SAMPLE

第一部　コンピュータに目がくらんで

●このファイルは、株式会社翔泳社が発行する書籍『コンピュータは、むずかしすぎて使えない！』
（アラン・クーパー著、山形浩生訳、定価：本体2,200円＋税、ISBN 4-88135-826-X）
の内容見本です。商業的利用を目的とした複製・再配付を禁じます。
●この本のお買い求めはお近くの書店、または翔泳社のインターネットショップ
SEshop.com（http://www.seshop.com/）へご注文ください。
●この本の内容に関するお問い合わせは、826-x@shoeisha.co.jpまで電子メールでお送りください。

SAMPLE

第一章
情報時代のなぞなぞ

コンピュータと飛行機のあいの子ってなーんだ？

一九九五年十二月、アメリカン航空の９６５便はマイアミからコロンビアのカリへ、いつもどおりに離陸した。着陸間際に、このボーイング７５７のパイロットは次の電波航行目標をＲＯＺＯにセットしなくてはならない。パイロットは、航行コンピュータにＲを入力。すると、コンピュータは、近くの航行目標のうちＲで始まるものの一覧表を表示した。緯度も経度もそれらしかったので、パイロットはリストのてっぺんにあるものを選んだ。が、不幸なことにパイロットが選んだのはＲＯＺＯではなく、そこから北東三〇〇キロのところにあるＲＯＭＥＯだった。

ジェット機は南に向かって飛んでいて、南北に走る谷に降下中だ。そこから横にずれると非常に危険な状況だ。だがフライトコンピュータの指示に従って、パイロットたちは機首を東に向けようとして、高度三千メートルで山に激突。乗客百五十二人と乗務員八名が死亡、助かった四人の乗客も重傷。アメリカ運輸安全委員会が調査を行って――いつものように――これは人間の操作ミスによるものだと発表した。

パイロットたちが従っていた航行支援システムは、まちがってはいなかったけれど、カリでの着陸には有効ではなかったわけだ。確かに文字通りにいえば、これは人間の操作ミスではある。パイロットがまちがったポイントを選んだのが悪い。でも、もっと広い目で見れば、これはパイロットのせいなんかではまったくない。

飛行機の航行コンピュータのパネルには、いま選ばれている航行目標と、そこからの逸脱を示す目盛りがついている。飛行機がコース上にあれば、針はまんなかにあるけれど、その針は選んだビーコンが正しいかどうかについて、何一つヒントを示してくれない。このゲージは、着陸直前だろうと激突直前だろうと、まるっきり同じ状態だ。コンピュータは、パイロットが選んだビーコンに向かうコースをきちんとたどっていることは告げてくれる。でも、そのビーコンの選択が致命的なものだということは教えてくれないのだ。

コミュニケーションは、厳密で正確であっても、同時にどうしようもなくまちがっていることもある。これはコンピュータとのやりとりでしょっちゅう起きることで、しかもコンピュータは現代生活のいたるところに入り込んでいる。われわれの乗る飛行機から、ありとあらゆる家電製品やサービスまでコンピュータはほとんどどこにでもあるし、そのどれも、コミュニケーションや動きがどうしようもなくお粗末だという点で共通している。

コンピュータ業界の有名な冗談にこういうのがある。ある人が、小型機で飛んでいるうちに雲のなかで迷

SAMPLE

SAMPLE

子になった。そこで降下してみると、オフィスビルがあったので、そこの開いている窓に向かって怒鳴った。「すいませーん、わたしいま、どこにいるんでしょうか?」すると窓の中の人はこう答えた。「あなたは地上三十メートルくらいを飛行中の飛行機の中にいます」パイロットはすぐに正しい航路に戻って、空港を見つけて着陸した。乗客たちは驚嘆して、どうしてあれで行き先がわかったのかをきいた。パイロット曰く「あの人の答は、完全に正しくて本当のことだったけれど、でもまったく何の役にもたたなかったでしょう。だからすぐに、あれがマイクロソフトのエンジニアだとわかったんですよ。マイクロソフトの建物から見て空港がどこかは知っていましたから」

９６５便の悲劇を思うと、このジョークのオチは陰惨に聞こえるけれど、でもデジタル業界の人たちはこのジョークをしょっちゅう嬉しそうに話す。それがコンピュータについての根本的な真理を言い当てているからだ。

コンピュータは、事実は教えてくれるけれど、役にたつことは教えてくれない。正確な案内はするけれど、いきたいところへは案内してくれない。９６５便のコンピュータは、パイロットたちに「カリへの進入にＲＯＭＥＯは適切じゃないですよ」と教えられたはずだ。その選択が変だとか、疑問だと匂わせるだけでも、この飛行機は救えただろう。でも、コンピュータは実際のフライトやその乗客たちのことは一切気にしていなかった。自分の中の計算のことだけを気にしていたわけだ。

使いにくいコンピュータはわれわれみんなに影響を与えるし、しかもその影響は致命的だったりする。ソフトウェア製品は、別に必然的に使いにくいわけではない。ただ、それをつくるときのプロセスがまちがっ

SAMPLE

ているのでそうなるだけだ。

この本では、この悪しきプロセスの影響とその原因を示すことで、それを明るみにだしてやろうと思う。それからそのプロセスを変えることで、ソフト製品を親切で強力で望ましいものにするやりかたを示そう。この章ではまず、問題がどれほど深刻かを示す。

コンピュータとカメラのあいの子ってなーんだ?

