

西暦2000年問題とLinux

Y2K Problem and Linux OS

坂本 佳則*

Yoshinori SAKAMOTO

平成11年度は、2000年問題(Y2K)などもあり、コンピュータやネットワーク関連技術の周辺をめぐる動きとして、なかなかエキサイティングな問題が続出した。

当センターでは、特にインターネットと所内LAN部分の運営のほぼすべてを自主的運営でまかっているため、Y2Kにもほとんど自力で対応した。このとき行った作業はY2K対処方法として有効であるばかりか、ネットワーク運営上の危機管理手法としてそれ以外の問題にも活用可能であるので紹介したい。また、11年度の情報産業界のもう一つの話題であった、フリーソフトウェアのネットワークOSであるLinuxについて、所内LANでの運営に取り入れて運営実験を行った。後者については、Y2K対応にも役だったのでここに紹介する。

1. Y2Kとは何だったのか

世界中を巻き込んだ騒ぎとなった、コンピュータ2000年問題(Y2K)。いざ西暦2000年を迎え数ヶ月経過してみると、最も騒動が発生していた時期は、大方の予想していた「西暦が2000年に切り替わった瞬間、及びその後」ではなく、「その直前の数ヶ月」であった。

2000年を迎えた直後、また2月29日の前後において、確かにさまざまなトラブル報告が相次いでいたが、通常よりも敏感な報道体制により問題が大きく見えていただけで、そのほとんどが通常であれば報道もされなかったであろう小さな不具合であった。

もっとも、中には鉄道用券売機の立ち上げ不能や、携帯電話の不具合など、全国規模のものもあったし、原発の制御棒の位置表示システムの障害という、ひやりとする事件もあった。また、大半は小さな不具合であったとしても、対処は必要なわけで、これらに関しては、あらゆる組織が臨戦態勢で挑んでいたことによって回避できた問題も多い。そういったことを考えるとY2Kが本当に単なる空騒ぎであったと片づけてしまうこともできない。

1999年の阿鼻叫喚の中で、肝を冷やしながらなんとが行った対処が功を奏していたケースも少なくないはずである。また、報道があればほど繰り返されなければ、各組織のシステム担当者が対処のための費用を算出し、組織上部に要求したところで、過大な対処不要、費用対効果が不明などという理由で放置され、社会的にはもっと問題が大きくなっていったかもしれないのである。

ここでY2Kの騒ぎがなんであったのか、また、コンピュータを中心としたシステム危機対処手法はどのようなものとなるのか、またこの騒ぎの後で我々は何を学ぶべきなのか、

センターでの対処例の紹介を交えながら、今一度Y2Kで起こったことを振り返ってきたい。

2. 起こりえた問題

2-1 日付問題の与えるインパクト

既に各所で紹介されていて、周知のこととなった感があるが、西暦2000年問題とは、計算機や集積回路の内部などにおいて西暦の年数を指定するときに4桁ではなく下二桁で表記しているときに発生する不具合を指す。

コンピュータや集積回路など、何らかの情報処理を行う機械装置は1900年代に花開いた。1900年代においてはその年号を現すときに慣例として頭の二桁を省略することが多かった。1998年は「'98」という具合に...。(正確には、1900年代に限ったことではないとは思われるが...)この表記は計算機が高価で、それを構成する部品を少しでも節約したかったものにとっては魅力的に映ったのであろう。あるいは、帳票入力などの際の労力を少しでも減らして使いやすさを強調することは、ある意味それをビジネスとしている者として当然の手法だったのかもしれない。わかりきった最初の「19」は省略するのが効率的であると考えたのであろう。

この処理方式のまま1999年から2000年を迎えると日付関係の処理に不具合が生じる。問題を一番単純に説明しても、頭の二桁の数値を「19」だという前提に立った表記法なのであるから、「00」は1900年だということになる。現実には1年経過したはずなのに100年もさかのぼったことになってしまう。コンピュータなどの日付計算は通常、常に増え続けるはずの数字を相手にしているので、突然現在の日付が過去になってしまったようにみえたときの挙動は予想しがたい。マイナスの結果を判断して、コンピュータ

が異常終了するか、マイナスはあり得ないので「大きいほうから小さいほうを減ずる」というような処理になっているとすれば、奇妙な問題が起こることは予想できる。

この問題を放置すれば、2000年になった途端に、105才の老人の住所に幼稚園の入園案内が送られるようなことになる。ある金融機関から99年分の利子が要求される。そして、停止してはいけないサービスの管理を司るコンピュータが停止してしまうことになる。

