



## 2001年武田賞 選考理由書

情報・電子系応用分野におけるテクノアントレプレナーシップに富む工学知の創造と活用

### 授賞業績

### オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

### 受賞者

#### 坂村 健

(オープンアーキテクチャ方式によるTRON仕様の開発と普及)

#### Stallman, Richard M.

(フリーソフトウェア概念の確立とGNUオペレーティングシステムの開発)

#### Torvalds, Linus

(オープンソース開発方式によるOSカーネルLinuxの開発)

アルファベット順に記載

賞金の3分の1ずつを上記の受賞者に贈る。

## 選考理由

情報・電子系応用分野の2001年武田賞は、オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践に貢献した坂村健、Richard M. StallmanおよびLinus Torvaldsに贈る。この業績はコンピュータの基本ソフトウェアであるオペレーティングシステム(OS)等の開発において、オープンな方法を用いることにより優れた成果を得たものである。

坂村は、1980年にリアルタイム組込み用OSであるTRON(The Real-time Operating system Nucleus)の基本構想を提案した。その提案に基づいて、誰でも参加できるオープンな組織としてTRONプロジェクトが1984年にスタートした。TRONの特徴は、リアルタイム組込み用であることと規模や種類の異なるプロセッサに実装することを考慮してインタフェース仕様を規定し、「オープンアーキテクチャ」として公開した点にある。決定した仕様の公開、メーカーでの実装、実装結果に基づく改良仕様の検討というサイクルにおいて、多くのユーザの衆知を集める

## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

「オープンな考え」が採用されている。その結果、TRON仕様は広く採用され、家電機器、携帯電話、自動車など多数の組込みシステムに適用されることとなった。

Stallmanは1984年にUNIX互換のコンピュータ用OSシステムをフリーソフトウェアとして作り始めた。これがGNUプロジェクトのスタートである。GNUは、「GNU's Not Unix」の再帰頭字語である。フリーソフトウェアの考え方はソースコードの公開、プログラム改良やコピー配布の自由、改良後のプログラムの公開などを含むものであり、これらの自由と公開性によって衆知を集めることが可能になった。GNUプロジェクトの成果として、高機能エディタ、コンパイラ、グラフィカルユーザインタフェースなどが作成され、その高機能性と高信頼性がフリーソフトウェアに対する高い評価をもたらした。

Torvaldsは、1991年にLinuxの最初のバージョンを独自に作成し、インターネット上に公開した。Linuxはコンピュータ用OSの中核部分(カーネル)である。その後のLinux開発は独自のスタイルで行われた。すなわち、公開されたソースコードの改良・デバッグには世界的規模の多数のボランティアが直接参加して提案を行い、提案の採用可否はTorvalds自身が決定するという手法である。頻繁にバージョンアップを繰り返すことにより、自主的で熱意ある作業がスピーディーに行われ、高性能で高信頼性を有するOSカーネルが生みだされた。

LinuxとGNUソフトウェアとの組み合わせにより、完全なUNIX互換のOSが誰でも自由に使えるようになった。Linux及びGNUソフトウェアは、その性能及び信頼性の高さとシステム開発に必要なコストの低さの故に、パソコンやサーバをはじめ一般的な用途のコンピュータシステムで広く使われ始めている。

以上述べたように、TRON、GNUおよびLinuxの各プロジェクトは、情報化社会の要となるコンピュータ基本ソフトウェアの開発において、オープンな開発スタイルあるいはオープンな利用の仕組みという新しい手法による解決方法を生み出した。その結果、広範な人々の知恵を集めることが可能となり、これまでになく高いレベルの成果をもたらしてコンピュータ利用の新たな発展をもたらした。さらに、この新しいオープンな手法は、新しい開発モデルを提示してコンピュータソフトウェア業界におけるそれまでの市場の常識を覆すような影響を与えている。すなわち、通常の市場経済とは異なる仕組みのなかで育まれた工学知が、市場経済側のテクノアントレプレナーシップを刺激し、生活者への価値の提供を実現しつつある。

