

The Learning Edge

Getting into the Flow of software development.

In his ground-breaking book, subtitled *The Psychology of Optimal Experience* Mihaly Csikszentmihalyi of the University of Chicago described the concept of “Flow” [1]. Flow, as described by the professor, is a state of altered consciousness in which our ability to concentrate and perform is enormously enhanced. People who achieve this state also report a tremendous increase in their sense of achievement and satisfaction. There are many disciplines in which Flow may be exhibited. Athletes participating in professional sports sometimes display it, demonstrating astounding feats of physical ability when they do. One only had to watch Michael Jordan when he was on form (anytime between 1987 and 1998) to see the results of this kind of concentration. But the ability to attain this highly effective and rewarding condition is not limited to those engaged in sports. In fact, the good professor described many examples of very different people

engaged in very different activities, from parents playing with their children to an ancient Chinese cook chopping up an ox, who attained this state.

ANXIETY AND BOREDOM

In our professional and personal lives, we sometimes navigate between two opposing conditions. When we tackle routine tasks that we can easily do, we find our minds wandering and we may become bored with the monotony of doing the same old thing. At the other end of the difficulty spectrum, when we tackle brand-new tasks where our competence is low, we may feel overwhelmed and out of our league. This is particularly true if there is signif-

icant pressure to perform well and to be “successful” and, at the same time, our lack of ability and experience is made public.

THE COMPETENCY ZONE

Between these two states lies the “Competency Zone” (see Figure 1). This is the area where our abilities and the demands of the task are in sync. The task is not so routine and mundane that it does not require effort and attention, and it is not so difficult that it induces anxiety. Too far in the boredom direction and we tend to become less effective, simply because the task is boring and our attention wanders. Too far into the anxiety region and we become less effective due to the drain on energy generated by our unease at our inability to perform. Csikszentmihalyi pointed out

The Business of Software

Figure 1. The Competency Zone (adapted from [1]).

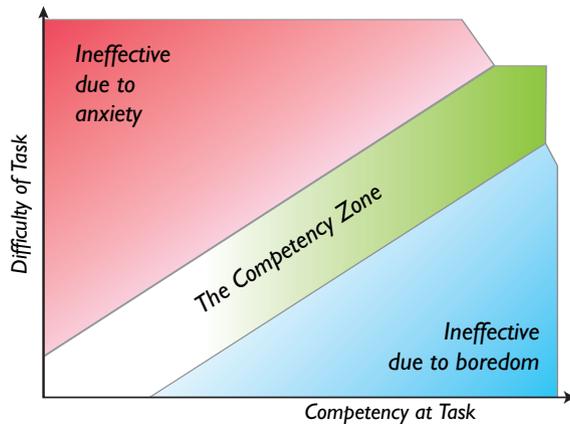
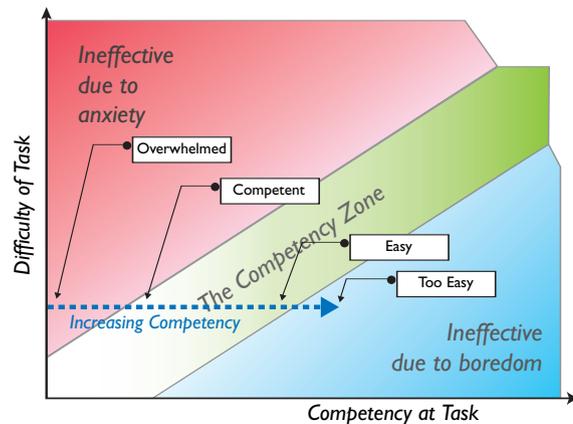


Figure 2. Increasing competency.



that we tend to arrange our work and social activities to remain within the Competency Zone. Of course, we do not stay there all the time, or we would never learn anything new. Tackling any new activity will tend to position us more toward the anxiety region. Too far into that region and people may shut down and move away from the psychic, or even physical, pain caused by the lack of ability. Assuming we stick to the task, and begin to learn, our competency increases (see Figure 2). We become more competent until we are able to successfully complete the task. This is often the point where people experience the greatest sense of reward as they flex and explore their new skill. Continuing to apply the skill makes the task easier and easier, until finally it becomes boring and learning stops.