情報時代のなぞなぞを一つ。「コンピュータとカメラのあいの子ってなーんだ?」答:「コンピューター!」

三十年前のわたしの最初のカメラは、35ミリのペンタックスＨで、光量メーターの電源用に小さな電池が入っていた。腕時計の電池と同じように、それを何年かおきに換えるだけ。

十五年前には、初のオートマチックカメラを買った。35ミリのキヤノンＴ70で、単三電池二本を電源にして、単純な露出制御のコンピュータと自動フィルムドライブが使えた。簡単なオン／オフスイッチで、電池が無駄に減らないようになっ

SAMPLE

ている。

五年前に、フィルムを使わないロジテックのカメラを買った。第一世代のデジタルカメラで、これにも似たようなオン／オフスイッチがあるけれど、今度のは中身にちょっとしたコンピュータが入っている。だからスイッチを切るのを忘れても、一分間操作がないと、自動的に電源が切れる。なかなかいいね。

一年前に第二世代のデジタルカメラを買った。パナソニックのPalmCamだ。中にはもっと頭のいいコンピュータチップが入っていて、オン／オフスイッチも、オフ／録画／再生スイッチにまで進化した。カメラにモードができたわけだ。録画したければ録画モードにして、小さなビデオ画面に出したいときは、再生モードにするわけだ。

最新のカメラはニコンのCoolPix 900だ。第三世代のデジタルカメラで、いまだかつてないほどインテリジェント化されている。中には、完全に立派なコンピュータが内蔵されていて、「起動」するときにウィンドウズみたいに砂時計まで出てくる。頭がたくさんついた奇形の魚みたいに、オンオフスイッチもすすんで、ポジションが四つもある。Ｏｆｆ／ＡＲｅｃ／ＭＲｅｃ／Ｐｌａｙ。ＡＲｅｃというのは自動録画で、ＭＲｅｃというのはマニュアル録画なんだけれど、わたしに言わせれば、どっちもぜんぜんちがわない。「オン」というポジションはなくて、友だちも、長々と説明を受けないと、だれ一人として使い方がわからない。

この新しいカメラはとても電気を喰うので、エンジニアたちは親切にも、バッテリー消費を管理する高度

SAMPLE

なプログラムをつけてくれている。よくあるシナリオはこんな感じだ。

邪悪なオフ・なんとかスイッチを、ＭＲｅｃにあわせて、カメラが起動するまで七秒も延々と待って、目標にカメラを向ける。カメラをかまえて、いい構図になるようにズーム。ちょうどシャッターを押そうとしたところでこのカメラは、ズームと、フラッシュの充電と、画像表示とを同時にやると電力が足りないことに気がつく。そこで自衛のため、カメラは写真を撮る機能をサスペンドしてしまう。

が、わたしはそんなことは知らない。ファインダーをのぞいて腕をふりまわしつつ「笑って笑って」と言いつつシャッターを押しているからだ。コンピュータは、シャッターが押されたのは知っているけれど、でもいまは残念ながらダメなんですよ、というわけ。そこで、なんとかしてあげましょう、というわけで、電力管理ソフトがしゃしゃり出てきて、上位判断を下す。配電統制をかけるのだ。そして、電力喰いの液晶ディスプレイを消してしまう。

わたしは、あれれ、とカメラを眺め、なんで写真が撮れなかったのかな、と不思議に思い、肩をすくめて、カメラを持った手をおろす。でも、液晶が消えると、ほかの機能にまわす電力に余裕ができる。電力管理ソフトはこれを検出して、ああこれで写真を撮るだけの電気がある、と気がつく。そこで、カメラのプログラムに制御を戻す。このプログラムは、わたしがシャッターボタンを押したときの指令を実行しようとして、ずっと待っていたわけだ。

そして、オートフォーカスで、露出も計算して、とても高画質の写真を撮ってくれる。わたしの膝小僧の写真を。

むかしの機械式ペンタックスは、ピントも手動、露出も手動、シャッター速度も手動だったけれど、完全

SAMPLE

にコンピュータ化されたいまのニコンCoolPix 900より、ずっと頭にくることは少なかった。ニコンのほうは、ピントも露出もシャッター速度も自動なのに。こいつは、写真は撮るけれど、やることはカメラよりはコンピュータになっているわけだ。

⌘

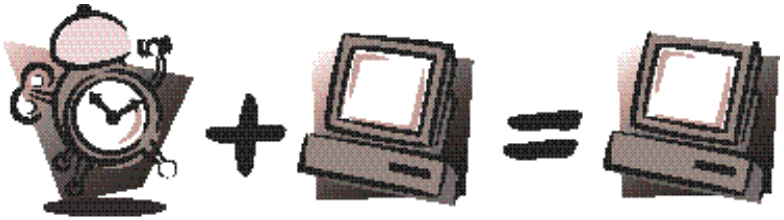
カエルを水に入れて火にかけると、カエルは水温の上昇に気がつかない。熱でカエルの感覚は麻痺してしまうからだ。わたしもカエルと同じく、カメラが使いやすいものから、コンピュータ化されて使いにくいものへとゆっくり移行したのに気がつかなかった。われわれみんな、日常生活の中にコンピュータ的なふるまいがジワジワ入り込んでいるのに、麻痺してそれに気がついていないのだ。