## 業績とその創造性

### 1. 情報化社会におけるコンピュータの重要性とオペレーティングシステムの役割

コンピュータと通信に象徴される情報技術の急速な発達、ワールドワイドな広がり、広範囲なデータへのアクセスを可能にし、生活者に様々な価値をもたらした。情報技術においては、ハードウェア技術とともにソフトウェア技術が重要であり、利用者にとっては応用ソフトやコンテンツが身近な存在である。また、通常のコンピュータだけでなく、生活者を取り巻く様々な機器がインテリジェント化され、マイクロプロセッサを組み込んだ機器すなわち組み込みシステムの数は、通常のコンピュータの数より桁違いに大きなものになっている。

情報は世界中の生活者によってわけへだてなくアクセスされることが重要であり、情報を提供するためのハードウェアやソフトウェアもまたそれを欲する人々にわけへだてなく提供されることが重要となる。情報化社会においては、「オープンであること」、すなわち自由にアクセスでき、自由に利用できることが重要なキーワードになる。この「オープンであること」は、コンピュータのソフトウェア開発、特に基盤的共通的なソフトウェア開発においても、広範な人々の知恵を集めるために重要な要素である。

通常、コンピュータのソフトウェアにおいて、応用ソフトウェアやコンテンツの開発を容易にするために、共通する基本的な機能セットであるOSが用いられる。最近では、メモリ容量の制約が厳しい小規模な組み込みシステムにおいても、開発コスト削減と信頼性の向上のためにOSを用いたシステム構成が採用されるようになってきた。

OSは、多くの応用ソフトウェアから共通して利用される基本的な機能を提供し、コンピュータ全体を管理するソフトウェアである。また、エディタやコンパイラなど応用ソフトウェア開発に必要なツールとなるソフトウェアもOSユーティリティとして重要な基本ソフトウェアである。応用ソフトウェアの開発者は、OSの提供する機能を利用することによって、開発の効率化が可能となり、応用ソフトウェアの操作性を統一することができる。また、ソフトウェアはOSが共通であれば他のコンピュータでも動作することになる。すなわち、OSは、情報化社会の重要なインフラストラクチャの役割を果たすコンピュータやインテリジェント機器におけるソフトウェアの要となる、基盤的共通的なものである。OSとしては、例えば、Windows、Mac OS、UNIXなどがあり、応用ソフトウェアとしては、ワープロソフト、表計算ソフトなどがある。

OSは作成者の能力と個性を反映し、実行速度や異常発生時の対応などにおいて優劣が生じる。また、OSはハードウェアおよびソフトウェアの発展に従って、大規模化、高機能化してきており、

OS自身の開発においてバグや非効率なプログラムが作り込まれる確率が高くなり、その対策がきわめて重要になってきている。また、OSの仕様はハードウェアの機能と応用分野に応じて最適化が必要とされる。

このように、ソフトウェアの規模が大きくなり、開発に多大なコスト(人員、時間)がかかるようになると、ソフトウェアはその開発コストを負担した企業の財産であるという考え方が支配的となり、利用者がソフトウェアを改変することを禁じ、ソースコードは非公開とされるようになった。しかし、大規模ソフトウェアにはバグや障害が避けられず、特にOSや基本ソフトウェアの場合には深刻な影響を及ぼす。従って、自由にソースコードが入手でき、改良改変できるようにしたいという要求が強くなってきた。

また、家電機器や携帯電話などの小規模組込みシステムへの応用においては、通常のOSは規模が大きくて使いにくい。また、外部からの入力信号に対して時間的な制約の中でも優先的な処理ができるようなリアルタイム性が特に要求される。そこで、小規模組込みシステム用のOSが様々なベンダから提供されている。しかし、ベンダにより仕様がバラバラであるために、プログラム担当者が混乱を起こしやすく、設計資産である応用ソフトウェアの流通ができず、従って、ソフトウェア生産性が低いという問題があった。一方、OSを統一すると、システム全体で多種類のプロセッサを使う場合には、個々のプロセッサの特性を活かしきれないという問題も出てくる。

本受賞対象は、これらの問題点を解決するものであり、コンピュータ及び組込みシステムのOS開発を対象としたオープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践である。

上記の開発モデルに基づいて進められた3つのプロジェクトについて要点を述べる。また、表1および図1は、これらのプロジェクトが対象とする領域やその特徴を比較して示したものであり、図2はプロジェクトにおける経過の概略を示している。