For the task to continue to be rewarding, we must increase its difficulty. Doing so too much, of course, may put us back into the anxiety region. Music teachers are

very familiar with this challenge; music students often “plateau” in their skills where they become increasingly capable at their current level of practice. The challenge for a good teacher is to increase the difficulty to help the student transition to the next level without making the task so difficult that the student retreats from the challenge.

SOFTWARE FLOW CHART

Software development, being primarily a learning activity, is subject to the laws of Flow. Tackling tasks that are way beyond our competencies tends to be anxiety-inducing and ineffective for both individuals and organizations. Our lack of expertise in the subject, and our apprehension over performing inadequately, combine to generate lower performance and more defects. The lower quality and the additional effort we have to throw at the task to compensate for our lack of experience also tend to increase our anxiety level, which further compounds the problem.

However, if we play it safe and operate only at a comfortable level of competency it means that we, or our company, are only doing things we know well. It also means we are probably doing what other companies can also do well. This generates serious competitive and price pressure and the business will usually go to the lowest bidder.

At its core, the purpose of a software project is always to do something new. “New” implies learning, and learning implies that we are not quite operating within the competency zone. This can be a problem for some individuals and some companies.

THE COMFORT ZONE

Most of us are most comfortable in the “Comfort Zone,” which occupies the lower portion of the Competency Zone. It is in this area that things are easiest for us—we can effortlessly show our skills and flawlessly produce our products. It is called the Comfort Zone because it is the low energy state; it is the area where we are

most effective at the lowest output of effort. Any further into the boredom region and tedium will likely overwhelm our mindless effectiveness and automatic competence.

While the Comfort Zone is the low energy state and is the “easiest,” it is not the most rewarding. Csikszentmihalyi found that many of us spend

opment were entirely the application of existing knowledge it would be a manufacturing activity and we would automate it.

Another name for the discovery of knowledge and the reduction of ignorance is “learning.” The Comfort Zone is not very rewarding in the business of software because in this zone we don’t learn much. When learning

always seek to learn new ways of doing things, even ones that they can already do. Most of us, once competent, will seek to fine-tune that competence by learning ways to optimize our performance. But this has more to do with reducing effort than it has to do with increasing learning—there just isn’t the same pressure to innovate that

If software development were entirely the application of existing knowledge it would be a manufacturing activity and we would automate it.

large quantities of time in this zone, but we don’t find it rewarding at all. Watching television from the safety of the couch might qualify under this category and, perhaps surprisingly, we seem to spend a lot of our life doing such low-value things.

Operating within the Comfort Zone while we are building software allows us to easily display competence and easily build systems. But it may also be unrewarding and ineffective. The reason is that at the low end of the Competency Zone we don’t learn much. While the business of software does involve some translation of what we already know into an executable form, it also involves the discovery of what we don’t know. The application of our present knowledge does not usually involve much effort, especially when compared to the resolution of our ignorance. In fact, if software devel-

is the primary business function, avoiding learning is a very negative thing to do. Arguably, one of the key jobs of businesses engaged in creating software should be to optimize the rate of learning. But this operates somewhat against human nature which, like most dynamical systems, may tend to move toward the low energy state of the Comfort Zone.

THE LEARNING EDGE

At the other side of the Competency Zone lies the Learning Edge. The Learning Edge is just *outside* the main part of the Competency Zone. If we are truly operating within our range of competencies we don’t tend to learn much—we don’t need to because we are already sufficiently capable. Of course, there are self-actualized individuals who will be perennially dissatisfied with the status quo and will

exists when it is obvious that we cannot do our job at our current level of knowledge.

