

機能過多製品の開発における開発者の心理的特性：
個人内における態度の乖離について

メタデータ	言語: jpn 出版者: 公開日: 2016-06-06 キーワード (Ja): キーワード (En): 作成者: 高橋, 晃 メールアドレス: 所属:
URL	https://doi.org/10.14945/00009437

機能過多製品の開発における開発者の心理的特性 — 個人内における態度の乖離について —

Engineer's psychological characteristics in the development of the excessive
function product

— Dissociation of the two attitudes in an individual —

高橋 晃

Akira TAKAHASHI

静岡大学 学術院 情報学領域

akirtaka@inf.shizuoka.ac.jp

論文概要：個人の中における、開発者としての態度とユーザとしての態度の乖離を、将来システム開発・運用にかかわる可能性が高い学生 347 名を対象にした調査に基づいて検討した。インターフェイス関連の講義において、ユーザインターフェイスが類似しており、構造が相対的に単純および複雑なゲームを 2 種類体験させ、“販売するゲームとしてはどちらが良いか”と“自分の好みのゲームはどちらか”について質問をした。その結果、個人の内部において、“ユーザとして”の好みは“シンプルなソフトウェア”を選択したにもかかわらず、ソフトウェアを販売するという“開発側として”の前提を与えた場合には、多機能で複雑なソフトウェアを売らなければならない、と同時に考えている回答者が、全体の約半数を占めることがわかった。ソフトウェアや携帯機器の機能の肥大化の要因の一端が、こうした心理的特性によるものである可能性が示唆され、その防止策が検討された。

キーワード：機能過多製品、開発者、ユーザビリティ、心理的特性

Keyword: excessive function product, engineer, usability, psychological characteristics

序：多機能機器の問題点

現在の携帯端末機器は複雑な道具であり、膨大な数の機能が含まれている。その昔、電話機は“受話器”と“ダイヤル”からなる単純な利用構造をもった道具であった。ユーザのすべきことは、電話を受ける際には受話器を取って相手と話すこと、電話をかける際には受話器を取ってから適切にダイヤルを回すことのみであった。しかし、1980 年代から機器にコンピュータチップが内蔵されて以来、電話機には様々な機能が追加されていった。現在におけるその発展形としての“携帯端末機器”には、さらに膨大な数の機能が内包されている。小さな携帯端末に含まれる機能数であっても、様々な水準のものを含めて数百から数千の機能が実装

されている。また、近年、携帯型コンピュータに通話機能を追加した機器である“スマートフォン”が開発されると、それらのオリジナル機能に加えて、ソフトウェアとして「アプリ」という形での機能をユーザが後から追加することが可能になった。アプリについては機能数に制限はなく、機器の性能の限界まで機能を追加することが理論上は可能である。このように、現在の電子機器は複雑な様相を呈している。

だが、そうした複雑な機器を「使いこなせる」と感じているユーザは必ずしも多くない。むしろ、多くの機能に混乱し、機器の利用に困難を感じる例が報告されている [1]。そのため、結果的に、ユーザの一部が、多機能すぎるスマートフォンから、通話機能を中心としたシンプル

な「携帯電話」に回帰する傾向は、この 2010 年代においても認められている [2]。また、一部の企業がシンプルな機能のみの端末をあらためて開発したところ、料金の安さもあいまって、高い人気が出ている [3]。その一方で、開発者側においては、従来からの“通話機能を重視した携帯電話”を商品展開上では不要とみなし、開発を中止するという方向性が主張されている [4]。これは、現状で全契約の 47.7% を占める旧来の携帯電話ユーザを、スマートフォンに移行させたいという“開発者側”の強い圧力を意味している。

問題は、機能の多さゆえに多くのユーザが使いこなせないと感じる製品を、開発者がなぜ開発し続けるのか、というところにある。本来は、開発者も“ユーザ”の一人である。開発者自身が使いにくいと感じることがあれば、姿勢としてそうした製品を開発しないものと想定されるが、しかし、現実の携帯機器を観察すると、実際にはそのような“振る舞い”にはなっていないことがうかがえる。

一般的に、機能の多さは“市場ないしユーザの要請”によるという説明がなされる [5]。ユーザ自体が「あれも、これも」という多機能化を望んでいるため、開発側がそれに対応せざるを得ない、というとらえ方である。機器販売の現場においては、ユーザは使い勝手について尋ねることよりも、その機器を利用して何ができるのかが興味の対象となる。通例、さまざまなことが（可能性として）“できる”、という状況は、何かが“できない”ことよりも望ましい状況とみなされる。そこで、電子機器においても、販売の現場において“〇〇ができます”という説明を数多く可能にするために、より多くの機能が必要とされる、という観点である。これらは、外的な要因である。