## 2. TRONプロジェクト

### 2.1. 概要

坂村健は、1980年 社 日本電子工業振興協会( JEIDA )の委員会においてTRON( The Real-time Operating system Nucleus )の基本構想を提案した。4年間に及ぶ委員会での検討の結果、1984年にTRONプロジェクトが発足した。その中心となるのはITRON( Industrial TRON )サブプロジェクトであり、それまでになかったOS、すなわち組込みシステムに適し、リアルタイム性に優れたOSのインタフェース仕様を標準として作り上げ、それをオープンな仕様すなわち「オープンアーキテクチャ」として誰でも使うことができるようにすることであった。

ITRONプロジェクトでは、仕様を作り、規模や種類の異なるプロセッサに実装し、その結果を見て改良して新しい仕様を作るというサイクルが繰返された。仕様およびその実装の結果をオープンにすることによって、多くの人々の英知を集めることができ、多くの人々が納得する標準への道を歩むことになった。

標準化においては統一性と適応性とのバランスを重視した。すなわち8ビットから32ビットまでの

## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

幅広いレンジのプロセッサに対応するための対策として、下位のプロセッサにおける性能を制限するような仕様は作らないという「弱い標準化」にとどめることが坂村の指導のもとに注意深く行われた。

1988年には(社)TRON協会が発足し、TRON仕様の普及が促進された。決定されたITRON仕様<sup>1)2)</sup>に基づく組込み用OSをマイクロプロセッサメーカーはじめ各社が実装することとなり、ITRONが各社のマイクロプロセッサで使えるようになった。

その結果、応用ソフトウェアの標準化を図ることができ、家電機器、携帯電話をはじめ自動車やファックス、デジタルカメラなどの広範な組込みシステムに適用されることとなった。

国外では、米国に連絡事務所が開設されており、韓国においても2000年8月に韓国TRON協会が発足し、TRONの普及活動が国際的な広がりを見せている。

## 2.2. TRONプロジェクトの経過

最初のITRON1仕様は1987年に公開された。また、この仕様の使用実績や反応をもとに、1989年には、より小規模なマイクロプロセッサ用に機能を絞り込んだμITRON2.0仕様およびより大規模なマイクロプロセッサ用のITRON2仕様を公開した。

特に1993年に公開されたμITRON3.0によって、8ビットから32ビットまでのマイクロプロセッサが統一的にサポートされるようになり、アプリケーションの標準化がいっそう進んだ。また、μITRON3.0では、複数のマイクロプロセッサを相互接続した分散システムをサポートする機能を追加し、ネットワーク化の進展に対応できるようにした。

ITRONの他に、ITRONの成果を利用する形で、BTRON(パソコンやワークステーション用のOS仕様とその関連仕様)、CTRON(通信制御や情報処理を目的としたOSインタフェース仕様)、TRON電子機器ヒューマンインタフェース(各種の電子機器のヒューマンインタフェースの標準ガイドライン)の各サブプロジェクトが進行している<sup>3)</sup>。

また、1997年に公開されたJTRONによって、移植性とネットワーク透過性に優れたJavaの実行環境とITRONの統合が可能となり、両者の特徴を生かすことが可能になった。

1999年には、μITRON4.0仕様を公開し、スタンダードプロファイルを定義することにより、互換性を高めた。

## 2.3. TRON仕様の実績

世界中で使われているほとんどのマイクロプロセッサに対応してITRON仕様の組込み用OSが提供されており、TRON協会の日本のユーザへのアンケート結果によれば、組込みソフトウェアでは一番大きなシェアを持っている<sup>4)</sup>。日本国内の組込み用ソフトウェアの30%から40%はITRONを使っており<sup>5)</sup>、なかでも家電機器組込みシステム用OSでは80%がTRONを使用するに至っている(1999年)。また、最近ではNTTドコモの携帯電話や、トヨタ製自動車の制御など広く組込みシステムに適用されている。



### 3. GNUプロジェクト

#### 3.1. 概要

Richard M. Stallmanは1984年にマサチューセッツ工科大学を退職し、UNIX互換のコンピュータOSをフリーソフトウェアとして作り始めた。これがフリーソフトウェアを作成配布するGNUプロジェクトのスタートである。1985年には、完成したエディタをインターネット上に公開して自由に使えるようにすると共に、フリーソフトウェアを作成配布するGNUプロジェクトの推進組織としてFSF( Free Software Foundation )を設立した<sup>6)</sup>。

