

局所最適化バイアス：今ここで 対 後にどこかで

私たち人間という動物は、自然と目先のことを優先的に考えます。今この場で起きていることに囚われてしまいがちです。人の生活がまだ単純だったころは、これでよかったでしょう。ただし、私たちがとる行動は、ときに普段あまり考えていないような結果を伴います。これは、時間（時間が経過した後に発生する場合）、あるいは距離（離れた場所で発生する場合）によるものです。つまり、私たちがとる行動が、思わぬときに思わぬところで、あまり関わりのない人やものごとに影響を及ぼすということです。

この盲点は、局所最適化バイアスと呼ばれることがあります。

例1

次の動画では、典型的な組織での局所最適化が見られます。管理者は、包装されていないチョコレートが隣のラインに混入しないように従業員に指示します。ほぼ問題なくできています。しかし長期的に、且つ全体的に見たら、局所的な最適化はビジネス・従業員・顧客全体に被害を及ぼしてしまいます。これは滑稽な動画のはずですが、あなたの会社では毎日のように似たようなことが起こっているの、真剣に受け止めてください。

例2

1980年代に、私はしばらくの間IT部門に配属されたことがあります。当時の会社のIT部門は、Information Services&Control (IS&C) と呼ばれていました。IS&C在籍中の私の仕事には、エンジニアリングチームが使用したいツールを購入できないようにするということが含まれていました。というのは、IS&Cは「ほぼ同じ」機能を持つわずかに安い代替品を大量購入することにより経費を節約できる、そして扱いにくいUnivacメインフレームの使用を強制させることにより、全てのデータをコントロールできると考えたからです。こうしたやり方は合理的で、場合によっては正しいこともあります。

私の上司は、彼自身の上司に対して自分たちがどれだけ会社の経費を節約したのかを示すレポートを作成しました。この上司のような人々は、自分たちは正しいことをやっていると思いながら定年退職したでしょう。しかし、経費をごくわずかに節約するという私の過去の行いが、長期的にはかえって害をなしていたということを、私は今になって悟りました。「安物買いの銭失い」は、局所最適化バイアスの一例です。

数字の読めないやつは話にならない。数字が見えない現場もいかん。だけどな、数字しか見ないやつが一番いかん。

-大野耐一

例3



初心者のスクラムマスター（本の著者を含む）は、製品開発組織の教育を改善したり、チームの適応力向上を支援したりするのではなく、単に超生産的(hyperproductive)なチームを作ること躍起になっています。

ソフトウェア開発について深く理解していない人は、大規模なアジリティは複数の「超生産的」なチームの作業の総計だと考えるかもしれません。一見、論理的に思えるかもしれませんが。馬の足が早ければ早いほど、馬車も速く進むと考えませんか？しかし、賢明な読者のみなさんは、ソフトウェア開発は馬車を引っ張る馬とはわけが違うことは理解しているはずです。全体像を見ると、超生産性を目指そうとする人はだれもかれも、より速い会社ではなく、とてつもない混乱を創り上げるだけです。これでは競合他社に勝つことはできません。局所的な最適化しか行わないスクラムマスターは、まさに大規模なアジリティの大敵です。

もし局所最適化の例がありましたら、ぜひご連絡ください。

らせん状の通路の中、幾重にも重なり合う輪
その模様は壮大で複雑
何度も何度も僕らは途中で視界を失い
原因は思わぬ結果を生み出す

ーラッシュ

English version (<https://seattlescrum.com/local-optimization-bias/>)