コンピュータと目覚まし時計のあいの子ってなーんだ？

答：コンピュータ！

わたしはちょうど、ベッドルーム用に高い新品の時計つきラジオを買ったところだ。JVC FS-2000だ。とても高度なコンピュータ頭脳を持っていて、ハイファイ、デジタルサウンド、機能てんこ盛り。時間をセットしておけばCDをかけて起こしてくれるし、朝六時にセットしておくと、ゆぅぅぅぅっくりボリュームを

SAMPLE



上げていってくれるだけの繊細さと知恵を持ち合わせている。この機能はほんとに気持ちがいいし、なかなかユニークだ。うっとうしい機械を窓から放り出したくなる気分を、補ってあまりある。

ただアラームがセットされているかどうかを見るのはとってもむずかしいので、ときどき月曜の朝に起こしてくれなかったり、土曜早朝にベッドからたたき起こしてくれたりする。はいはい、確かにアラームがセットされているかを示す表示はあるけれど、それが使いものになるか？

この時計には高機能の液晶ディスプレイがついていて、機能のすべてはここに出てくる。液晶の左上に小さな時計マークが出ていると、アラームがセットしてあるわけなんだが、薄暗いベッドルームでは、そんなもの見えやしない。液晶のバックライトがつけば時計マークも見えるが、CDプレーヤーかラジオが手動でオンになっていないと、このバックライトはつかないのだ。しかしながら、ここが落とし穴で、CDを手動でつけてあると、アラームはどう設定してあっても絶対に鳴らない。この堂々巡りのおかげで、わたしはしょっちゅう寝込みを襲われることになってしまう。

アラームを解除するのは簡単だ。アラームボタンを一回押せばいい。すると時計マークは消える。でもアラームをセットするには、アラームボタンを正確

SAMPLE

に五回押さなくてはならない。

一回押すと、アラームの時間が表示される。二回目では、アラームのサウンドが切れる時間が表示される。三回目では、アラームでラジオが鳴るかCDが鳴るかが表示。四回目では、そのボリューム。五回目で、やっと最初の状態にもどって、このときにはアラームがセットされている。

でも、もう一回押すだけで、そのアラームは解除されてしまうわけだ。眠い頭で、暗いベッドルームで、このちょっとしたデジタルバレエを正確に踊るというのはなかなかの難行だ。

わたしはおたくがかった小道具マニアなので、いずれ使いこなせると期待してこの装置と格闘を続けている。でも妻は、この邪悪なマシンをとうの昔に見放した。精悍でモダンなデザインは好きだし、音質のよさも好きだけれど、でも使うのがむずかしすぎて、目覚まし時計として役にたたないからだ。この機械は、目覚まし役もするけれど、でもやることはコンピュータなのだ。

それにくらべて、古い十一ドルのコンピュータなしの目覚まし時計は、いきなり騒々しいブザーでこっちを起こす。セットしてあると、赤いランプがつく。セットしていないと、それが消える。この目覚ましにはいろいろ気に入らないところはあるけれど、でも少なくとも、目を覚ましてくれるかどうかはちゃんとわかった。

⌘

SAMPLE

メーカーとしては、装置の内部機能を制御するのに、古い機械装置を使うよりもコンピュータを使うほうが安上がりだ。だから経済的にも、人の生活上のありとあらゆる製品やサービスに、コンピュータが忍びこんでくるのは避けられない。

ということはつまり、身の回りのあらゆる製品が、いずれ最高に頭にくるコンピュータじみた振る舞いをするようになるということだ。なにかやり方を変えない限り。

⌘

この現象は、消費財に限った話ではない。コンピュータ化された装置やサービスは、ほとんど必ず、マニュアル式のものよりも機能やオプションが増えている。でも現実には、シリコンチップ技術を使った現代版装置を使うよりも、マニュアル式の装置を使うほうが、柔軟性も高いし、細やかさも、わかりやすさの面でもずっと使いこなせるのだ。

ハイテク企業は――製品を改良しようとして――単にややこしいだれも使わない機能を足しているだけだ。開発プロセスは、この悪い製品の問題を解決するどころか、新しい機能を足すようにしか機能しないので、メーカーは機能をひたすら増やす。この本では後で、いらない機能を足して余計な作業をせずに、ユーザーを喜ばせるような、もっと優れた開発プロセスを紹介しよう。

SAMPLE

コンピュータと車のあいの子ってなーんだ？

コンピュータ！

ポルシェの見事な最新式ハイテクスポーツカー、ボクスターは、複雑なシステム制御にコンピュータを七つ使っている。一つはエンジン制御専用。そして異常事態に対処するように、特別な処理もついている。が、それがときどき墓穴を掘る。

初期のモデルでは、ガソリンがかなり少なくなると（残り一リットルとか）、急カーブをまがったときの遠心力で、燃料パイプに空気が入り込むことがあった。コンピュータはこれを、燃料そのものが何らかの理由で急変したんだと解釈して、これは燃料噴射システムが完全にイカレたんだと判断。ダメージを避けるべく、コンピュータはイグニッションをシャットダウンして車を停めてしまう。