2-2 ライフラインが狂う？

日付の計算というのは、コンピュータ処理のあらゆる場所で利用されているものなので、直接日付処理が目的ではない業務を行うコンピュータシステムであっても、たとえば何らかの記録を処理する部分が一週間の積算を計算しようとして、マイナスの数値に驚き、システムの基幹部分から異常停止するケースがありうる。

さまざまな人々が、さまざまな場所で異常があった場合に起こりうる状況を指摘した。

銀行は、日付処理を基幹業務の中核データとして扱う職種の一つである。銀行機能が停止することはすなわち世の中の経済基盤がストップすることを意味する。

発電所、浄水場などにも当然のごとくさまざまな処理がコンピュータ化され自動化している。万一、発電所がストップしたりすればその影響は計り知れない。何しろ、現在文明のほとんどのサービスは電力の供給を前提としているのだ。

こうしたライフラインを支える社会構造が崩れることを懸念した人々は、非常食、非常用燃料や発電器などの買いだめに走った。

2-3 コンピュータだけではない

日付の計算を行っているのはコンピュータだけではない。マイクロチップに日付を計算して返す機能を持ったものも多く存在する。また、日付だけを返すマイクロチップというものも存在し、コンピュータはいうに及ばず各種制御機械の設計に活用され続けているのである。

ということは、どの機械に問題のあるチップが使われているかを特定しなければならないことを意味し、これはコンピュータのようにソフトウェアの修正で済みそうなものよりもさらにたちが悪そうである。

1999年から2年間動きつづけるよう設定された自動機械は、2000年の到来とともに現在の日付が1900年であることを知り、動作を停止したまま、てこでも動かないかもしれない。

2-4 全システムのチェックは無理

2000年問題自体は1990年よりもさらに以前から指摘されてきた。にもかかわらず直前になってこれほどの騒ぎになっ

てしまった原因は、誰もが「少しずつ改善していけば、いずれは修正できていくのだらう」と考えていたところにある。「現在可動しているコンピュータシステムも、そのころには（対策済みの）新しいシステムに置き換わっているのだらう」という発想である。

しかし、現実には10年経っても使われ続けているシステムも存在していたし、1998年に新発売された著名なOSでさえ、2000年問題に完全に対応していたわけではなかったのである。

これは、歴史を積み重ねたことによる悲劇だと言える。問題を放置することが明らかに弊害を生むという時期になってから、改めてゼロから開発されたシステムならば対処を盛り込めたであろう。また、問題が明らかになった時点で1999年のように世界規模で取り組んでいけば、もう少し落ち着いた対処が可能だったのかもしれない。

ところが、新しいソフトウェアといえども過去からの積み重ねの上に成り立っており、簡単な動作確認により出荷されていくのである。

また、各種制御機器にしても、新製品は過去からの製品の改良という形で開発され、既にさまざまな（過去からの）機器が導入されている現場に組み込まれていくのである。

過去に導入された機器については、マニュアルや設計図すら手に入らないケースも多く、ひどい場合には製造元が既に存在しないこともある。

これらのものをすべて含めてチェックすることは、理論的には不可能ではないかもしれないが、非現実的な人員の投入と膨大な工程数が必要となる。また、現在の産業構造は内外で非常に複雑に絡み合ったつながりを有していることも多い。特定の範囲内だけ西暦2000年にして異常がないかテストしてみても、社会全体でそのテストを同時に実施してみないと結局何が起こるかかわからないのである。

「あと10年ある」時には、いずれ解決されるであろうと思われていた事象が、現在に至り徐々に取り返しのでない様相を呈してきたのである。

3 . はじまりは...

3-1 対処したいけどできない

この問題に関して担当技術者が座して何もしなかったわけではない。問題を提起し、システムの修正を図ろうとした人たちはたくさんいた。特に現場のメンテナンスに携わる人々は来るべき悪夢の時代を予感し、生きた心地がしなかったに違いない。

ところが、この世の中は技術者が動かしているのではない。技術者はシステムを動かしているのだから社会は動かさない。各種機器やソフトウェアの修正には費用がかかるのだ。しかし、10年先や5年先の問題のためにいつ費用を投じるべきかという経営判断がそこに入って問題をややこしくする。上層部の理解を得られずに、予算がつかず、

対応しようにもできない状況が続いた。

先駆的な技術者たちは、自分たちが必要なことができるように動きはじめていた。インターネットの存在は彼らの心強い味方である。世界中で、Y2Kの及ぼす社会的影響が情報交換されていった。Y2K FAQ (Freequency Asked Question,よくされる質問)という文書がインターネット上でメンテナンスされていたが、この文書中のメッセージには1989年の文章も納められている。