一方、開発者の内部の要因としては、開発者は、自分自身の製品が詳細にわかってしまっているため、その“使いにくさ”を把握できないこと、そこから“使い方がわからないというこ

と自体がわからない”という心理的状况がある [6]。開発者はそのシステムを開発する過程において、その内部構造も含めた機器全体の様相を深く学習し、把握している。そのため、たとえばある操作が、機器にはじめて触れるユーザにとって不自然であるとしても、開発者自身は“知りすぎている”ために、その不自然さを直感できない可能性がある。こうした形での過剰な理解や慣れによる弊害が、システムの複雑さや難しさを、開発者自身に見えにくくしていることも考えられる。

しかし、それ以外に、機能の多い複雑な機器を開発する際の、開発者側の内的な要因はないのだろうか。たとえば、開発者自身が機能の多さや複雑さを好んでいる可能性である。この場合には、複雑な機器を作成するのは開発者自身の心的方向性がそちらの方向に向いているためであると解釈できる。本研究は、事前の予測として、こうした開発者の“複雑な機器を好む心理的傾向”が認められることを仮定した。

この仮説をもとに、古典的 RPG ゲームの体験を題材とし、ソフトウェアの機能的複雑さと開発者側の心理的要因の関連を調査した。

方法

参加者 静岡大学情報学部 情報システムプログラム受講生¹ 計 347 名

調査期間 2012-2015 年 各年 10 月第 1 週から第 3 週

手続き 参加者は静岡大学情報学部 情報システムプログラムにおいてインターフェイスを扱う講義「ユーザビリティ設計・評価論」の講義内課題の一環として、以下の (1)、(2)、(3) の 3 つの課題を 3 週にわたって行った。

(1) 第 1 週 第 1 課題：1980 年に開発された探索型のゲームである「Rogue」(付録 1 参照) を各自のノート PC 上で 20 回以上プレイし、点数や自分なりの戦術などを用紙に記述し、

¹ 情報システムプログラムは、大学卒業後、情報システムの開発・管理・運用などに携わる可能性が高い学生が所属するコースである。本研究では、この学生たちを将来の開発者の卵とみなし、その心理的特性を探ることを目的としている。

提出した。

(2) 第2週 第2課題：上記 **Rogue** に要素や機能を追加し、発展させた「**Nethack (jNethack)**」テキスト UI 版（付録2 参照）を10回程度プレイし、**Rogue** との差異を考察し、提出した。

(3) 第3週 第3課題：三週目の講義において、全参加者は以下の質問に理由をつけて回答し、提出した。

問1 もし販売するとしたら **Rogue** と **Nethack** のどちらを選択するか。

問2 あなたは **Rogue** と **Nethack** のどちらが好きか

全参加者は講義課題として上記の質問に回答した²。

結果と考察

表1に、問1「販売」と問2「好み」についての2つの質問に対する回答を各軸とした回答カテゴリの人数分布を示す。結果として、個人的な好みについては、全体の85.9%がシンプルなソフトウェアを好んでいるが、その半面、販売については60.2%が「複雑なソフトウェアを販売すべき」と考えていることがわかった。さらに「自分はシンプルなもの (**Rogue**) が好きだが、

売るのであれば機能が多いもの (**Nethack**) を売る」という、自分の好みと販売が乖離したカテゴリの回答者が全体の50.7%を占めた (**Rogue** と **Nethack** の逆方向も含めれば55.3%)。

理由を説明した自由記述欄における、個人内での態度の「乖離例」を表2に示す(表中下線部は、同一人物内部における矛盾部分)。ここでは、同じ人物の内部において「何が売れるのか」という想定と、自分自身の「好み」の間に、ギャップがあることが示されている。

表2の感想から、開発者の態度として、特に2つの特徴が認められた。一つは、ユーザの学習能力を考慮しないこと、ないしはその過大視である。「ゲームをやる人、ないしお金を出して購入する人であれば、このくらいは学ぶことができるだろう」という推測が多く見られた(例 受講生 A、B、C、D)。実際には、自分自身が講義課題としてある程度時間をかけていても、**Nethack** の習得はできていない。それにもかかわらず、ユーザに対して一方的にそのような推測をしていることは、ユーザの性質に対して誤った前提をおいていることを示している。理由の中には「慣れてきたら」という条件付きの文言も見受けられたが、そもそも学習が進行しない限り「慣れる」という状態には達しない。実際には、ユーザの学習資源(学習時間、意欲等)は有限であり、金銭を支払ったからといって、無条件に学習がなされるわけではない。想定される利得と比較して {支払いコスト+学習