さらに被害拡大を防ぐべく、このコンピュータはエンジンを再起動させてくれなくなる。レッカー車を呼んで、サービス工場で修理を受けろ、というわけだ。

初期のボクスターのオーナーたちがこの問題に気がついたとき、ポルシェ側が出せる解決方法は一つしかなかった。エンジンルームを空けて、バッテリーを五分ほどはずすように。そうすればコンピュータは、この事故のことを忘れますから。スポーツカーは、二車線アスファルト道路を高速で下ってはいくけれど、急カーブになると、コンピュータみたいにふるまうのだ。

ボクスターのオーナーたちを守ろうという立派な努力のために、プログラマたちはそのオーナーたちを、

SAMPLE

バカにされた被害者にしたててしまった。

高性能乗用車マニアならだれでも、ポルシェはその顧客に対して敬意と特別扱いを大盤振る舞いするのを知っている。こんなことが見逃されたと言うことは、この車の中のソフトが、車のその他の部分をつくっているポルシェとはちがうポルシェからきているということを物語る。それは会社の中の会社からきている。プログラマという会社。伝説的な自動車エンジニアたちではない。

どういうわけか、新技術の導入のせいで、伝統ある高名な会社はまごついてしまい、その看板ともいうべき価値観の一つを台無しにしてしまったことになる。ソフトエンジニアにとっての品質基準は、これまでの工学分野の基準よりずっと低いのだ。

コンピュータと銀行のあいの子ってなーんだ？

コンピュータ！

キャッシュディスペンサーからお金を引き出すたびに、あのコンピュータ特有の、つっけんどんで扱いにくいふるまいに出くわすことになる。ちょっとでもまちがえたら、これまでの操作全部がダメになって、振り出しに戻される。カードを引き出して、差しこみなおして、暗証番号を入れて、またこちらの要求を出す。それにふつうは、そのまちがいもわたしのせいではなくて、この機械のせいでまちがえるようにし向けられるのだ。

SAMPLE

こいつはいつも、お金をどの口座から引き出すかきいてくる。小切手用の口座か、普通の口座か、投資用口座か。でも、わたしは小切手用の口座しか持っていない。だからいつも、その口座の名前を忘れてしまい、機械にきかれてまごつくはめになる。月に一度くらい、うっかり普通口座を選んでしまうと、この地獄の機械はこちらの操作を完全にキャンセルして、最初からやりなおせと言ってくる。

「普通口座」をはねるわけだから、この機械はわたしが普通口座なんか持っていないのを知っているわけだ。でも、それを平気で選択肢に出してくる。わたしが「普通口座」を選ぶのと、９６５便のパイロットがＲＯＭＥＯを選ぶのとで何か差があるだろうか？　結果として受ける罰の大小がちがうくらいだ。

この機械はまた、引き出し限度額を一日二〇〇ドルとしている。すべてのステップをちゃんとこなしても――暗証をちゃんと入れて、口座も選んで、金額も選んで――そして二二〇ドルほしいといったら、コンピュータは全操作をキャンセルして、おまえは限度額を超過した、と無礼に言ってくる。その限度額がいくらなのか、あるいは口座の残高がいくらなのかは教えてくれないし、金額だけ入れ直すのもやらせてくれない。単にカードを吐き出して、こっちは最初っからやりなおすしかないし、何の新しい情報ももらっていない。背後の行列はどんどん増えて、みんなもじもじ、キョロキョロ、ため息。

キャッシュディスペンサーは、厳密で正確ではあるけれど、何も助けてくれないのだ。

向こうには向こうの決まりがあるのはわかるし、それにしたがうのはやぶさかではない。でも、その決まりがなにかを教えてくれなかったり、それに反するような選択肢をよこしたりしておきながら、無邪気にも決まり以外のことをしてしまうとすぐに罰するというのは、どうしようもなくコンピュータだ。この行動

SAMPLE

――いかにもコンピュータ――は、コンピュータ固有の性質ではない。

実はコンピュータにとっては、何一つ固有ではない。コンピュータはソフトウェア、つまりプログラムのかわりに行動するだけなんだから。そしてプログラムは、人間のことばと同じくらい融通がきく。

人は、無礼にもしゃべれるし、礼儀正しくもしゃべれる。親切にもなれるし、意地悪にもなれる。人間が敬意をもって親切にふるまうのは簡単だし、コンピュータだって同じくらい簡単にそういうふるまいができるのだ。だれかがそのやりかたを示してやれば。残念ながら、プログラマという連中は、それをコンピュータに教え込むのがあまり得意ではないのだ。

コンピュータは簡単に面倒を引き起こせる

デスクトップのコンピュータは、昔ながらのコンピュータと同じ、頭にくるふるまいを続けている。何ともあいの子にする必要はない。

友だちのジェーンはむかし、広告代理店で顧客管理部門の仕事をしていた。メモや契約を書くときには、ウィンドウズ95で走るマイクロソフト・ワードを使っていた。ウィンドウズ95の中核は、階層型ファイルシステムだ。ジェーンの文書はすべて小さなフォルダに入っていて、そのフォルダがさらに別のフォルダに入っている。

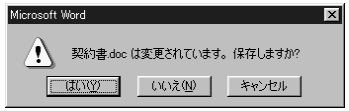
ジェーンはそんなことは理解していなかったし、そんなふうにものを保存するメリットもわからなかった。