こうして、上司の理解を得られなかった技術者たちが社会に警鐘を与えはじめていった。

3-2 2000年が来る前に・・・

2000年問題に最初に直面した組織がどこであったかは定かではない。しかしながらそれは明らかに1999年よりも以前に存在した。

ある食品メーカーでは、在庫期限が5年の真空パック商品を製造していた。1995年に製造した製品の在庫期限を計算した彼らの在庫管理プログラムは、「正しく」計算し、在庫期限を00/MM/DDと登録した。

次に在庫管理システムが稼働したとき、在庫期限が00/MM/DDの商品を発見したプログラムは、この95年も前に在庫期限が切れたと主張している商品を破棄してしまったのである。ただ、この食品メーカーが幸運だったことに、このシステムにおいて現実に廃棄作業をするのはロボットではなく、人間の作業員であった。何かおかしいと感じた現場の作業員によって、システムの異常が発見できたという。

同様の事件は、たとえば返済時期が2000年を越えてしまうような金融システムなどで散発的に起こっており、そのたびにシステム停止と修正が繰り返されていた。

4. ネットワークとY2K

4-1 工技センターの直面した問題

公共機関においては、予算立案は執行の前年度前半に行われるのであるが、その際に必要性などを説明するに十分な資料を揃えておかなければならない。しかも西暦2000年は1999年度中に到来するのである。計画は遅くとも1998年前半まで、計画を立てるのに必要な資料自体は1997年中に必要だったのである。

正直なところ日本において、各種メーカーのY2Kに対する明確な表明が出始めたのは、1998年後半以降であり、潤沢に情報がある状態に至ったのは1999年になってからだった。遅きに失していた感がある。

センターのような公共機関は、計画的にシステムの置き換えを行えるような資料を揃えることができず、1999年度を迎えてしまっていたのが実情である。

さて、工業技術センターにおいても電源関係の管理システムや、電話交換機システム、会議室予約システムなどに

おいてY2Kの問題を抱えたシステムが散見された。このうち、会議室予約システムについては廃棄した上で、同等の新システムを独自に開発・運営することで対処することにし、あとはメンテナンスの業者へ依頼した。

また、各種計測システムや研究開発用の制御システムについて、そもそも連続的な運転を必要とするシステム以外は、日付問題が存在しないので除外できる。連続的な運転であっても、累計時間の管理しかしていないものは関係ないため、大部分の機器について対象から外すことができた。

残ったのは、所内各所にあるコンピュータである。すべてのコンピュータ上のソフトを置き換えることは予算的に不可能であったし、個々のアプリケーションについてはそれぞれの専門ソフト会社の対応を待つより手がなかった。

ただし、たとえばファイルの日付が狂うかもしれないだけで、必要とする計測データ消失の恐れや、システムクラッシュの恐れまではないものが大半を占めることから、そのまま放置して本当に問題が出るかどうかを見極めるより手がないと判断せざるをえなかった。

そして、最終的に残されたのが本当に24時間の連続運転を行っているインターネット関連の各種サーバと所内LANシステムであった。

4-2 インターネットの活用

Y2K対応をネットワーク関連機器に限定することにして、どの部分に改良が必要なのかを考察する必要があった。幸いにして、ネットワーク関連のハードウェアやソフトウェアを提供しているような企業はたいていインターネットに情報提供の窓口を持つものがほとんどである。センターで導入している各種機器やソフトウェアなどについてのチェックはすべてこのインターネットのチャンネルを通じて行うことができた。

ネットワーク運営でY2Kの影響が関係ありそうなものは次のようなものであった。

- ネットワークOSそのもの (UNIX, WindowsNTなど)
- WWWサービスのアクセス管理
- 電子メール/ネットニュース/DNSなどインターネット関連
- メーリングリスト処理プログラム
- ファイルサーバ及びバックアップシステム
- Windowsファイル共有サポートシステム
- ルータ
- スイッチングハブ

から については、いずれも基本的にネットワークOSの上になりたっているので、ネットワークOSそのものにY2Kの問題が存在しているのは、何の解決にもならない。特に本センターでは、 と については、ネットワークOSの機能をそのまま活用しているので、直接的な影響を被ることになる。それ以外のシステムについては、基本的に