²本調査では、常に先に **Rogue** をプレイさせ、次に **Nethack** をプレイさせたため、順序性の問題(先に **Rogue** をプレイしたことによる **Rogue** に対する好意的影響や、**Nethack** に対する学習干渉)が包含されている可能性が高い。しかし、ゲーム初心者に対して、高機能で複雑な **Nethack** を最初から一定の回数以上プレイさせることは、それ自体が大変困難であるため、講義課題として成立しにくい。そのため、学習の順番として「単純から複雑へ」という一般的な順序性を保った状態で課題を出している。

表1 個人的好みと販売の選択
「どちらを売りたいと思うか」

		Nethack	Rogue	計
「どちらが好きか」	Nethack	33 (9.5%)	16 (4.6%)	49 (14.1%)
	Rogue	176 (50.7%)	122 (35.2%)	298 (85.9%)
	計	209 (60.2%)	138 (39.8%)	347 (100%)

表2 乖離のある感想の例

受講生	販売するのであればどちらが売れると思うか	自分自身はどちらが好みか
ユーザの学習能力の過大視		
受講生A (2012年度2年生)	Nethackの方が多く世間的に売れるゲームになるのではないかと思います。このゲームを買うということは、それなりにこのゲームをクリアしようという気力があると思います。このゲームをお金を出して買うのなら、割と多いコマンド操作も覚えようとするのではないかと思います。	私は、どちらかというとRogueの方が好みます。Nethackは自由度や、ストーリー性が向上したのはいいのですが、なんにせよマニュアルが読みにくく、そして長いため、僕にはとっつきにくい物でした。(中略)マニュアルを覚えるというのに時間を費やしては元も子もないと思いました。
受講生B (2014年度2年生)	ゲームを買うことを考えたときにある程度はやりこみ要素がほしいので、Nethackを売り出す。(中略)操作を覚える前に諦めてしまう人がいるかもしれないが、そこで耐えることができれば楽しめると思う。お金を払ったことを考えれば、操作方法を覚えるところくらいまでは頑張ってプレイする気がする。	私はRogueの方が好みだと感じた。普段からあまりゲームをする方ではないのでRogueの操作を覚えるだけでかなり苦労したし、敵との効率的な戦い方もまだあまりちゃんとできていない。Rogueでさえまだまだ上手くプレイできていないと感じるので、Nethackは難し過ぎる。
受講生C (2013年度2年生)	Nethack 理由:おそらくこのようなダンジョン系のゲームを購入する方はゲームをやりこむことが好きな顧客層が多いと予想できる。(中略)また、複雑さも逆に顧客への好奇心をそそるのではないかと考えられる。	Rogue 理由:それほどゲームをやりこんだことがない自分にとっては、短時間でコマンドをたくさん覚えることは苦痛だし、Rogueのほうが単純であるので、私はRogueのほうが好きである。
受講生D (2014年度2年生)	RogueはNethackに比べて単純であるので売るとしたら自由度の高いNethackのほうが売れると思う。よく分からないからこそプレイしがいがあるのではないと思う。ゲームを買ってくれる人はやりこむ人が多いと思うのでRogueよりは売れると思う。	私はRogueのほうが好きだ。Nethackは操作が意味不明でゲームがうまく進んでいくかないのでRogueのほうがプレイしやすい。インターフェイスが単純で理解してしまえば分かりやすいし楽しい。
多機能への傾倒		
受講生E (2013年度2年生)	世間的に売れるゲームになるもの…Nethack (理由)お金を出して買う場合、より機能が多いものの方が売れると思うから。あまり簡単に攻略できて買った甲斐がないと思われるから	自分自身の好み…Rogue (理由)あまり機能が強いと攻略する前に飽きてしまうから
受講生F (2014年度2年生)	売り出した場合はNethackの方がまだ荒削りな部分もあるが、機能はととても多いため売れるのではないかなと思う。	私自身はRogueの方が圧倒的に好きだ。(中略)機能が少ないことで戦略をしっかり立てないと先に進むことができなくなるというゲームの面白さを十二分に味わわせてくれるゲームだと思っている。
受講生G (2012年度2年生)	世間的には恐らくNethackの方が売れるのではないかな。一般人として二つのゲームを見た場合、内容が豊富な方を選ぶ。さらにどちらが最新版かということも意識する。	自分自身は簡単な操作で単純にやり込めるRogueの方を選ぶ。Nethackは懲りすぎていて、ゲームをクリアするために必要な最低条件(Rogueが実装しているもの)を大幅に超えていて、これらは自分にしてみたらゲームを楽しむのに無駄な要素であるとみなす。
受講生H (2014年度2年生)	私は、Nethackの方が売れると思います。(中略)今販売されているゲームは命令やコマンドがたくさんあるゲームばかりです。そのようなゲームでないで現代人は満足しないと思います。	私は、Rogueの方が好きです。Nethackのようにコマンドが多すぎると混乱してしまうし、簡単に操作出来るほうが飽きずにできるからです。
単純さへの不安		
受講生I (2012年度2年生)	売り出した場合を考えるなら、jNethackのほうが売れるのではないかなと思います。理由は、やはり機能が豊富でやり込めるのではないかと考えたからです。商品として出すとしたらRogueは少し単純すぎるのではないかなと思いました。	Nethackはあまりに機能が複雑だと操作を覚えきる前にうんざりしてしまうと思います(実際少しくんざりしました)。Rogueはすぐにゲームを開始できてコマンドの数も少ないので、自分が個人的にプレイするならRogueのほうがいいかなと思いました。
受講生J (2012年度3年生)	きっとNethackではないだろうか。Rogueはシンプルすぎるとか、原始的すぎるゲームだといった声が生まれそう。Rogueは機能的には問題ないが、見た目に華がないように感じられる。	Rogue 理由:余分な装飾や属性などがなく、シンプルでわかりやすいのに奥が深い。やっていてルールや操作方法になれるまでにはRogueの方が楽だった。余分なものが増えたNethackはやりづらかった。
受講生K (2014年度2年生)	多分Nethackの方が売れるものとなると思う。理由としてはrougeレベルでは内容が薄すぎて、ゲームとして売るにはインパクトが無いから。Nethackのほうは(ある意味)ストーリーなどがしっかりしている上にちゃんと解説などしてくれるので売りやすいと思う。	自分はRogueの方が好き。なぜならざりやすい上に分かりやすいから。NethackはRogueの延長線上にあるようなものなので、もしNethackがもう少しとっつきやすかったらNethackのほうを好きになっていたかもしれない。
受講生L (2013年度2年生)	Nethackのほう売れると思う。機能が強いから、販売するとなるとNethackのほうに向いている。Rogueだと物足りないという不満が多そう。	Rogueのほうが好きです。ちょこっとやるゲームならRogueぐらいの機能がちょうどよかった。中途半端に機能増やしたところで、ちょこっとやるゲームであるのは変わらないから。