SAMPLE

というか、ジェーンはそんなことはまるで考えたりせずに、いちばん面倒のないやりかたをしていただけだった。

ジェーンはちょうど、シリコンバレーの新興企業とPR契約案を書き上げたところだった。ファイルメニューから「閉じる」を選ぶ。ワードは、指示通りにファイルを閉じるのではなく、ダイアログボックスを出してきた。もちろん、もう腐るほど見た「文書は変更されています。保存しますか?」確認ボックスだ。彼女は――いつもとまったく同じく――「はい」を押した。もういつも必ずそう答えていたので、もうダイアログをいちいち見もしていなかった。

最初のダイアログに続いて、すぐに次のが出てきた。同じく腐るほど見た「ファイル名を付けて保存」ボックスだ。ややこしくてわけのわからないボタンやアイコンやテキスト入力欄が並んでいる。ジェーンに理解できて使えるものといったら、「ファイル名」の入力欄だけだ。彼女は適当な名前を入れて、「保存」ボタンを押した。するとソフトはそれを「マイ　ドキュメント（My Documents）」フォルダに入れた。ジェーンはこのどうでもいい演習に慣れっこになれすぎていて、なにも考えていなかった。



SAMPLE

昼休み、ジェーンがオフィスにいないとき、会社のコンピュータ技師のスンイルがウィルス対策ソフトの新しいバージョンを彼女のマシンにインストールした。ジェーンのマシンで作業しつつ、スンイルはそのソフトの説明文書readme.docを読むのにワードを使った。そしてそのあとでスンイルは文書を閉じ、昼休み前とまったく同じ状態にマシンを戻した……とスンイルは思った。

昼休みから戻ると、ジェーンはさっきのPR契約を開き、印刷して上司に見せようとした。ファイルメニューから「開く」を選んで、「ファイルを開く」ダイアログが出てきた。

ジェーンはそこに、自分の契約書や文書がABC順にきれいに並んで出てくるものと思っていた。が、かわりに出てきたのは、見たこともない、きいたこともないファイル名ばかり。その一つがreadme.docという名前だったのだが……。

もちろんスンイルは、このreadmeファイルを読むのにワードを使ったわけだ。そしてそのときに、ジェーンのワードを六階層も深い変なフォルダを見るように指示してしまい、うっかり通常の「マイ　ドキュメント」フォルダから引き離したのである。

いまやジェーンは完全に面食らっていた。真っ先に思ったのは、これまでいっしょうけんめいやってきた仕事が全部消されたんじゃないかということで、彼女はとても不安になった。そこで友だちの同僚ルネにき

SAMPLE

いてみたけれど、ルネもさっぱりわからない。パニック寸前の状態になって、やっと彼女はスンイルに電話して助けてもらおうとしたけれど、スンイルは外出中で、月曜の朝になってやっと、立ち寄って事態を収拾できた。ジェーン、ルネ、スンイル――そしてこのＰＲ会社――はみんな、半日分の生産性を無駄にしたことになる。

コンピュータのオペレーティングシステムは、階層型ファイルシステムを必要としている。でも、それを使う人々にはそんなものは要らないのだ。コンピュータのプログラマは、ベースとなる階層型ファイルシステムを見たがるかもしれない。でも、ジェーンのようなふつうのユーザーがそんなものを気にしないのも、同じくあたりまえではないか。

でも、われわれみんなが使うソフトをつくるプログラマにとって、それはあたりまえには思えないようだ。かれらは、自分たちのいちばん好きなふるまいと情報の提示をつくりあげるけれど、それはジェーンにとっていちばんいいふるまいや情報の提示とはぜんぜんちがうわけだ。

ジェーンのいらだちと能率の悪さは、ジェーンのせいにされる。彼女に爆弾をしかけたプログラマたちのせいではなく。

少なくともジェーンには仕事がある。でも「コンピュータリテラシー」が低いと思われて、だから仕事がない人もたくさんいるのだ。そしてコンピュータとのやりとりが必要な仕事が増えるにつれて、仕事に就ける人と就けない人との格差も広がるし、それを埋めるのもむずかしくなる。

政治家たちは、恵まれない人たちのために仕事をつくれと要求するかもしれないけれど、コンピュータを

SAMPLE

使う能力がなければ、企業だって訓練を受けていない人たちにコンピュータをさわらせるわけにはいかないだろう。必要な訓練は多すぎるし、データを破壊したり、貴重なデータベースをめちゃくちゃにしたりする危険も大きすぎる。

ソフトウェアを使った製品が示す、むかつくふるまいやわけのわからぬ操作は、わたしが「ソフトウェア・アパルトヘイト」と呼んでいるものを社会の仕組み化してしまっている。ごくふつうの人が、コンピュータを使えないというだけの理由で、労働市場に参加できず、社会参加ができない状態だ。

われらが啓蒙された社会では、社会活動家がいっしょうけんめい、人種や階級の障害をうち倒そうとしている横で、テクノロジストたちがいっしょうけんめい、意図せずに、新しいもっと大きな障害をつくりだしているわけだ。ソフト製品を意識的にもっと人間的で寛容なものにすれば、もっともっと万人を含む、階級や肌の色に関係ないものにできるのに。

商業ソフトだって困っている

コンピュータの侵略は、ジェット旅客機のコクピットだけではない。乗客の客室にまで入り込んできている。そしてあのあまりにおなじみで使いにくい、むかつく変なやりかたでふるまってくれる。