インターネットで公開されているフリーソフトウェアの活用によって提供している機能である。

と については、専用ハードウェアである。

それぞれの機能について、インターネットで公開されている状況をチェックしていった。

4-3 OSアップグレード

Y2Kに限らず、最近では各種ソフトウェアの改良パッチがインターネット上で配布されている。大きなバージョン番号の変更をとまなわれない小さな変更であれば、無料配布であることが多い。現実には、インターネットに関する新しいセキュリティ問題が発覚した場合などは、その部分の穴を塞ぐための小さな変更パッチを無料配布するということは日常茶飯事に行われている。

Windowsシリーズ(95,98,NT)については、この方法で対処できた。バージョンアップではなく、機能アップをとまなわれない不具合修正であるため無料配布を行ったようである。

ところが、センターのLAN運営、インターネット運営を行っている根幹のサーバであるワークステーションのOS、UNIXはバージョンが古すぎてパッチによるY2K対応が不可能なことが判明した。

経理関係担当と相談した結果、どうしても停止させると業務に支障を来しそうなファイルサーバのOSバージョンアップのみを行い、それ以外については自助努力で出来る限りの対処を行うことにした。

4-4 フリーソフトウェア

WWWサーバ、Windowsファイル共有サービス、メーリングリスト処理ソフトなど、フリーソフトウェアで運営しているものは、ソフトウェアの配布を行っているオリジナルサイトで詳細な情報を得ることができた。こういうときには、そもそもインターネットでの配布を旨とするソフトウェアは心強い。Y2Kに関しては、そのソフトウェアが動作するOSでの対処は不可欠とした上で、さらに細かい部分の不具合が報告され、逐一修正されていた。いったん修正したつもりのもので、まだ修正不足であったこともあったが、随時変更を追いかけていくことで対処が可能であった。

この「随時変更を追いかける」という行為は、ネットワークセキュリティ関係の不具合が発見され、修正が行われる場合にも必要であるので、おろそかにしない姿勢が大切である。

4-5 ファームウェアアップグレード

次にハードウェアの対処である。幸いなことに、ルータや、スイッチングハブというインテリジェントな機能を備えた機器類は、複雑な処理を内部的にソフトウェアで行う

設計になっている。このソフトウェアは、ファームウェアと呼ばれ、そのハード専用ではあるが、何らかの不具合が発覚したときに入れ換えることができるように設計されている。

今回のルータ、スイッチングハブは、このファームウェアの入れ換えが可能な機種であり、かつインターネットでその入れ換えソフトと手法が公開されていた。

英文ではあったが、この情報のおかげで自力で対処することができた。

4-6 ノンストップサービスへの配慮

OSの入れ換え作業は単なるマイナーバージョンアップのパッチ修正とは異なり、システムをいったん白紙に戻し、再構築する作業が必要となる。これは定常運用しているシステムにとって過酷な試練である。

あらゆるものをいったんバックアップして、構築してから再び元にコピーしなおせばよいという単純なものではない。OSのバージョンが異なっているのであるから、設定ファイルの形式が同じであるという保証はないし、データについては、運営しながらバックアップなどしていたのでは、最新のバックアップである保証はない。

また、インターネットとの接続サービス部分についても同様である。電子メールなどは文字どおり24時間対応で、いつメールが飛び込んできても受け取ることができなければならない。また、DNS(インターネット上の名前解決の手段)の停止はインターネットの世界からセンターが消失したかのようにみえる事態を招く。

いかにしてシステム停止時間を最小限にとどめながら、多くのサーバを一度に入れ換えるかが焦点となった。より単純な手法としては、運営の停止はしかたないものとあきらめ、所内外各所にアナウンスして頑張るという方法である。現実には他府県より一時的な運営停止の連絡が相次ぎ、この手法をとった組織は多かったことがうかがえる。

Y2K対応の内容としては、OSレベルからの入れ換えが発生することから、サーバのリプレースに相当すると考えればよい。サーバのリプレースを行うとき、いかにしてサービスの停止期間を少なくできるかという問題に置き換えた考察を行い、計画を立てた。

センターが運営するネットワークサービスは、インターネット接続はいうにおよばず、所内LAN向けのサービスも全てIP(インターネットプロトコル)ベースで行っている。この場合、コンピュータの特定はIPアドレスをベースに行われる。そこで、次のような手順でサーバの置換を計画してみた。

1. 現在行っているサービスと全く同等のサービスが提供できるサーバを用意する。
2. 現在行っているサービスのデータを代替用サーバに移行し、動作を確認する。

3. 従来サーバを停止後、速やかに代替用サーバのIPアドレスを従来サーバのアドレスに変更する。
4. 従来サーバを従来とは別のIPアドレスに変更してY2K対応を行う。