GNUプロジェクトは、汎用的ソフトウェアを対象としたオープンな開発方式の先駆である。Stallmanの提唱するフリーソフトウェアの考え方はソースコードの公開、プログラム改良やコピー配布の自由などを含むものであり、これらの自由と公開の原則によって、世界的に衆知をあつめ、高品質で信頼性の高い基本ソフトウェアを実現することが可能になった。この原則をまとめたものがGPL( General Public Licensing )<sup>7)</sup>である。Stallmanはこのフリーソフトウェアの考え方のもとに、UNIX完全互換のソフトウェアシステムの開発を目指した。その結果、高機能エディタEmacs、コンパイラGCC、グラフィカルユーザインタフェース( GNU desktop )GNOMEなどのGNUソフトウェアが作成された。また、OSカーネルも作成されているが、それが実用的に使える段階になる前に、Linuxが開発された。LinuxとGNUソフトウェアとの組み合わせにより、完全なUNIX互換のオペレーティングシステムがフリーソフトウェアとして誰もが自由に使えるようになった<sup>8)</sup>。

GPLというフリーソフトウェアの考え方は、他のオープンソフトウェア開発への基盤の役割を果たすものであり、高機能エディタEmacsなどの成果は、その高機能と高信頼性によってフリーソフトウェアに対する高い評価をもたらした。

#### 3.2. GNUプロジェクトの成果( GNUソフトウェアとGNUGPL )

GNUプロジェクトによって開発されたソフトウェアとして、800余りのGNUソフトウェアがあり、また14以上のGNUプロジェクトが登録されており、進行中である。

GNUが主張するフリーソフトウェアのフリーというのは、価格が無償であることを意味するのではなく、次のような自由を確保することを意味しており<sup>9)</sup>、ソースコードが入手できることはその前提条件である。

- ・ 目的を問わずプログラムを実行する自由( 第0の自由 )
- ・ 必要に応じてそのプログラムに修正を加えたり取り入れたりする自由( 第1の自由 )
- ・ コピーを配布する自由( 第2の自由 )
- ・ プログラムを改良しそれを発表する自由( 第3の自由 )

再配布者がそのソフトを変更していたとしても、再配布するときにはいかなる制約条件も追加せずに、フリーソフトウェアとして再配布することを条件とすることを「コピーレフト( Stallmanの造語 )と呼んでいるが、上記の条件はそのための具体的な条件セットとなっている<sup>10)</sup>。

## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

## 3.3. フリーソフトウェアの信頼性

1990年と1995年のUnixユーティリティプログラムについての調査によれば、商用のUnixシステムの故障率が15%から40%であるのに対して、GNUの故障率は6%ないし9%に過ぎない<sup>11)</sup>。フリーソフトウェアの信頼性が高いことは、2000年9月の米大統領情報技術アドバイザーコミッティの報告書でも認められているが<sup>12)</sup>、その理由は次の2つと考えられている。

- 1)フリーソフトウェアの場合は、問題を解決するためにコミュニティの全員が協力しあう。すなわち、ユーザはバグを報告するだけでなく、自分で修正プログラムを提案したり、バグを修正するためにメールで会話しながら一緒に作業するので、効率的に修正される。
- 2)ソースコードが万人に見えるので、開発者が信頼性を高める努力をする程度が高く、ソースコードのデバッグが徹底的に行われる<sup>13)</sup>。

## 4. Linuxプロジェクト

## 4.1. 概要

Linus Torvaldsは、1991年にLinuxの最初のバージョンを独自に作成し、インターネット上に公開した<sup>14)</sup>。Linuxは、UNIX互換性を備えたコンピュータOSの中核部分(カーネル)である。その後、デバッグおよび改良が世界的な広がりによるボランティアで実行された<sup>15)</sup>。また、得られた改良版のOSカーネルを誰でも自由にかつ無料で利用することができるようにした。

このLinux開発における特徴は、公開されたソースコードの改良・デバッグに多数のボランティアがいつでも自由に直接参加する開発スタイルの創案と実践である。改良案の採用やバージョンアップの最終判断はTorvalds自身が行うことによって、変種を生みださず、統一した形でOSカーネルの改良を進めることができた<sup>16)、17)</sup>。また、ボランティアの行ったデバッグ作業やプログラム改良結果がすぐに公開されるよう、頻繁にバージョンアップを繰返して、多くのボランティアの意欲的な努力を効果的にまとめあげた。

