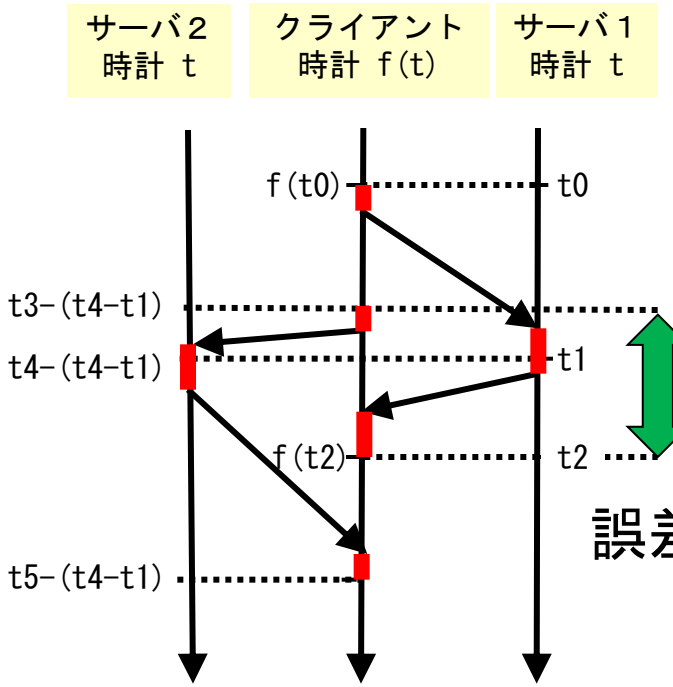


NiCT National Institute of Information and Communications Technology

時刻の誤差限界(複数サーバ)



(t4-t1)
移動



誤差の限界

$$t0 \le t1 \le t2 \quad \text{かつ}$$

$$t3-(t4-t1) \le t1 \le t5-(t4-t1)$$

NiCT National Institute of Information and Communications Technology

時刻の誤差限界(複数サーバ)

$$t0 \le t1 \le t2 \quad \text{かつ}$$

$$t3-(t4-t1) \le t1 \le t5-(t4-t1)$$

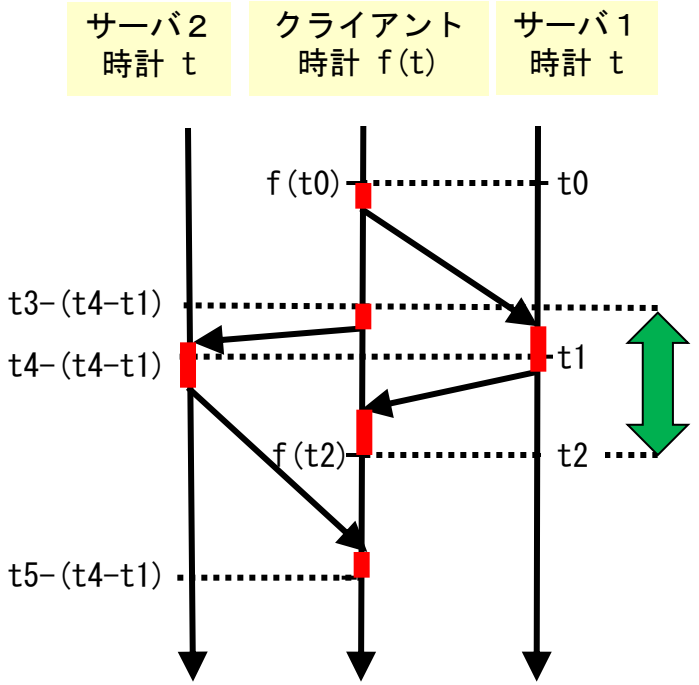
$$\min(t0, t3-(t4-t1)) \le t1$$

$$\max(t2, t5-(t4-t1)) \ge t1$$

クライアント時計の偏差 c の

推定値 : $\dfrac{\max(t2, t5-(t4-t1)) + \min(t0, t3-(t4-t1))}{2}$

誤差の最大値 : $\dfrac{\max(t2, t5-(t4-t1)) - \min(t0, t3-(t4-t1))}{2} + E$



記録される情報 :

送信タイムスタンプ :	T0 = f(t0) + e0
サーバタイムスタンプ :	T1 = t1
受信タイムスタンプ :	T2 = f(t2) + e2
送信タイムスタンプ :	T3 = f(t3) + e3
サーバタイムスタンプ :	T4 = t4
受信タイムスタンプ :	T5 = f(t5) + e5

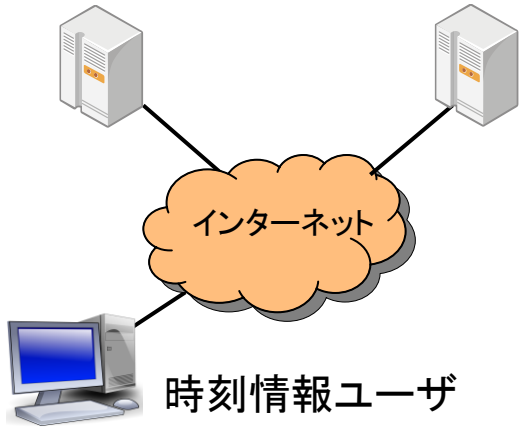
但し e0 ～ e2 は、時刻取得関数の粒度などに起因する誤差

NiCT National Institute of Information and Communications Technology

HTTP時刻供給サーバ(JSON形式)