コスト」の方が高いと予測される場合には、対象の学習を放棄するほうが合理的な解決になりうる可能性もある。この場合には、購入した機器は利用されない、ないしは日常的に用いる単純な機能のみの利用となる。そして、それ以外の利用されない機能は事実上“ノイズ”となり、機器の学習を阻害する要因になりうる。

もう一つは「販売」行為に対する意味合いとして「機能数や自由度の高いものを売らなければならない」という観念である(例 受講生 E、F、G、H)。具体的には、「お金を出す場合には機能がたくさんあるほうが売れる(受講生 E、F)」「内容が豊富なもの、および最新のものを選ぶ(受講生 G)」「現代人は複雑なゲームでなければ満足しない(受講生 H)」といった主張が見られた。しかし、一方でその同じ当人が「自分は機能が多いと攻略する前に飽きる(受講生 E)」「機能が少ないことで戦略の組み立てが重要になり面白くなる(受講生 F)」「Nethack は無駄な要素が多い(受講生 G)」「Nethack は複雑すぎて混乱する(受講生 H)」ために、Rogueの方が好みであると感じている。これは個人の内部における態度の矛盾である。

自分はシンプルなものが好みであるにもかかわらず、他者はそうではないと考える理由は、ここでは主として立場の相違による理由付け(「自分はゲームが得意ではないが、ゲームをお金を出して買うような人はゲームが得意だろう」)が認められる。しかし、この考え方を一般の利用者が多い機器に拡張した場合には、この「立場の相違」という理由は成り立たなくなる(例:「自分は単純な携帯端末が好きだが、携帯端末を購入するような人はみな複雑な携帯端末をいじって機能を使い込むのが好きだろう」この考え方が誤っているのは明白である)。