このように、オープンな形でのコンピュータソフトウェア開発方式を採用することにより、多くの自主的で熱意ある作業がスピーディーに行われ、高性能で高い信頼性を有するOSカーネルが生みだされた。

## 4.2. Linuxの開発経過

Linuxは1991年9月に最初のバージョンである0.01がリリースされた。その後、1994年にリリースされたバージョン1.00によってカーネルが完成し、1996年のバージョン2.0で、実用的なOSとなった<sup>18)</sup>。

Torvaldsが、Linuxの開発の中心的な役割をはたしている。全世界にいるボランティアの開発者が、みずからの意思でみずからの必要性にしたがって必要な部分を開発し、Torvaldsが彼の支援者の協力により、ボランティアが開発したコードを採用するかどうか決定している。その機能の必要性を感じたボランティアが、自分の要求を満たすために開発するので、短い工期で品質の良いものができるという特徴がある。

## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

また、Linuxの仕様そのものは、UNIXの標準仕様POSIXに基づいておりUNIXクローンであるが、ソースコードは独自のものであり、誰でも自由にソースコードを手に入れることができる。

Linuxの各バージョンのリリース経過は次の通りである<sup>19)20)</sup>。

1991年 Torvaldsが開発を開始、バージョン0.01カーネルをリリース

1994年 1.00カーネルをリリース

1996年 2.0(安定版)、2.1.0(開発版)をリリース

1999年 2.2.0(安定版)、2.3.0(開発版)をリリース

2001年 2.4.0(安定版)をリリース

バージョン0.01から1.00までは、2年5ヶ月間に119回のバージョンアップが行われ、1.00から2.0までは、2年3ヶ月間に234回行われた。また、2.0から2.1.0までは4ヶ月間に21回、2.1.0から2.2.0までは2年4ヶ月間に142回、2.2.0から2.3.0までは4ヶ月間に9回、2.3.0から2.4.0までは1年8ヶ月間に74回のバージョンアップが行われている。頻繁にバージョンアップされた時期(特に1994年から1996年)においては1ヶ月間に10回以上の改訂が行われている。

#### 4.3. Linuxのディストリビューションの特徴

Linuxは、OSのソースコードに加えて、インストールを簡易化するソフトウェア、ネットワークサポート、ユーティリティプログラムなどのソフトを組合わせて、ユーザが使いやすい形にパッケージ化して配布することをディストリビューションと呼ぶ。ボランティアベースの無料ディストリビューションもあるし、ビジネスとして料金を取って販売している会社もある。このディストリビューションという仕組みは、Linuxのソースコードの統一性を損なわずに、したがって互換性を損なわずに、Linuxを普及させる仕組みである。

LinuxのもととなったUNIXの場合は、ソースコードは有償でAT&Tから公開されていた。カリフォルニア大学バークレー校で開発したBSDのコードは、そのライセンスを持っていれば入手できた。また、多くのシェルやユーティリティが大学などで開発され、そのソースコードは公開されていた。それらのソースコードをそれぞれの状況に応じて書換えて、それぞれのベンダが製品化を行った。

UNIXにおける標準化の努力は、ソースコードそのものを統一するのではなく、POSIXのような標準仕様を決めて行おうとした。カーネルのソースコードはそれぞれのベンダにまかされた。したがってそれぞれのベンダによってソースコードが異なり、細かな仕様も異なるようになってしまった。そのため、同じUNIX系のOSでも互換性は保証されなかった。また、ベンダのソースコードは公開されていないため、アプリケーションプログラムを動かしてトラブルが発生しても、OSカーネルの細かな仕様の違いによるものなのか、アプリケーションの障害なのか、あるいは、OS自身の障害に起因するものなのかを特定することは、OSベンダの協力がなければ不可能であった。



## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

これに比較して、Linuxでは、どのディストリビューションでも統一されたLinuxカーネルを使っているため、互換性は保たれている。ソースコードが公開されているので、サポートもやり易いという利点がある。また、Linuxのライセンス料は無償である。特に、多数の端末を使う場合や、組み込み用のOSとする場合はOSのライセンス料は台数分だけ必要となり、これが無償であることは大きな利点となる。