通常の Web サーバに CGI プログラムを設置

ntp-a1.nict.go.jp　ntp-b1.nict.go.jp



http://ntp-a1.nict.go.jp/cgi-bin/json?1368668174.987

レスポンス

```
{ "id": "ntp-a1.nict.go.jp",
  "it": 1498765432.123,
  "st": 1498765432.234,
  "leap": 36,
  "next": 1483228800,
  "step": 1 }
```

物理的なサーバ固有の文字列

クライアントからの送出時刻

サーバの時刻

うるう秒関係情報：
次回(または前回)のうるう秒イベント(UNIX TIME=1483228800)は、うるう秒挿入(step=1)で、それまでの UTC-TAI は36秒

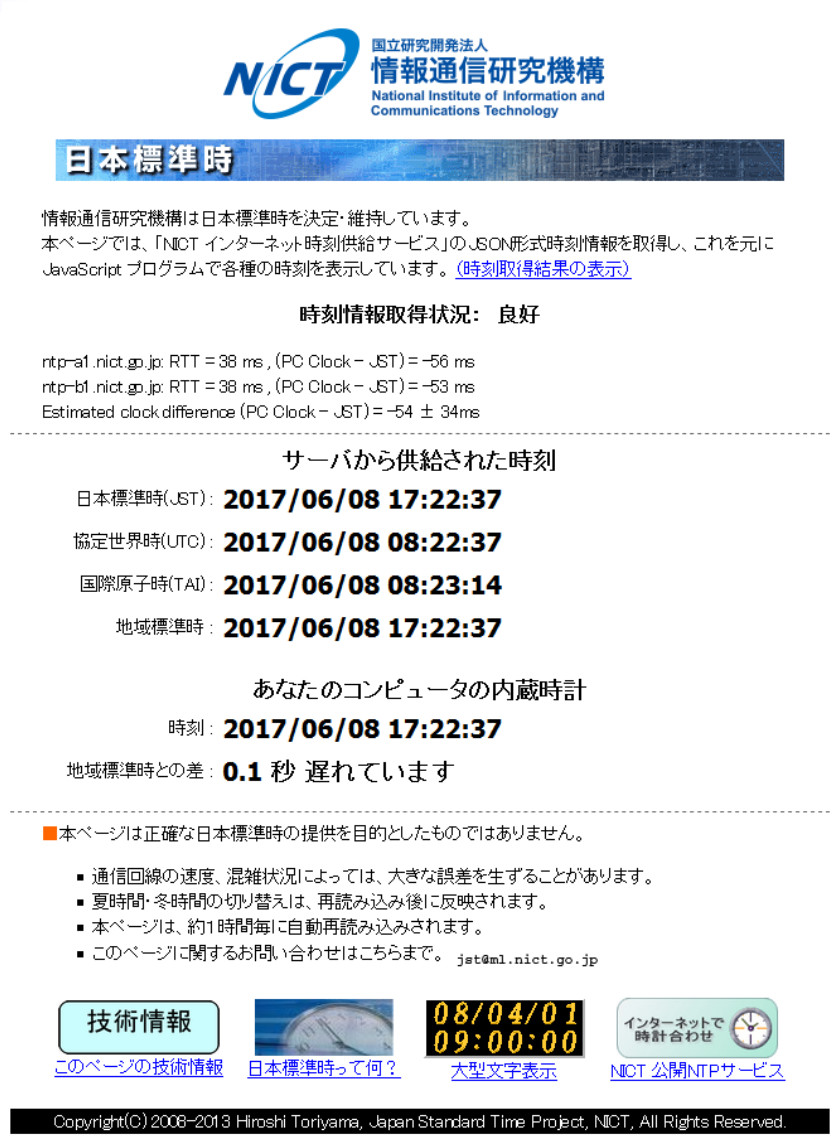
NiCT National Institute of Information and Communications Technology

提案方式の特長

- 伝送遅延時間による誤差の低減が期待できる
  往復の伝送遅延時間を相殺することにより、誤差の低減が期待できる。
- 時刻誤差の最大値を知ることができる
  「信頼できる時刻」には重要
- 特殊なサーバを構築する必要がない
  CGI プログラムの作成のみ
  SSLとの併用(https)が容易
- Javascript からの利用が容易
  JavaScriptからのクロスドメインアクセス、動的アクセスが可能
  時刻利用クライアントが CGI である必要がない

NiCT National Institute of Information and Communications Technology

提案方式の応用例(AJAX方式:2013年～)



複数のサーバ(3 URL以上)から、

- AJAX により2サーバにアクセスし、得られた時刻情報を比較し、信用できると判断される場合は、重複する誤差限界範囲の中心値を取得時刻として表示する。
- サーバからの応答がタイムアウトした場合、または、時刻情報を比較した結果、信用できると判断されなかった場合、信用できる2サーバの組が得られるまで、他のサーバにアクセスする。
- 上記の結果、信用できる2サーバの組が得られなかった場合には、その旨表示する。

複数サーバへのアクセスを省略した簡易版が、Sistema Interamericano de Metrologia (SIM) のトップページでも利用されている。
http://tf.nist.gov/sim/