いまの旅客機には、機内エンターテインメントシステム（ＩＦＥ）があって、乗客に映画や音楽を提供する。

SAMPLE

このＩＦＥは、ちょうどオフィスと同じ、ＬＡＮでつながったただのコンピュータだ。高度なＩＦＥシステムは、ふつうは大洋を越える路線の大きな航空機にしか設置されていない。

ある航空会社のＩＦＥは、客室乗務員にとってあまりに使いにくかったので、乗務員の多くは短い国内路線を希望して、このむずかしいシステムを勉強して使ったりしないですむようにしていた。

これは驚異的なことだ。昔ながらの航空路線の希望はずっと年功序列の世界だったし、長距離国際線は、シンガポールやパリのようなエキゾチックな場所で長いステイがあるので、いちばん希望の多い路線だったからだ。

客室乗務員がＩＦＥを使いたくないばっかりに、つまらないし華やかでもないデンバー・ダラス往復やロサンゼルス・サンフランシスコ便なんかの希望を出すというのは、かなり深刻な労働意欲上の問題を示している。いちばん有能な職員――顧客とすごす時間の一番多い職員――に、ひどいツールをおしつける航空会社は、おろかな判断をしているし、お金や顧客の信頼や、職員の意欲をドブに捨てているようなものだ。

別の大航空会社のコンピュータ式ＩＦＥはもっとひどかった。この航空会社は、映画の配信を集金システムと結びつけるような機内エンターテインメントをつくりあげた。

さて、高度一万三千メートルを飛んでいる、密閉されたジェット旅客機では、集金手続きはそんなに厳格にはなっていなかった。だって、だれも逃げ出すわけにはいかないんだから。乗務員は、手の空いたときに商品やサービスを提供したし、集金をその場ですぐに要求したりもしなかった。そうすれば、せまい通路を不必要にいったりきたりしなくてもすむ。はいはい、たまにミスはあったけれど、でもどうせたかだか数ド

SAMPLE

ルの話だ。そこそこ人間的で寛容なシステムだし、みんな満足していて、仕事も抑圧的ではなかった。

ところが集金システムが、コンピュータのコンテンツ配信と結びつくようになると、乗務員はまず乗客からお金を集めて、それから乗務員用コンソールのあるキャビンのずっと端まで戻って、乗務員パスワードを入れて、それからレジ係みたいな操作をすることになる。この操作が完了しないと、乗客は映画も観られないし音楽も聴けない。

このばかげた製品デザインのおかげで、客室乗務員はフライト中にせまい通路を何百回も余計にいったりきたりしなくてはならない。頭にきた乗務員たちは、長距離フライトでは離陸直後にこのＩＦＥのブレーカーを切ってしまう。そしてそしらぬ顔で乗客たちに、「システムがこわれていてこのフライトでは映画はごらんいただけません」とアナウンスする。

この航空会社は何百万ドルもかけたのに、できたシステムはあまりにひどくて、使わなくてすむようにユーザーがわざと切ってしまうような代物だったわけだ。退屈した乗客たち数千人は、あわれな犠牲者だ。しかもこれが、長距離の国際線で、みんなのほしがるリピート利用客のたくさん乗った便で起こるわけだ。これがこの航空会社にどのくらい損害を与えたか、数字はわからないけれど、それが大惨事級にものすごい金額だったということは、裁判官なみに断言してもいい。

ＩＦＥ内部のソフトはみごとに正確に機能したのだけれど、でもご主人たる人間たちに対するふるまいがまともでなかったので、とんでもない欠陥品となった。どうして企業は、こんな悲しい結果を予想できなか

SAMPLE

ったんだろうか。どうしてこの連想ができなかったんだろうか。

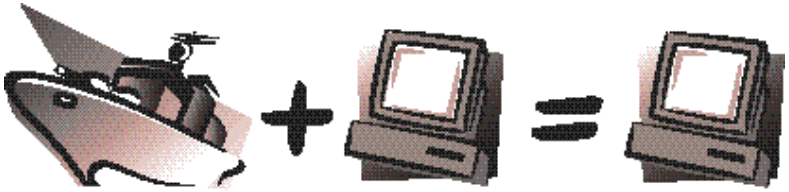
この本の目的は、いまの質問に答えて、こういうハイテク騒動を避けるための方法をお示しすることだ。

コンピュータと戦艦のあいの子ってなーんだ？

一九九七年の十一月、大西洋で艦隊訓練中の、最新海軍イージス艦ヨークタウンは、海上で完全停止に陥ってしまった。海軍の技師が、艦上の燃料バルブを調整中に艦上管理コンピュータにゼロを入力したためだ。

このコンピュータはウィンドウズNTを使ったペンティアム・プロマシンだった。プログラムは別の数字をこのゼロで割ろうとした――これは数学的に定義されていない操作だ。このために、艦上の管制システムすべてがクラッシュ。コンピュータがなかったために、エンジンは停止してしまい、この戦艦は波間に二時間と四十五分の間、ひたすら漂い続け、港まで曳航されなくてはならなかった。戦闘水域でなかったのがせめてもの救いだ。

コンピュータと戦艦のあいの子ってなーんだ？　ニミッツ提督は、墓の中



SAMPLE

で歯がみしていることだろう！　こんなできごとにもかかわらず、海軍は人件費の節約のため、戦艦すべてをコンピュータ化しようとしている。そしてこの計画への批判をかわすため、この「事件」は人為的な操作ミスのせいにされている。