予備サーバの確保さえ出来れば、この手法を行うことができる。幸い、後述するPCベースのUNIXシステムのLinuxベースのサーバを構築することで、これに対処することができた。

4-6 経費ゼロの対処

センターに課せられた試練は、これらの作業をほとんど経費ゼロで実行しなければならないことだった。今回、インターネット上でどこまで情報が公開されているかという調査を徹底的に行ったことで、劇的な経費削減が可能となった。調査をよくせずに委託による対処を選択してしまうと、いきなり人件費に関する経費が発生してしまうため、見積金額に驚くことになる。

経費を押さえようとする場合には、「自力による対処」がどこまで出来るかをよく調査することが肝要である。今回はY2K対処が要件であったが、新しいウィルスへの対処などシステムアップグレードの必要性は何かにつけて出てくるものである。組織ごとに事情があって対処不可能な場合はしかたないが、できる限りコンピュータやネットワーク関連の専門部署を置き、自主的対処の可能な態勢を整えておくことが望ましい。

5. 現実に起こった問題

注目されていた、年末年始と2月29日のいわゆる閏日の問題となる時期を経て、結局社会的な大混乱は起こらなかった。確かに指摘されたように、謝った日時を表示したシステムも存在したし、停止してしまったシステムもあったが、比較的短時間で復旧された。

気象情報システム、アメダスの機能停止、閏日における郵便局のATMの異常停止はある程度国民の関心を集めたが、経済が混乱するほどではなく早期に復旧された。病院や役所の証明書発行システムなどで日付が誤って表示されたものも存在するが、手作業の対応で事足りた。その他、年末年始を返上して「事件」に備えた組織は多く、担当者によって様々な不具合が報告されていたが、中には「本当にY2Kかどうかは不明」とされるものも少なくなかった。通常運転しない年末年始に試験運転されたことによる不具合もあったと思われる。

1998年9月に策定された高度情報通信社会推進本部の「コンピュータ2000年問題に関する行動計画」を皮切りに各所で行われた問題に対する対処方法の広報もいきなり、それを元にしたシステム改修が十分であったのか、最終的

には現在の社会基盤はそれほどコンピュータに依存しておらず、いざというときには手で対処可能なものであったのか、おそらく両者のかねあいで比較的うまいところに軟着陸したのではないかと思われる。本問題に1000億ドルを投じたと言われるアメリカに比べ、日本は300億円程度の投資だった*1とされることから、これでも経済的な対処を成し得たのではないかという評価もある。

6. Linux を使ったネットワークサービス

6-1 自由によるこそ

ネットワークサービスをストップさせずにサーバのリプレースを行うために、IPアドレスの変更作業をベースにしたサーバ切り替えの手法を使うことは前に述べた。この場合、通常運営を続けながら予備のサーバに切り替えるわけであるから、現在用いているサーバと同等のネットワークサービスを行うことができるコンピュータが一台余分に必要になる。

幸い、1999年は、フリーソフトウェアのネットワークOS、Linuxが注目された年であった。Linuxは、フィンランドのヘルシンキ大学でLinus Torvalds氏によって、1991年にインターネット上に最初のバージョン(0.02)が公開されたUNIX系のOSである。以来、インターネットを通じて数多くのプログラマの助力により、飛躍的な発展を遂げてきた。

Linuxは、自由に再配布することができるソフトウェアであり、筆者が1994年当時に試用した時点で、すでに本格的なUNIXと同等の機能を発揮していた。Linuxが動作するコンピュータアーキテクチャは、DOS/VパソコンのIntel系のコンピュータ以外にも、Motorola 68k, Digital Alpha, SPARC, Mips, Motorola PowerPC など多岐にわたる。

コピー自由、利用自由、改変自由のこのネットワークOSは、以前から技術者の間では話題になるという時期を過ぎ、すでに定着していたが、1999年は一般商業雑誌がこぞととりあげたことによって、一般の人にも広く知られることとなった。

UNIX系であり、育まれた土壌がインターネットであるため、当然のことながらインターネット関係のサービス機能は充実している。さらに、通常のDOS/Vパソコンはるか、センターでインターネットサーバとして使っている、Digital Alphaのワークステーションでも動作する。もちろんインターネットからインストールイメージをダウンロードすることによって、経費はゼロで最新OSが入手可能である。

このような理由により、センターのリプレース用OSとして最適であると判断した。

* 1 我が国の場合、昭和から平成への元号の変更に際して、Y2Kと同等の対処済みであったとする意見、300億円以外に最初から対処と称せずにシステムのリプレースを行った組織も多いとする意見もある。