## 4.4. LinuxおよびGNUソフトウェアの利用状況

Linuxは、当初スタンドアローンのデスクトップコンピュータ用OSだったがマルチプロセッサもサポートされ、スーパーコンピュータ用OS、大規模システム用OS、ネット端末用OS、携帯端末用OS、組み込み用OSとしての用途にも広がっている。必要性を感じた人が必要とする部分の拡張を行い、ソースコードが公開されるので、他の人も利用できることになる。

Linuxのシェアは、1999年の全世界サーバOSの出荷本数シェアにおいては25%(130万本)、2000年における日本のサーバOSの出荷本数シェアにおいては7.8%を占めており、安定性のある有力なサーバ用OSとなった。OSそのものは無償で出荷されているため、金額に換算することは難しいが、非常に大きな市場を作り出したことは確かである。

GNU及びLinuxは、その性能及び信頼性の高さシステム開発に必要なコストの低さの故に、サーバを始め一般的な用途のコンピュータシステムで広く使われ始め、最近では、主要なコンピュータメーカーがすべてのコンピュータでLinuxを搭載可能とした。さらに、2001年5月には、IBM、富士通、日立、NECが、銀行の勘定系など企業の基幹業務システム用OSについてLinuxをベースとして共同開発することで合意した。Linuxの特徴である低価格、システム安定性、高いネットワーク機能、システム互換性に加えて、4社が協力することで開発コストの削減と開発期間の短縮が可能になるとしている。

## 5. 波及効果

TRONプロジェクトの基盤となっているITRONは、家電機器や携帯電話、自動車等の組み込みシステムに適用されており、日本国内の組み込みシステム用OSの30%から40%でITRONが使われている。なかでも家電機器組み込みシステム用OSでは80%がTRONを使用するに至っている(1999年)。

米国にはTRON協会の北米連絡事務所が設けられており、ITRONをベースにしたOSを扱うソフト開発・販売会社が10社ほどある。また、韓国では2000年8月に韓国TRON協会が設立され、韓国語によるホームページを開設して現地企業からの問い合わせに対応するなど、TRON普及活動を開始している。

GNUとLinuxは組合わされ、広範なシステムにおいて使用されており、特に高信頼度を必要とするサーバOSとして25%の出荷本数シェアを占めている。また、最近では、主要コンピュータメーカーがLinuxを採用する計画を発表している。さらに、高信頼性を要求される米国陸軍の

## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

車両モニタリングシステムにもLinuxが採用された(2000年12月)

オープンソフトウェア方式の波及効果として、これ以後の様々なソフトウェア開発に開発方式として採用された。例えばWebサーバソフトウェアの開発においてこの方式が採用され、LinuxなどのUNIX系OSやWindows上で動作するWebサーバソフトウェアApacheが生まれている。Apacheは、1996年にはWebサーバの30%程度を占め、2000年にはさらに増加して60%を占めており、現在最も広く使われているWebサーバソフトウェアである。

## 6. 結論

以上述べたように、坂村、Stallman、Torvaldsがそれぞれの中心的役割を果たしたTRON、GNUおよびLinuxの各プロジェクトは、情報化社会の要となるコンピュータ基本ソフトウェアの開発において、オープンな開発スタイルあるいはオープンな利用の仕組みという新しい手法による解決方法を生み出した。これは探究心に富んだテクノアントレプレナーシップの発揮によるものである。その結果、広範な人々の知恵を集めることが可能となり、これまでにない高いレベルの成果をもたらしてコンピュータ利用の新たな発展をもたらした。さらに、この新しいオープンな手法は、新しい開発モデルを提示してコンピュータソフトウェア業界におけるそれまでの市場の常識を覆すような影響を与えている。すなわち、通常の市場経済とは異なる仕組みのなかで育まれた工学知が、市場経済側のテクノアントレプレナーシップを刺激し、生活者への価値の提供を実現しつつある。