Also, our learning within the Competency Zone tends to be incremental, interpolating what we already know and marginally enhancing our existing skills. There are few among us who will voluntarily replace entire bodies of knowledge and skills with completely new capabilities and concepts unless we are absolutely forced to. This is what Thomas Kuhn called the “paradigm shift.” The history of science is dotted with instances where very, very smart people simply couldn’t replace the existing mental models that had served them so well, even when the scientific evidence clearly showed the limitations of that knowledge [2].

Finally, we could assert that new knowledge is the only true differentiator in software development, since new functionality

The ability of a person or an organization to get into and stay at the Learning Edge and in the state of Flow is critical.

is the primary reason for running a project in the first place. Even if it is not some ground-breaking revolution in software thought, each system is (or should be) at least *some* new assembly of existing knowledge. Our focus and efficiency at acquiring this new knowledge is what we call “productivity” and it is highly sensitive to our position with respect to the Competency Zone. Therefore, the ability of a person or an organization to get into and stay at the Learning Edge and in the state of Flow is critical.

ACCELERATED LEARNING

Accelerated learning occurs on the outside edge of the Learning Edge (see Figure 3). This is where the most is learned most effectively in the least time.

The trick with managing accelerated learning is getting people as far as possible into the anxiety region *without them shutting down*. Some organizations are very good at doing this. The U.S. Marine Corps brings its recruits into the Parris Island and San Diego depots at 2:30 in the morning for a reason. They intentionally stress the recruits and push them way into the anxiety region because it is there that they learn the most. Specifi-

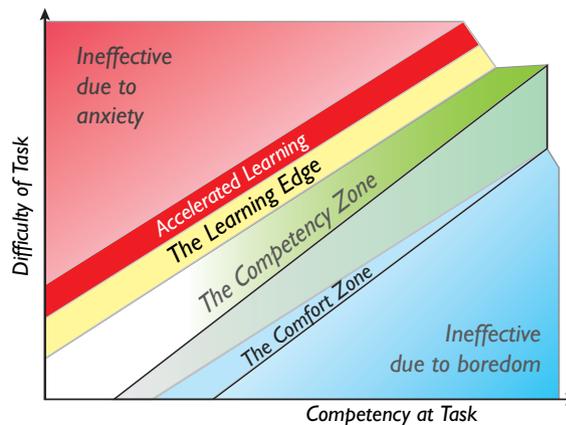


Figure 3. The Comfort Zone and the Learning Edge.

cally, it is there that they learn how to be U.S. Marines. The Outward Bound school¹ uses a similar, albeit less extreme, approach when it takes at-risk teenagers out into the backwoods and wilderness and places them in a challenging and stressful environment where they can learn the self-respect and self-confidence they often lack. Both organizations freely acknowledge that learning, particularly accelerated learning, is not a “comfortable” activity.

Given the potential for the enormous increases in productivity that have been shown when people and groups attain the state of Flow, it would seem this

should be a primary goal for any organization engaged in the business of software.

It is interesting that few companies seem to think this way and seem to be willing to step up to the challenge of truly generating and perpetuating a sense of Flow in their projects and with their people. It may be that, as companies, they simply prefer to flop down in the business

Barcalounger or vegetate in the corporate Comfort Zone. If so, this is sad, because such companies deprive themselves of discovery and innovation, productivity and profits, and truly satisfied and fulfilled employees. And, according to Csikszentmihalyi, they also deprive themselves of meaning and purpose. ■

REFERENCES

1. Csikszentmihalyi, M. *Flow: The Psychology of Optimal Experience*. Harper and Row, 1990.
2. Kuhn, T. *The Structure of Scientific Revolutions*. University of Chicago Press, 1962, 1970.

PHILLIP G. ARMOUR (armour@corvusintl.com) is a senior consultant at Corvus International Inc., Deer Park, IL, and a research director in the Center for Software Development Innovation at Number Six Software, Vienna, VA.

¹Outward Bound is a non-profit organization that provides intense learning experiences for teenagers and adults (see www.outwardbound.com); of course, the U.S. Marine Corps does the same.