6-2 無料で出来る範囲

Linuxとは、正確にはOSのカーネル部分だけを指す。カーネルの機能は、コンピュータの各種ハードウェアのアクセスや、メモリ管理、プログラムの実行のためのファイル入出力など本当に最低限でありながら必要不可欠な基本機能に限定される。Linus Torvalds氏が最初に開発し、さまざまな人々によって拡張されながら、Linus自身が中心となってとりまとめてきたソフトウェアは、このカーネルだけである。

ファイルの編集を行ったり、ディスプレイに絵を描画したり、ネットワークサービスを行うなどのさまざまな機能はカーネルの上で動くソフトウェアで実現される。

Linuxはその利用が自由であったために、さまざまな人々によって周辺ソフトウェアが整備され、今では多機能な統合OSとなっている。そして、その周辺ソフトウェアもフリーソフトウェアで固められている。LinuxがGNUライセンス^{*1}によるフリーソフトウェアだったこともあって、Linuxを含めてその周辺ソフトウェアは、その配布と利用のみならず、ソースの閲覧と改変も自由である。こうして、世界中の不特定多数の人々がさまざまな改変を加えて発展するLinux OSシステムという構図が確定した。

Linuxカーネルを中心として統合化された環境はひとかたまりで配布され、「ディストリビューション・パッケージ」という形で入手することができる。ディストリビューション・パッケージを作成するのは誰か一人のこともあるし、グループや企業であることも多い。目的や手法の異なる組織などによって複数のディストリビューション・パッケージが出現しており、そのほとんどのものがインターネットから無料で入手可能である。

こうして統合化したネットワークOSのパッケージ形態までのものがすべて無料で配布され、入手することができるが、そのすべての行為がボランティアベースであるということではない。現在、一番利用者が多いとされているディストリビューション・パッケージのRedHatは、その開発母体は営利団体の企業である。彼らはOSシステムを組み上げる努力と、バージョン管理システムなど必要部分の開発を有償で雇った社員を使って行いながら、その成果を無償で配布し、なお利益を上げている。

じつは、RedHatは、確かに配布したソフトウェアに対して無保証を宣言しているが、ユーザが何らかの保証を得る手段を禁じているわけではない。企業自体は、情報関連業界に貢献しているというインパクトによるネームバリューを武器に、システム構築や、メンテナンス、技術サポートの請負を通じて糧を得ているのである。

もし、ユーザがRedHatが配布する無料のOSシステムを使いこなすために十分な技量を持っていなかった場合、対価を支払うことによって、彼らの技術サポートを受けることができる。また、OSだけでは十分でないシステムの構築の依頼をすることもできる。

つまり、入手および自主的な利活用については全くの無料で行うことができ、かつ対価の支払いによるサポート依頼の道も残されている。

6-3 Y2Kへの対処の様子

UNIXをベースとしたシステムについては、その中心となるカーネル部分では、内部的に日付情報の管理は32bit整数で現されており、2000年ちょうどに何らかの不具合が起こる可能性はない。LinuxはUNIXベースのOSなので、そのカーネル部分では最初から2000年問題は発生しない。

ただし、これまで述べてきたように、Linuxシステムとしてとらえた場合、それは様々なユーティリティソフトウェアの組み合わせによって構成されているため、個々のソフトウェアについて固有の問題がある可能性がある。

ここで、Y2Kが世間一般で頻繁に取りざたされる以前からインターネットでは様々な情報交換が行われていたことを思い出してほしい。そして、Linuxは他でもないインターネットで育まれたソフトウェアである。そして、GNUライセンスにより誰にでも開放されたソフトウェアであった。Linuxシステムは今となっては数え切れないほどのソフトウェアの組み合わせで構成されてはいたが、これまた数え切れない人々のチェックとメンテナンスにより、次々と修正が施されていった。

ユーザは、常にディストリビューションの最新パッケージのリリースと、アップデートモジュールのリリースをチェックしていればよかったのである。もし、何らかの不具合を発見すれば、ユーザ自身もその報告をすることによって、パッケージの改善に協力することができた。

6-4 Linuxに救われたインターネットワーク運営

実際にセンターがネットワーク運営のY2K対処を行った頃には、すでにLinuxシステムはY2Kに関して、かなり安心できるレベルに達していた。

そして、経費的にOSのバージョンアップで対処できないSUN Sparcワークステーションや、Digital Alphaワークステーションにさえ、Linuxをインストールし、幸いにも、同等のサービスを継続することができたのである。