この後者の思考パターンが、わが国を含めた世界の多くのテクノロジー機器の開発企業が「使えない多機能性」を満載した各種の機器を生む理由の一端を示唆していると考えられる。すなわち開発者の内部において「お金を取って

販売する」という行為が、売る対象の“多機能性”という概念と直接につながっている。ここから、開発者が「価格と機能数の間には正の相関があるべき」という信念を持っていることが伺える。このことはまた、逆に機能の少ないシンプルなシステムを(高い金額では)売れない、という考え方にもつながる(例:表2における受講生 I、J、K、L)。

先にも述べたが、この回答者は、情報学部における“情報システムプログラム”という、将来において情報機器の開発や販売、保守等にかかわる職種に就く可能性が高い学生である。そうした学生の多くが、将来において上記のような「心情」で開発に望むこと、そして、そうした心情のメンバーが開発チームにおいて半数を占めるとしたら(あるいはプロジェクトリーダーがそうした方針を取る心情のリーダーであるとしたら)チーム全体の方針が機能過多の方向に偏ることも自然であると考えられる。これはおそらく、現状の日本企業における情報機器の開発方針とも相関しているだろう。

以下に、この個人内部における態度の矛盾を、外的要因と内的要因の二つに分けて検討する。

外的要因：機能数の増加とユーザビリティの低下

従来より、わが国をはじめとした各国の高機能製品開発は、基本方針として「付加価値を高くして売る」ことでライバル製品との差異を明確化し、その結果として高い価格で販売を行うことを狙ってきた。この“付加価値”の概念が、1980年代から始まる電子制御機器を含んだ商品開発においては“商品の持つ機能数の増加”につながった。

しかし、コンピュータチップによる制御機器においては、この方針が裏目に出ている。コンピュータ内蔵機器においては、単にソフトウェアとしての「機能」を増加させることは、物理的な側面を変化させること(金型等の制作)と比較して容易である。そのため、付加価値を求

める際に、より多くの機能を追加することが“できて”しまう。こうして、多くのユーザの要望に沿う形で次々と機能を追加していくことを繰り返していくと、個々のユーザには必要のない機能を数多く持った「製品」ができあがる。しかし、機能数がある水準（それほど高くない）を超えると「開発者が想定する価値」が「ユーザにとっての価値」と、ずれてくる可能性が高くなることが推測される。また、一般に、集約された電子機器においては、一度追加された機能を同一システム内で「削除」することは、機能を“追加”することよりも困難である[7]。ここから、製品の“改善”が累積されるほど、当該機器の内包する機能は増える傾向にあることが伺える。

本調査の結果から、開発者は、あるシステムの「品質」ないし「付加価値」を（おそらく意図せずに）「機能の多さ」と結び付けていることがわかる。この現象は多機能電子機器の開発における「忍び寄る機能主義 (Creeping Futurism)」として知られている現象のとらえることができる[8]。

ユーザにとっての道具の「価値」とは、それを使って効率よく自分自身の目的を達成できることにある³。多くのユーザに販売する場合には、各ユーザの多様な目的に対して、それに対応するために多様な機能をつけることは、一見その目的にかなっているように見える。しかし、この方針は結果として、あるユーザにとって必要となる機能が、不要な機能（しかしこれは別のユーザにとっては必要な機能である）の中に埋もれてしまうという状況を生じさせている。これはユーザ相互にあてはまり、最終的に「誰にとっても使いにくい」すなわち「価値の低い」製品を産み出す結果となっているものと考えら

れる。ここから、価値を高めようとして、誰にとっても均等に価値の低いものを生み出してしまおうという矛盾が生じている。

内的要因：個人的好みと開発者としての心理傾向の乖離

さらに、本当の問題は、実は開発者自身の心のありように矛盾があり、そのことに気がついていない、あるいは気がついていても自分自身の“開発者としての常識的観点”によってそれを覆い隠している点である。本調査結果からは、事前の予測と異なり、開発者自身も、必ずしも複雑な機器を好んでいるわけではなく、むしろ多くの開発者が「シンプルなものの方が好き」であることがわかった。それにもかかわらず、その意識を自分が行う開発行為自体に反映させることができないことを示唆している。すなわち、開発者の心情において「販売する」「お金を取る」という行為が「多くの機能を提供する」行為（のみ）に自然に結びついていることが問題である。