ソフト生産プロセスが手に負えなくなっているので、ハイテク産業はそのプロセスをなんとかしなくてはならない。さもないと、ふつうのユーザーばかりを責めて、巨大化し続けるマシンは海上で身動きとれず、ということになり続ける。

テクノ憤怒

先日のウォールストリート・ジャーナルの記事で、ある匿名のビデオクリップが紹介されていた。これは電子メールで広く出回っているもので、「（前略）ひげをはやしたふつうの男が、半袖シャツをきて、コンピュータ端末に向かって身をのりだして、あれ？という顔をしている。いきなりその男は激怒して、モニタの横を叩く。同僚たちがなにごとかとかれの区画をのぞきこむなか、男はキーボードをモニタに叩きつけて、それを床にはたき落とす。男は椅子から立ち上がると、落ちたモニタに怒りの蹴りを入れてとどめをさす」。記事いわく、このビデオクリップに対する反応は「強烈」であり、これは明らかに「社会に流れる強力なテクノ憤怒の底流に触れたものだ」。

SAMPLE

このビデオを送ったり見たりするだけでも、それなりにコンピュータの知識がないとダメだというのは皮肉なことだ。このビデオに出てきた人物は、実は役者だとのことだけれど、それはビジネス界に広がっている同情心の琴線に触れたわけだ。使いにくくて不愉快なソフト製品は、われわれの生活にすさまじい勢いで大量のフラストレーションをもたらしているのだ。

私的なメーリングリストでは、「コンピュータ・トゥレット」なるものについての冗談が出回っている。これはトゥレット症候群という疾患をもじったものだ。この症候群にかかると、人はどうしようもなくひたすら罵倒を続けてしまう。

冗談というのは、最新のオフィスビルの通路を歩いていると、ごくふつうに見える人たちがモニタの前で歯ぎしりしていたり、何度も何度も怒りに満ちた呪詛を吐いているよ、というものだ。なにがそんな状態を引き起こしたのやら。ファイルがなくなったか、画像がアクセスできないか、操作がわけわからないか。それとも、そのソフトがたったいま、五百ページの原稿をあっさり消してくれたのかもしれない。確認ダイアログにうっかりＹｅｓと答えたら、それは実は「保存しますか？」ではなく「消去しますか？」だったから。

SAMPLE

認めようとしない業界

われわれはハイテクツールだらけの世界に住んでいる。コンピュータは職場や家庭を支配して、乗り物もシリコン製のガジェットまみれになってきた。こうしたコンピュータ化された装置は、とてつもなく高性能で強力だけれど、そのどれも、どうしようもなくむずかしくて、使うのが面倒だったりする。

ハイテク産業は、携帯電話やワープロを持っている人ならだれでもはっきりわかる簡単な事実を見ないようにしている。コンピュータ化されたツールはむずかしすぎて使えない、ということだ。

ソフトやハイテク装置をつくる技術至上主義者たちは、自分のやったことに満足している。それを作ったソフトエンジニア【原注1】たちは、全力をつくしてそれを使いやすくして、まあ微々たる進歩くらいはしている。自分たちの製品は、技術的に製造が簡単なのと同じくらい、使いやすいはずだと信じている。エンジニアたるかれらは技術を信じていて、ユーザーの体験を改善してくれるのは、音声認識や人工知能などの新しい技術だけだと考えている。

皮肉なことに、ソフト製品の使い勝手を高める上でいちばん貢献しないものが、新技術なのだ。複雑でわかりにくいプログラムと、単純で楽しくて強力な製品との間には、技術的にはほとんど差がない。問題はそれをつくる人々の文化と技術と態度の問題で、チップやプログラミング言語の問題ではない。われわれの欠陥は、開発プロセスであって、開発ツールではないのだ。

SAMPLE

ハイテク産業は、気がつかないうちにプログラマやエンジニアに支配権を与えてしまったので、かれらの使いにくいエンジニアリング文化が支配するようになってしまった。

見た目にはどうだろうと、ハイテク産業を支配しているのは企業重役なんかではない。舞台を仕切っているのはエンジニアたちだ。シリコンチップからのメリットを享受しようとあわてるあまり、われわれは責任を放棄してしまった。いまや、病人たちが精神病院を運営している状態なのだ。

病人たちが精神病院を運営してしまうと、自分たちを襲っている問題の性質をはっきり見るのはむずかしくなる。鏡を見ると、自分のいいところばかり見えて、イボは見逃してしまいがちだ。

ソフト製品の製作者が自分たちの手をかけた製品を見るときには、悪いところは目に入らない。そのすさまじいパワーと柔軟性しか見えないのだ。機能や特徴がいかに豊富かは見える。それがとんでもなく使いにくいとか、使い方を覚えるのにうんざりするほど時間がかかるとか、あるいはそれを毎日の暮らしの中で使わなくてはならない人々を、軽んじてバカにするものだということについては、無視してしまうのだ。

