

コンピュータのソフトウェア障害(例)

その1: 日付に関すること・・・いわゆる日付に関する「問題」について

2000年問題(西暦2000年, YEAR2000, Y2K)

(すでに済んだことではありますが、考えてみましょう)

原因(今となっては古いプログラムテクニックのため)

日付を下2桁だけで処理するようにしていたため 記憶容量を節約

(参考)

1994年を1とした場合の価格比	1981年	1994年
メモリ価格(単位当たり)	35	1
ハードディスク価格(単位当たり)	133	1

(当然、現在ではさらに価格比は大きくなっている!)

(参考2)

	1997年	2004年
128MB SDR SDRAM	¥50,000	¥2,500
1GB当りのハードディスク価格	¥10,000	~ ¥50

(最近の傾向ではあるが・・・実売価格で比較してみた)

Mooreの法則(情報システム概論)

それに従うように集積度は高まっており、同時に価格も下がっていている

影響範囲

- ・初期のコンピュータやプログラム
- ・埋め込みチップ 盲点だった

問題なのは

- ・時間がない
- ・技術者が足りない(いない)
- ・お金が掛かる この投資は、生産的な付加価値を生まない大出費 多くの人が他人事としていた

影響

- ・西暦2000年をまたいだ日付データの比較が困難化
- ・データの並べ換え(sort)の不正
- ・日付の帳票印刷処理
- ・1900年から2000年にまたがった日付の指定不能
- ・2000年を1900年と認識
- ・年齢/金利算出処理の不正
- ・和暦・西暦変換処理
- ・満了日付チェックの不正
- ・旧型のPOSレジ端末 00 クレジットカード未対応

結果

結果としては、それほど大きな影響はなかったのだが・・・

うるう年問題

閏年とは・・・

「西暦を4で割った余りが0かつ100で割った余りが0でない」

- ・4年に1回、西暦年が4で割り切れる年
- ・ただし、4で割れても100で割れる年はうるう年ではない
- ・ただし、100で割れて、かつ400で割れる年はうるう年

```
if (  
    (((year % 4) == 0) && ((year % 100) != 0)) ||  
    ((year % 400) == 0)  
)  
    leap = true;
```

GPS End of Week

原因

カーナビの位置確認に使用されているGPSはグリニッジ標準時1980年1月6日を基準に1週間ごとに週番号を加算して日付データを扱っている。

ところがこの週番号には10ビット(=1024)しか割り当てられていなかった。それを越えるとオーバーフロー(ロールオーバー)。

日時

1999年8月22日(日本時間午前9時)・・・最初のロールオーバー日

影響

- ・GPS受信機からデータが出力されない
- ・測位状態が安定しない
- ・測位状態は安定するが、誤った年月日や位置データが出力される
- ・不安定な測位状態が数分間続いた後、安定した状態となる

結果

- ・各メーカーは広告などで改修を呼びかけたが、浸透していなかった
- ・ユーザ側も甘く見ていた

全てが終わったわけではない

同じようなトラブル・問題は今後も発生する可能性がある