特別に所内ファイルサーバに関しては、そのバックアップシステムと無停電電源装置(UPS)のサポートという点で、メーカー製のOSでなければならない事情があり、メー

* 1 Richard Stallman によって設立された Free Software Foundation によってメンテナンスされているソフトウェアのライセンス。日本語による正式名称は「GNU 一般公有使用許諾書」。フリーソフトウェアがユーザー全てに対してフリーであることを保証するために考案された。作者が有する著作権により、その著作物について、複写や改変の自由が阻害されないように制限を加える。条項には、それを手にしたユーザーに第三者への無償配布や改変の許可を与えることを義務づけ、独占を禁ずる内容が含まれる。

カー製OSのバージョンアップというやむを得ない手段を執ったが、その他のネットワークサービス、特にインターネット関連のサービスについては、完全にLinuxに置き換えることが可能であった。

5. Y2Kは終わっていない

5-1 問題の先送り

Y2Kへの対処には、様々なレベルのものがある。UNIXシステムなど、根本的には西暦を2桁では処理していなかったものは、エンドシステムで2桁の処理をしていた部分を4桁に修正すればよい。ところが、そのシステムの根本設計の部分で西暦を2桁という前提で設計されて運営されてきたシステムに4桁拡張を施すと旧来からのデータ形式との互換性の問題を生む。

ソフトウェアや処理機械が送受信するデータの相手が組織外にも多数存在する場合、システム全体の修正は事実上不可能になるということもある。

センターのようにソフトウェアの置き換えという処理で対応できるところはよいが、専用ソフトでがっちり固められたところは、旧来のシステムをばっさり捨てるわけにはいかない面もある。OSのバージョンアップにより動かなくなるアプリケーションソフトは枚挙にいとまがない。

たとえばこのような場合、1970年くらいを境に考え(基準年) 00から69までを2000年~2069年、70から99までは1970年~1999年と考えて処理するというアルゴリズムを組み込んで対処した例が多い。

この対処により2000年ちょうどに起きる問題は回避できたが、要するに今回いわれていたと同様の問題を2070年にまで先送りしていることになる。

さらに、たとえばWindows等でも同様の処置が行われているが、この場合1930年を基準に設定されており、破綻するのは2030年となる。それまでには、地球社会上のシステム全体の置き換えが完了することを祈りたい気持ちである。

5-2 より深刻な2038年問題

先に、UNIXベースのシステムには、根本的には2000年問題は存在しないはずであると書いた。これは次のような理由による。

UNIXはC言語というプログラミング言語で記述されたOSである。C言語で日付を取得するために用いる場合、time_tという名前の変数を参照するという形で取得する。このtime_tの仕様が問題で、1970年1月1日午前0時からの経過秒数を表している。そして、その変数の型は「符号付き整数(Unsigned Int)」となっている。C言語の規定ではこれは32bitで表現される。これは、-2147483646から2147483647までの整数が表現できる型である。

ところで、1970年1月1日0時0分0秒から、2147483647

秒経過した日付はいったいつか? 2038年1月19日12時14分7秒である。この時刻を1秒でも過ぎるとこの仕組みは破綻する。意外に近い。上で述べた2070年よりも早い時期にそれはやってくる。Y2Kの経験を元にして考えても、2038年より先の計算ができないことにより支障を来すという時期はそれよりも以前にやってくる。

Y2Kを乗り越えたんだから、今度もうまくやっていけるだろうと考えてはいけない。この問題は、Y2Kの時よりも根が深いのである。

すぐに思いつく解決法では、time_t変数を64bitに拡張することだが、これは各プログラマや企業が勝手に行ってはいけない。C言語の仕様というのは国際的な規格で定められるものであるので、独自仕様を勝手に出すわけにはいかないのである。そして、このC言語で作成されたプログラムというのはUNIXをはじめとして、それこそ至るところに偏在しているのである。UNIXだけを考えたとして、世界規模のコンピュータネットワーク、インターネットの運営がUNIX系のOSに支えられていることを忘れてはいけない。そして、それらのOSの上で動いている各種ソフトウェアは、C言語で開発されていることが多い。これらをすべて置き換えていく必要がある。

あと38年もあるとんびりしては、この問題を子孫に先送りしているだけになってしまう。国際的で計画的な取り組みが必要である。

5-3 Y2Kは技術的な問題か

プログラマが、年号の処理を2桁で行うことを思いついたとき、心の中で「2000年までしかうまく動かないけど・・・」と思わなかったはずはない。本来4桁の数字を2桁で表している状況をみれば、子供でもそこに気がつくはずである。