## 7. 受賞者の略歴

## 坂村 健

1979年 工学博士

1984年 TRONプロジェクト設立

現在、東京大学大学院情報学環 教授

## Stallman, Richard M.

1971年 マサチューセッツ工科大学AI研究所

1984年 マサチューセッツ工科大学を退職し、GNUプロジェクトを始める。

1985年 Free Software Foundation (FSF) を設立

現在、FSFのPresident

## Torvalds, Linus

1969年 ヘルシンキ大学入学

1991年 ヘルシンキ大学在学中にコンピュータのOS、Linuxを開発する。

現在、トランスメタ社勤務



## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

## 参考文献

- 1) K. Sakamura. The TRON Project. IEEE Micro, Vol. 7, No. 2 ( Apr. 1987 ) pp. 8-14
- 2) K. Sakamura. The TRON Project. Microprocessors and Microsystems, Vol. 13, No. 8 ( Oct. 1989 ) pp. 493-502.
- 3) [http://www.tron.org/tronproject/tp\\_view.html](http://www.tron.org/tronproject/tp_view.html)
- 4) <http://www.itron.gr.jp/survey00/result-j.html>
- 5) トロン協会 ITRON 専門委員会ホームページ, Introduction to the ITRON Project,  
<http://www.itron.gr.jp/panph98/panph98-j.html>
- 6) <http://www.gnu.org/gnu/thegnuproject.html>
- 7) <http://www.gnu.org/japan/manifesto-1993j-plain.html>
- 8) [http://www.Linux-mag.com/1999-07/stallman\\_01.html](http://www.Linux-mag.com/1999-07/stallman_01.html)
- 9) <http://www.gnu.org/philosophy/free-sw.ja.html>
- 10) <http://www.gnu.org/philosophy/categories.ja.html>
- 11) [ftp://grilled.cs.wisc.edu/technical\\_papers/fuzz-revisited.pdf](ftp://grilled.cs.wisc.edu/technical_papers/fuzz-revisited.pdf) "Fuzz Revisited: A Reexamination of the Reliability of Unix Utilities and Services" by Barton P. Miller et al.
- 12) [http://www.itrd.gov/ac/letters/pitac\\_itr\\_sep11.html](http://www.itrd.gov/ac/letters/pitac_itr_sep11.html) "Report to the President on Open Source Software for High End Computing", September 11, 2000
- 13) <http://www.gnu.org/software/reliability.ja.html>
- 14) What is Linux?, Linux Journal Home Page, June 24, 1999.  
[http://www2.linuxjournal.com/about\\_linux/index.html](http://www2.linuxjournal.com/about_linux/index.html)
- 15) 大谷卓史, Linuxはこうして生まれた, 「まるごとわかる最新Linux読本」技術評論社, 1999年5月, pp102-105.
- 16) リーナストーパルス, デビッドダイヤモンド, 「それがぼくには楽しかったから」小学館プロダクション, 2001年5月.
- 17) Wonderful World of Linux 2.4, Jul 17, 2000, Linuxtoday.com
- 18) Linuxとは, [www.Linux.or.jp/general/Linux.html](http://www.Linux.or.jp/general/Linux.html)
- 19) Linux Kernel Version History, [www.memalpha.cx/Linux/Kernel/](http://www.memalpha.cx/Linux/Kernel/)
- 20) 岸哲夫, Linuxカーネルの変遷(年表), インタフェース2001年6月号p.56.  
[http://www.cqpub.co.jp/interface/toku/200106/if06\\_toku1-h1.htm](http://www.cqpub.co.jp/interface/toku/200106/if06_toku1-h1.htm)