これはある意味では開発者の“真面目さ”ゆえの弊害であると同時に、開発者に対する「ユーザにとっての真の価値とは何か」という“価値観の（再）教育”が必要とされる点でもある。現在は、無料で配布されているソフトウェアであっても相当数の機能を持っていることが多い。本研究の調査からは、金銭と引き換えに提供するソフトウェアに対しては、無料のソフトウェアよりもさらに多くの機能を提供する必要がある、と開発者が考えている可能性が高いことが示唆されている。しかし、先に述べたように、ユーザにとっての道具の価値は、機能数の大小に依存するものではない。道具を使う行為には必ず目的があり、その目的を効率よく達成することができなければ、道具はその役割を果たさない。ここから、開発者が開発すべき、妥当性の高いシステムとは、その道具やシステムが目指す“ユーザの目的を果たす方向へのベクトル”と、一方で学習の妨害要因となる“ノ

³ 一般に、製品開発において、ユーザエクスペリエンスを重視する企業（ex. Apple）では、製品の価値とは、ユーザがその機器・システムを利用して体験することそれ自体であると想定している。この場合には、機能数という形での自由度はむしろ少なく与えられるが、ユーザはそれゆえに迷うことが相対的に少なく、また自分が使う機能を効率よく探すことができる。

イズとなるベクトル”の比が、大きいものであるといえる。多くの機能が実装されていた場合に、その一つ一つの方向性がばらばらであれば、その道具がどこに“向いている”のかわからなくなるためである。そして、前者が大きいほど、そして同時に後者が小さいほど、ユーザにとってその機器は理解がしやすく、また効率よく使えるようになるものと考えられる。

これを実現するためには、システムの開発組織に属するメンバーが、開発者、あるいは営業者、経営者としての意識と並行して、個人的な感覚、とくに任意のシステムに対する“初心者としてのユーザ意識”を持続することが大切であると考えられる。自分をユーザとしての立場から俯瞰した際に、対象システムが、多すぎる・複雑すぎる仕組みや多方面にわたる機能などを含んでおり、そのために使いやすさを低下させていると感じられる場合には、その感覚を信頼すべきである。それに基づいて、複雑さを低減したり、機能数を減少させたりといった改善を提案することは、開発組織メンバーの義務である。常にその意識を開発中のシステムに対して向け続けることができれば、この問題は改善される。もしこれが困難である場合には、初心者ユーザを想定しての人間中心的な開発プロセス [9] を導入することが望ましい。具体的には、製品の開発過程において、初心者ユーザを用いての利用テストを必ず行うことである。これにより、開発者自身の「わからないことがわからない」状態を低減させ、使いやすさの改善に向けた指標を得ることができるだろう。

そして、この観点をとり入れた、開発者に対する心理的な方向付け教育を、日本における情報・工学部系の諸学部で、学部教育の早期段階から取り入れることが望ましい。なぜなら、将来のわが国の開発者が、前世紀の価値観である“付加価値 = 多機能化”を、それと意識することなく持ち続けてしまい、その結果、ユーザにとって“高機能”だが“価値の低いモノ”たちを生産し続ける、という事態は、今後は避