だが、必ず顧客から要求のある、システム納入経費の削減(昔はコンピュータのメモリーはとても高価であった)と、利便性の追求により、2桁処理を採用した。そして、密かに西暦2000年が訪れるころには、これから納入するシステムは時代遅れになっており、2桁の処理の問題も修正されたシステムが稼働していることだろうと考えていたのではないか。

ところが、現実にはY2Kが影響を与え始めるような時期になっても、西暦年号2桁処理のプログラムは生産され、市場に投入され続け、あるいはリプレースされることなく使い続けられた。

Y2Kは、技術的には比較的稚拙な問題である。システムの寿命を考慮し、それまでに必要十分な資源を割り当てておけばよかつたのである。ところが、それが設計された時点から何年も経過しても見直されることなく、新しいシステムにそのままの形で移され市場に蔓延していった。ソフトウェアという技術の性質上、物理的限界を超えて未来にわたり存在しうることの判断が抜けていたのではないか

と思う。

しかし、それにもまして、現場の技術者が危機感を覚えているときに、それを問題として取り上げることができなかった組織が多い。対策期間が十分にとれていれば、計画的な機器のリプレースで対処できたところを、社会的問題に発展してから一度に行おうとして、その作業の影響の大きさや費用に驚いて、結局パニックに陥った組織も多い。

つまり、経営の問題が顕在化したケースが多いと思われる。技術者の覚える危機感は技術上の問題であり、それが組織全体に及ぼす影響への想像力には限界がある。また、たとえ想像がついたとしても、現実にはどうすることもできない。経営者は現場の危機感を正確に受け止め、それが自組織やあるいは他組織にまで影響を及ぼす可能性について熟考し行動に移す必要がある。

6. Y2Kから学ばねばならないこと

Y2Kによって浮き彫りになったことは、現代社会がコンピュータに依存しているかどうかという事実よりも、コンピュータに依存しすぎているのではないかという人々の恐怖感である。

現実には、コンピュータを中心としたシステムは未だに些細な機能停止を繰り返しているものであり、そのたびに人手による対処や修正が行われている。

2000年1月1日以降に多数の不具合が報告されたにもかかわらず、ほんの数件をのぞけばその大半が即座に対処が行われ、現実の業務自体には大きな影響がなかった。

たとえば、Y2Kがまだ訪れる以前によく噂されたこととして、世界各地の核ミサイルの制御システムの異常による自動発射が行われてしまうのではないかという話があった。しかし、現実には核ミサイルの発射に至る全行程がすべてコンピュータによる全自動システムであることはないという。すべてを一つの自動システムに任せた運営は危ないということが徹底されている。

また、危険性が指摘されていた大規模なパニックも、我が国においては起こることもなく、平静に時が流れた。これについては、行政機関の広報が行き渡っていたという事実もあり評価したい。

ネットワークやコンピュータシステム関係の技術者は、確かに残業の連続で悲慘な思いはしたが、囁かれていたように不足すぎて社会混乱が起きるほどではなかった。

では、あれは単なる空騒ぎだったのか？我々は通常通りの生活をおくっていれば、あの事件から学ぶものはなにもないのか？

ここで、2000年ではなく、1999年を振り返っていただきたい。来年になれば何か起こるかもしれない。確かに自分は何も対処していないが、そのとき感じた恐怖を思い出していただきたい。結果として空騒ぎで終わったところも多いかもしれない。あるいは、必死で対処して事なきを得た

ところがほとんどだろう。1999年では遅かったと思うが、それでもあれほどの社会現象になったので、緊急措置が可能となっていた。センターが直面した問題のように、開発元からの明確な情報が提示されずに、前もっての対処が停止してしまっていた時期が存在したのは問題だったと思う。開発元からの情報が提示されていなかったとしても、技術者による可能性の提示だけで、ある程度の対処が可能な体制の整備が必要ではないかと思う。外注だけに頼りすぎない体質づくりなどである。

Y2Kが私たちに問いかけたのは、自分たちが普段から使っているシステムが異常を来したときに、いかに組織を機能させるかという命題ではないか。また、異常が起こることの予測が発生したときに、いかにその情報を取り上げ、的確に予め対処していく組織の体質ができているかということではないかと考える。そして、皆が不安を抱いたシステムの大規模な混乱が起きなかったのはなぜかを熟考し、そこに学ぶべきは学んでおくべきである。

西暦2000年は到来した。しかし、Y2Kと同等の問題はまだ残されたまま、解決しなければならない問題も多い。また、問題の質は違っても、似たような事件が起こることは考えられる。非常時の対処、危機管理という視点で、この経験をふまえた組織作り、経営体制を今一度考え直していただきたい。