## 授賞業績

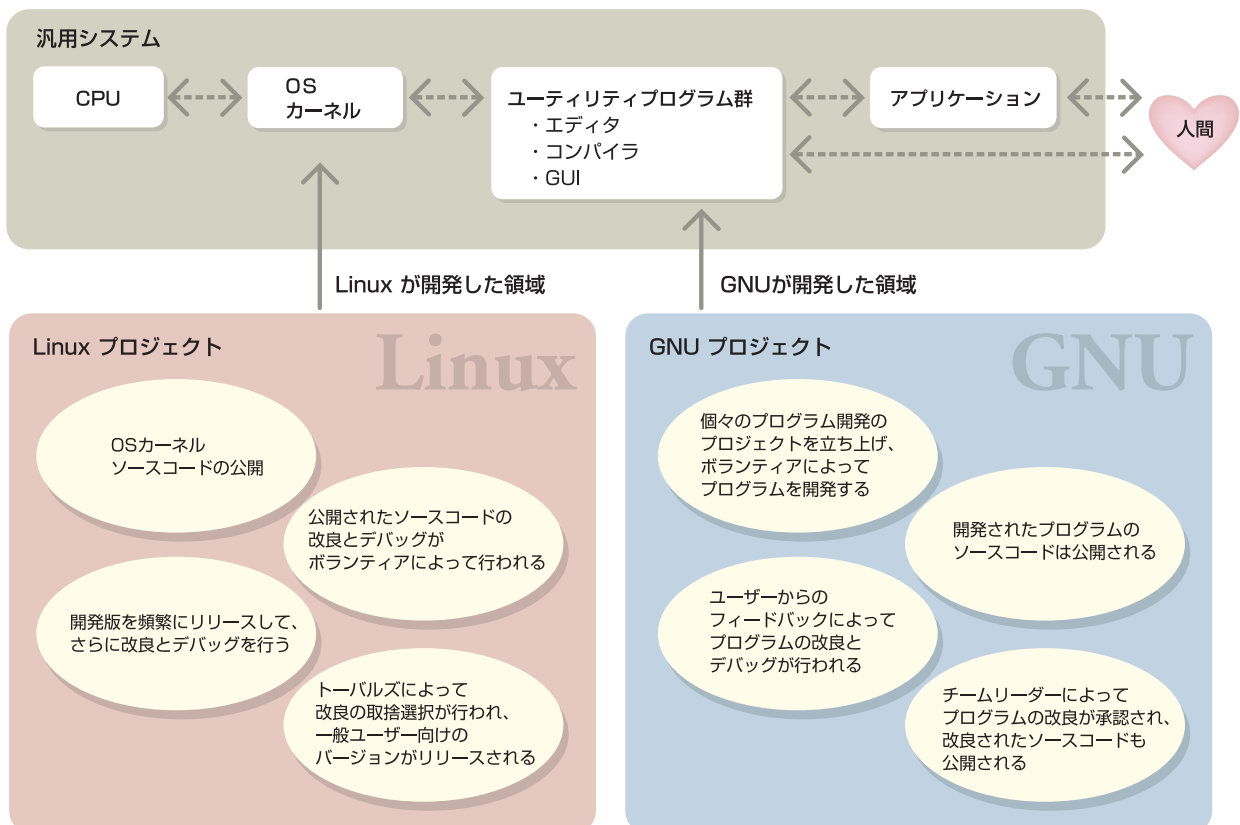
## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

図1 各プロジェクトの比較

## ● 組み込みシステムにおけるTRONプロジェクト



## ● 汎用システムにおけるLinuxプロジェクトとGNUプロジェクト



## 授賞業績

## オープンなコンピュータ基本ソフトウェア開発モデルの提唱と実践

表1 開発モデルの特徴

授賞業績	<h2 style="text-align: center;">TRON</h2> <p style="text-align: center;">TRON プロジェクト</p> <p>API(Application Programming Interface)仕様を公開して、組み込みシステムのOSおよびアプリケーション開発の促進をはかる開発モデル</p>	<h2 style="text-align: center;">GNU</h2> <p style="text-align: center;">GNU プロジェクト</p> <p>汎用システムのユーティリティプログラム群の開発を、フリーソフトウェアの考え方で行う開発モデルとGPL (General Public Licensing)の定式化</p>	<h2 style="text-align: center;">Linux</h2> <p style="text-align: center;">Linux プロジェクト</p> <p>ソースコードをオープンにして行う汎用システムのOSカーネル部分の開発モデル</p>
開発モデルの特徴	<p>オープンアーキテクチャーの考えのもとに、API仕様を公開する。それに基づき、OSカーネルやアプリケーションを自由に開発できる。また、開発者はAPI仕様の改良を自由に提案できる。APIの仕様の決定はTRONプロジェクトで行う。</p>	<p>フリーソフトウェアの考えのもとに、UNIX互換システムを構築するプロジェクト。プログラムを開発するチームがつくられ、開発されたソースコードは公開される。他の人がそれを改良したら、同じくソースコードを公開する。改良の取捨選択はチームリーダーが行う。開発されたプログラムは誰が使ってもよい。</p>	<p>ソースコード公開の考えのもとに、OSカーネルのソースコードを公開して、その改良とデバッグをボランティアで行う。改良の取捨選択はトールバルズが決定する。この方法により短期間でOSのバージョンアップが進められる。開発されたOS(Linux)は誰もが使える。</p>

図2 TRON、GNU、Linuxの開発経過