けなければならないためである [10]。

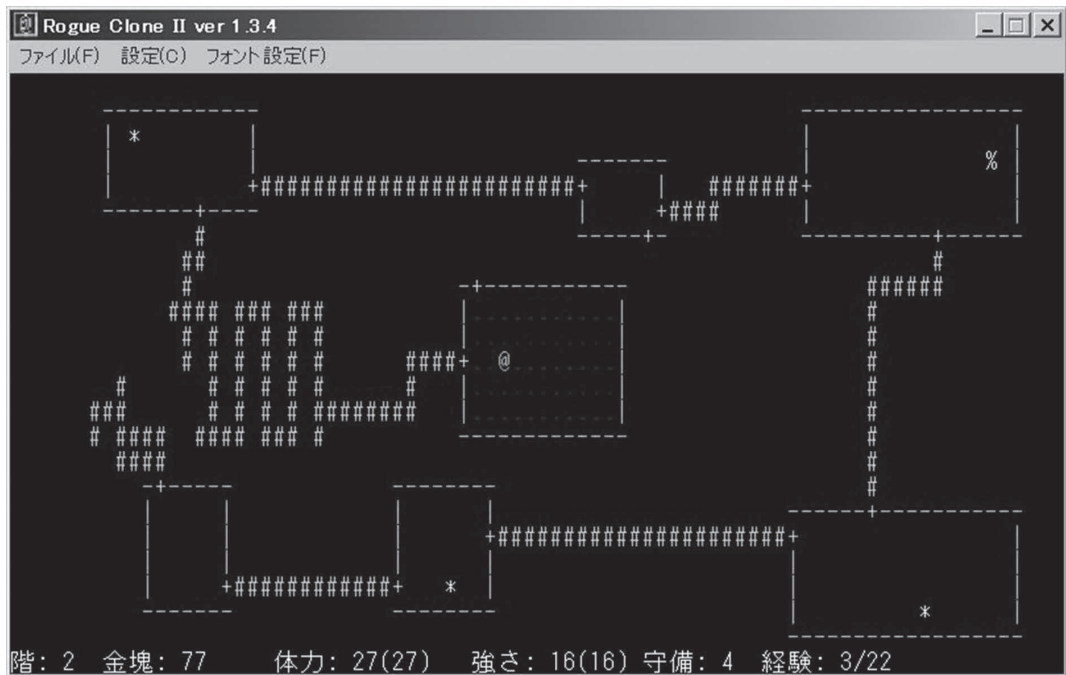
引用文献

- [1] スマートフォンに関する調査 約 4 割は、スマートフォンの半分も使いこなせていない
http://www.macromill.com/r_data/20110228smartphone/20110228smartphone.pdf
 2015/09/01 確認
 女子の意識調査！ 自分はスマホを使いこなせていると思う？ ⇒ 71.8% 思わない
http://www.excite.co.jp/News/woman_cdm/20150828/Escala_20150828_5163824.html
 2015/09/01 確認
 「ガラパゴスリモコン」が示す日本のものの作りの限界〜リモコンで飛行機でも飛ばしたいの？
<http://agora-web.jp/archives/1447660.html>
 2015/09/01 確認
- [2] 2014 年度通期国内携帯電話端末出荷概況 (MM 総研)
<http://www.m2ri.jp/newsreleases/main.php?id=010120150514500/>
 2015/09/01 確認
 「スマホはパソコンであって電話ではない」
 「ガラケー」復帰が進む理由はここにある
<http://www.j-cast.com/2015/03/01228260.html?p=all>
 2015/09/01 確認
- [3] FREETEL の SIM フリー「ガラケー」 Simple、発売 3 日で販売終了。増産予定無し
<http://japanese.engadget.com/2015/09/01/freetel-sim-simple-3/>
 2015/09/01 確認
 Nokia 222 Dual SIM(日本での発売は未定)
<http://www.microsoft.com/en/mobile/phone/222-dual-sim/>
 2015/09/01 確認
- [4] (日本経済新聞) 従来型携帯の生産終了 国内各社、17 年以降

- http://www.nikkei.com/article/DGXLASDZ21H8H_T20C15A4MM8000/
2015/09/01 確認
ソフトバンクモバイル宮内社長「ガラケーは必要なくなると考えている」
<http://www.j-cast.com/2015/05/19235613.html>
2015/09/01 確認
なお、この「ガラケー開発中止」の要因は、機能的な問題よりも部品代の高騰によるものと説明されている（私信）。また、上記 [3] の Simple についても、追加発売を行わない理由が、部品の入手困難のためと説明されている。
- [5] Strategy Letter IV: Bloatware and the 80/20 Myth
<http://www.joelonsoftware.com/articles/fog0000000020.html>
2015/09/01 確認
- [6] ノーマン, D.A. 誰のためのデザイン (1990) p.245
(Norman, D. A. 1988 The Psychology Of Everyday Things. Basic Books, New York)
- [7] 日本の家電は、なぜ“多機能”なの
<http://www.itmedia.co.jp/pcuser/articles/1004/21/news013.html>
2015/09/01 確認
- [8] ノーマン, D.A. (同上) p.281
スマホ UI 考 (番外編) なぜ機能追加をし続けるとアプリが破綻するのか?
<http://fladdict.net/blog/2013/08/appli-toomany-function.html>
2015/09/01 確認
スマホ UI 考 (番外編) 顧客やユーザーの要望に全て対応すると、アプリは 99% 破綻する
<http://fladdict.net/blog/2013/08/client-user-request.html>
2015/09/01 確認
日本の家電は、なぜ“多機能”なの
<http://www.itmedia.co.jp/pcuser/articles/1004/21/news013.html>
2015/09/01 確認
- [9] 黒須正明・松原幸行・八木大彦・山崎和彦 (2013) 人間中心設計の基礎 近代科学社.
- [10] 「トランスフォーマー 3/ ダークサイド・ムーン」劇中における日本製コピー機の評価
http://coco.to/movie/14732/review_bad/20
2015/09/01 確認
なぜ、インド人はサムスンを買うのか?
<http://toyokeizai.net/articles/-/27871>
2015/09/01 確認

