

Quantity Always Trumps Quality - 「量」はいつだって「質」に勝る

開始日	2008年08月09日
翻訳完了日	2008年08月09日
最終更新日(ちょこちょこ直したり)	2008年08月09日

はじめに

有名なブログ [Radium Software](#) の記事で、「[質より量に学ぶ](#)」というのがありました。リンク先を読んでもらえれば分かるのですが、とにかく今のボクにはハッとさせられる内容でした。

さて、その中に同記事の元ネタへのリンクが張られていました。

[Coding Horror - Quantity Always Trumps Quality](#)

自戒がてら、この記事の翻訳してみようと思います。

原著

「Quantity Always Trumps Quality」
<http://www.codinghorror.com/blog/archives/001160.html>

注意

- もともと個人利用を目的として日本語化したために、けっこう意識している部分があります。「意味分からないよ」とか「おかしいんじゃない?」とかいうのがあれば、コメントで指摘してもらえると嬉しいです(がんばって調べます)。

更新履歴

- 2008/08/09 作成開始 & 作成完了。考えたけれど、「量」というよりは「経験」と捉えた方がしっくり来るね。

訳文

August 02, 2008

Quantity Always Trumps Quality

「量」はいつだって「質」に勝る

[Nathan Bowers](#) pointed me to this [five year old Cool Tools entry](#) on the book [Art & Fear](#).

[Nathan Bowers](#) はボクにこの [5 年も昔に書かれた Cool Tools のエントリー](#) を紹介してくれたんだ。 [Art &](#)

[Fear](#)という本についての記事さ。

Although I am not at all ready to call software development "art" -- perhaps "craft" would be more appropriate, or "engineering" if you're feeling generous -- the parallels between some of the advice offered here and my experience writing software are profound.

ソフトウェア開発が「芸術」だなんてとても思えないけれど（多分「工芸」がより近いんじゃないかな。もしくは「エンジニアリング」とかならどうだい？）、いくつかの含蓄はボクのソフトウェア開発で得た経験を言い得ているんだよ。

The ceramics teacher announced on opening day that he was dividing the class into two groups. All those on the left side of the studio, he said, would be graded solely on the quantity of work they produced, all those on the right solely on its quality. His procedure was simple: on the final day of class he would bring in his bathroom scales and weigh the work of the "quantity" group: fifty pound of pots rated an "A", forty pounds a "B", and so on. Those being graded on "quality", however, needed to produce only one pot - albeit a perfect one - to get an "A".

Well, came grading time and a curious fact emerged: the works of highest quality were all produced by the group being graded for quantity. It seems that while the "quantity" group was busily churning out piles of work - and learning from their mistakes - the "quality" group had sat theorizing about perfection, and in the end had little more to show for their efforts than grandiose theories and a pile of dead clay.

陶芸の授業が始まると、教師はクラスを2つのグループに分けました。そして、工房の左側の生徒は作品を作成した「量」によってのみ評価し、右側の生徒は作品の「質」で評価すると伝えたのです。教師のやり方はいたってシンプルでした。

「量」グループの作品は、クラスの最終日にバスルームで重さを測り、50ポンドはA評価で40ポンドはB評価、と評価する。「質」グループは唯一の、ただし最高の1個を作れば良く、その質で採点する。

ところが、実際に評価の段階になると興味深い事実が浮かび上がりました。：もっとも高い質を持つ作品はすべて「量」グループによって作られたものだったのです。どうやら「量」グループは幾つもの作品をこねくりまわしているうちに間違いから色々学んでいたようで、一方「質」グループはその間完全性の理屈をこねくりまわし、けっきょく頭でっかちの理論と死んだ粘土の固まりしか生み出せなかったのです。

Where have I heard this before?

どこかで聞いたことのある話だよね？

- [Stop theorizing.](#)
- [Write lots of software.](#)
- [Learn from your mistakes.](#)

- [考えすぎるな](#)
- [とにかく沢山プログラムを書こう](#)
- [間違いから学ぶものさ](#)

Quantity always trumps quality. That's why the one bit of advice I always give aspiring bloggers is to pick a schedule and stick with it. It's the only advice that matters, because until you've mentally committed to doing it over and over, you will not improve. You can't.

「量」はいつだって「質」に勝るものさ。ボクは計画することと続けることをどん欲なブロガー達にいつもアドバイスとし続けているのはそういうわけなんだ。頭の中で考えているだけなら、いつまで

たっても成長しない。成長できないってことさ。

When it comes to software, the same rule applies. If you aren't building, you aren't learning. Rather than agonizing over whether you're building the right thing, just build it. And if that one doesn't work, keep building until you get one that does.

ソフトウェアについても同じルールが当てはまる。作らなければ、学べないんだよ。正しいものを作ろうと悩み続けるくらいなら、やってみればいいのか (Just build it)。もしうまくできなかったとしても、出来るまで作り続ければいい。

Posted by Jeff Atwood

(8111)

名前:	<input type="text"/>
コメント:	<input type="text"/>
<input type="submit" value="投稿"/>	