【原注1】コンピュータ業界では、「ソフトエンジニア」ということばは「プログラマ」ということばとまったく同義で使われている。この本で、わたしもそうしている。

SAMPLE

この本のきっかけ

わたしは二十五年にわたって、ソフトウェア製品を発明したり開発したりしてきた。この使いにくいソフトという問題のおかげで、わたしは何年も頭を抱えて途方にくれてきた。

とうとう一九九二年にわたしはプログラミングをやめて、ほかの開発企業が製品を使いやすくする手伝いに全時間を投入することにした。すると、すばらしいことが起きた！

プログラミングの要求から解放されてみるとすぐにわかったのは、そういう要求がいかに強くて支配的かということだった。プログラミングというのは、じつにむずかしくて没頭させられるものなので、ユーザーへの配慮といったほかの考慮点はすべて圧倒されてしまうのだ。

プログラミングの呪縛から逃れてはじめて、わたしにもそれが見えるようになった。

これを発見して、ソフト製品がユーザーからみてこんなにひどい代物になってしまった原因となる影響がわかった。一九九五年に、そこで学んだことについて本を書いたが【編注1】、それは今日設計されているソフトに対して少なからぬ影響を与えている。

いいプログラマになるためには、コンピュータの性質やニーズをわかってやる必要がある。でも、コンピュータの性質やニーズというのは、それをいずれ使うことになる人間たちの性質やニーズとはまるっきりちがう。ソフトを書くのはあまりに知的な要求が高く、没頭する必要があるので、プログラマはまったく人間とは異なる思考プロセスに入り込んでしまうことになるのだ。

SAMPLE

プログラマの頭の中では、プログラミングのプロセスの要求は、ユーザーたちという外の世界からの要求をすべて上回るものになってしまうのだけれど、この両世界は、言語からしてお互いに相反するものになっている。

プログラミングのプロセスは、使いやすい製品をつくるというプロセスをひっくりかえしてしまう。それは、プログラマの目標とユーザーの目標が完全にちがっているという簡単な理由のせいだ。プログラマは、プログラムをつくるプロセスがスムーズで簡単であってほしい。ユーザーは、プログラムとのやりとりがスムーズで簡単であってほしい。この二つの目標は、ほぼ絶対に同じプログラムにはならない。

いまのコンピュータ産業では、ユーザーを喜ばせるような操作性をつくる責任はプログラマにあるけれど、利害関係がまったく対立しているために、プログラマたちはそれを達成できないのだ。

ソフトウェアでは、ふつうは仕上がりまで見えるものはない。つまり、プログラマでない人間があとから「それはちがうんじゃないか」と言っても、たいがいは遅すぎて役にたたないということだ。

デスクトップコンピュータのソフトは、悪名高いまでに使いにくい。それは、それが純粋にプログラマによる産物だからで、かれらとユーザーの仲立ちをする人はだれもいないからだ。

電話やカメラのようなものだと、必ずガチッとしたメカ部分があって、いやでも公開レビューにさらされる。でも、これまで見たとおり、コンピュータはほとんどどんな代物と組み合わせても、コンピュータのふるまいが圧倒的に強く出てしまうものなのだ。

SAMPLE

問題を解決する鍵となるのが、操作デザイン（interaction design）だ。ソフトウェアのふるまいを設計する、新しいプロの操作デザイナーという集団が必要だ。

いまでは、プログラマたちはプログラムの中の「コード」を意識的にデザインするけれど、それと人間とのやりとりについては、ほとんどうわの空でしかデザインしていない。それが何をするかという部分はデザインするけれど、そのふるまいや、やりとりや、情報提示の方法はデザインしていない。

これとは逆に、操作デザイナーたちは、ユーザーがソフト製品を見て操作するやりかたに直接注目する。この操作デザインという職能は、プログラマにとって目新しく、馴染みないものだから、そんなものは認めないか、認めたとしても、プログラミングが完全にすんだあとで話をきくだけだ。その時点では、もう手遅れだ。

ソフト製品の創造を仕切る人々は、技術的な知識がないためにプログラマたちの言いなりになっているか、あるいは自分たちもプログラマなので、プログラマ側の言い分ばかりきいているかのどちらかだ。ソフト製品を使う人たちは、そうした製品がほかの優れたデザインのツールを使うのと同じくらい、使い心地のいい、強力なものになれるのだということを、そもそも知らない。

プログラマたちは別に悪意があるわけじゃない。ソフトが使いやすくなるように、いっしょうけんめい働いている。ただ残念ながら、かれらの基準となる参照点は、自分たち自身なのだ。だからほかのソフトエンジニアにとって使いやすいようにはつくるけれど、ふつうの人間に使いやすくはしてくれないのだ。

SAMPLE

デザインのまずいソフトのコストは、計算できないほどのものだ。ジェーンとスンイルの時間コスト、腹をたてた飛行機の乗客たちのコスト、９６５便の乗客たちの命のコストは、そうそう簡単に定量化できない。でも、最大のコストはわれわれがふいにしている機会のコストだ。ソフトのおかげでいらだち、金を無駄にして、混乱させられ、むかつき、あげくに殺されている一方で、われわれはソフト製品の真価をモノにしていないのだ。

ソフトウェア製品は、これまで想像もつかないほどに人間的で、快適で強力な被創造物になれるはずなのだ。ソフトというのは、ほかのメディアをはるかにこえて、本当にどうにでもなるので、とんでもない夢想家の期待をすら上回るものになる可能性を秘めているのだ。そのために必要なのは、プログラミングと操作デザインをきっちり組み合わせるという、それだけのことなのだ。

【編注-1】アラン・クーパー『ユーザーインターフェイスデザイン』翔泳社、ISBN:4881353683

©2000 SHOEISHA