付録 1 Rogue

Rogue は「平面ダンジョン探索型」のゲームの始祖である (付録図 1 参照)。グラフィック表示ではなく、テキスト記号を用いた二次元平面表示の世界の迷路を探索する。キャラクターやアイテム等はすべてキャラクター記号で示される。また迷路等のゲーム状況はプレイを行うごとにランダム生成される。この Rogue のインターフェイスを改善した各種ゲームが販売されている (「トルネコの不思議なダンジョン」 「ローグ・ハーツダンジョン」 等)。また本ゲームには必勝法がなく、クリアまでの難易度は高い。

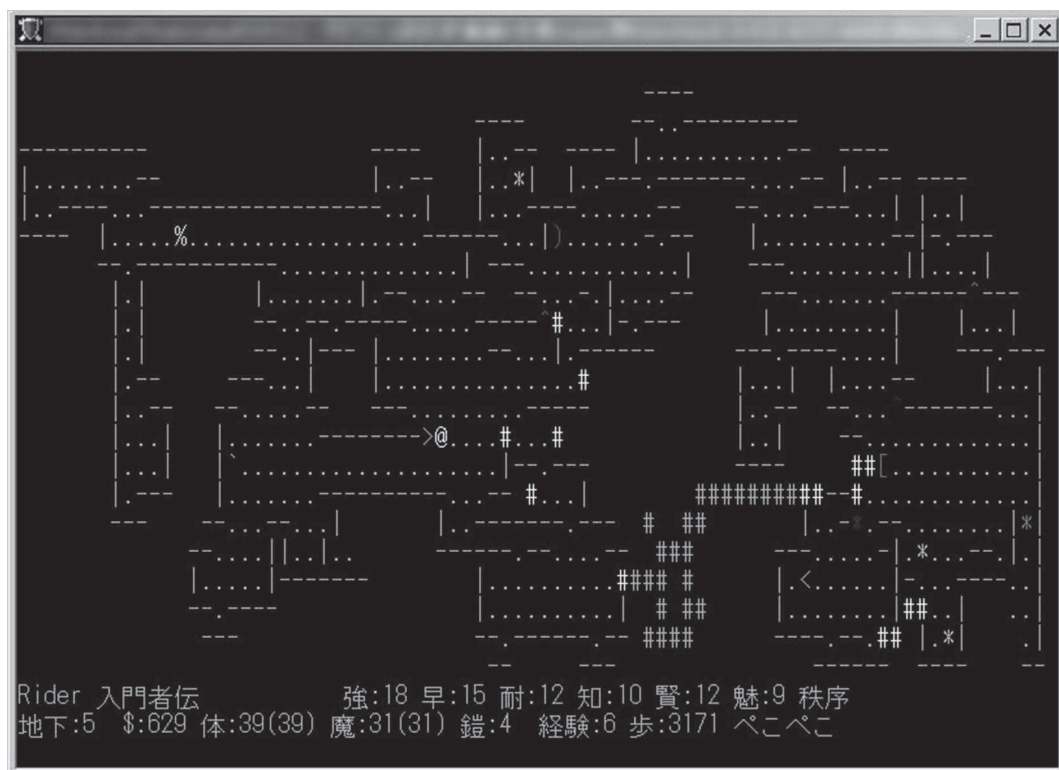


付録図 1 Rogue の画面例

付録 2 Nethack

Nethack は Rogue をいくつかの側面で大きく拡張したゲームである（付録図 2 参照）。表示面では Rogue と同様のキャラクターベースの表示であり、操作もキーボードで行うが、たとえば主人公の立場（秩序・中立・混沌）や職業（戦士・魔法使い・観光客など）、ストーリーの実装、それにとまなうアイテムやイベント、コマンド種類の増加など、多方面において拡張がなされた。ただし、スポイラーと呼ばれる「攻略情報」が公開されており、難易度は高いが、順序立った攻略は不可能ではないとされる。

なお Rogue と Nethack の機能数の比較は付録 3 に記した。



付録図 1 Rogue の画面例

付録3 :RogueとNethackの機能数の比較

	Rogue	Nethack
自キャラクター	1種類(選択肢なし)	種族5種類・職業数13種類・属性3種類・性別2種類
キャラクターのステイタス表示	5種類	14種類
helpによる表示コマンド数	43種類	142種類
環境設定コマンド	13種類	(起動オプションとして)70種類
敵の種類	27種類	60種類×それらの亜種(+イベントキャラクター多数)
道具類	7種類	13種類
備考		ペットとともに戦うことも可能