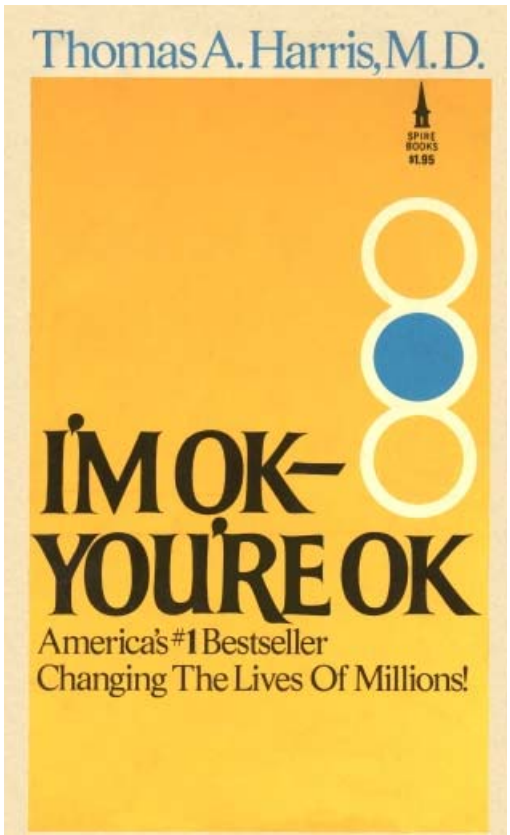
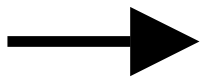
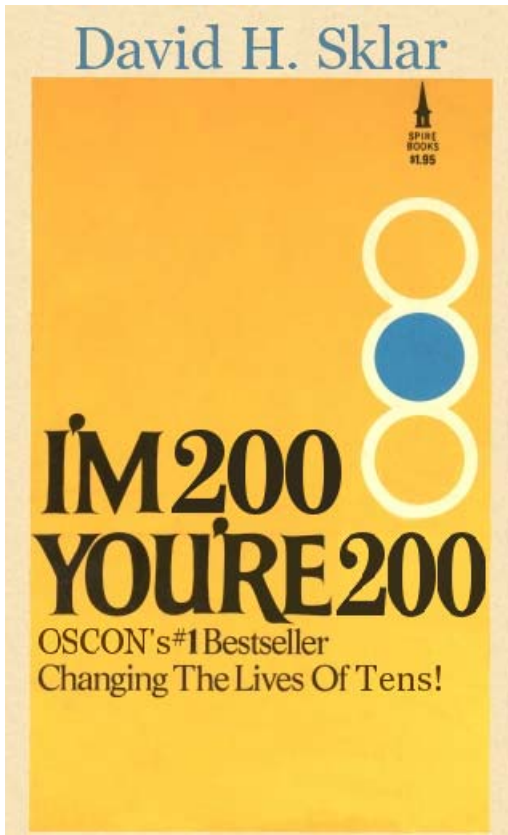






?

David Sklar • http://www.sklar.com/blog
sklar@sklar.com • david@ning.com

W** 2.*

- Lots of software these days is
  - Always on
  - Depends on “remote” resources
  - Has parts that are out of the direct control of the primary creator
- This is new-ish
- …or is it?

This means...

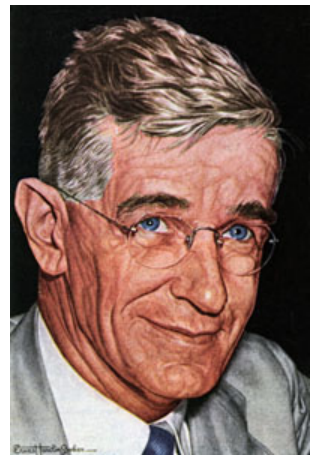
Dependency · Risk · Lock-In · Uptime · Outage · Integration · APIs · Timeouts · Injection Attacks · Service Guarantees · Use Restrictions · Non-Portable Data ·

Today's Topic?

God, give us grace to accept with serenity the things that cannot be changed, courage to change the things which should be changed, and the wisdom to distinguish the one from the other.
(so says Reinhold Niebuhr)

Picture from http://www.niebuhrsociety.org/

Today's Topic, cont'd.



WOPR, give us grace to accept with serenity the **dependency mitigation strategies** that cannot be **implemented**, courage to **implement** the **strategies** which should be **implemented**, and the wisdom to distinguish the one from the other.

Pic from http://freemasonry.bcy.ca/biography/bush_v/bush_v.html

Another Interpretation

It’s all tubes and they're playing a game of pick-up sticks.

More on Ted Stevens:
http://blog.wired.com/27BStroke6/?entry_id=1512499

Pic from http://stevens.senate.gov/about.cfm

No, Seriously

- What is your app dependent on?
- What happens to the app when one of those dependencies has a problem?
- Today's focus is primarily on web service **consumers** rather than **producers**.

“Dependency” = Not Yours

- Physical Control
- Organizational Control
- Intellectual Control

What are your dependencies?

Server dependencies:

- content created by others
- real-time external network calls
- internal network calls
- per-machine software
- per-machine hardware
- power + networking

What are your dependencies?

Code dependencies:

- who wrote the code?
- who knows how the code works?
- who wrote the documentation?
- who knows the places in which the documentation is wrong?
- who knows how to hire more people who know what you need them to know?

What are your dependencies?

Business dependencies:

- who supplies your data feeds?
- how many of your co-workers are in the National Guard?
- what are your copyright and patent risks?
- what do your SLAs say?
- are the penalties in the SLA really helpful?

Uptime is Not Binary

- Mitigate dependencies with modes of degradation based on:
  - Data freshness
  - App Features
  - Read-only/read-write data

Data Freshness

- Avoid Live WS Calls when possible
- Make calls “offline”, sanity-check results, and cache
- Build local data store (when possible) for when remote service has blips

Features

- Segmenting your app into potentially non-interdependent parts
- Can a detail page go without its map?
- Can you turn off an ad network?

Read/Write

- Perhaps just for internal issues, but:
  - provide “static” / read-only access to app while preventing data-changing operations
  - feature segmentation on personalization

Provider Switch

- Use ad network G instead of Y
- Use map provider Y instead of G
- Lack of common APIs can require custom development...look for the rise of meta-API standards
- APP, S3, “ad HTML” already filling that need?

Monitoring

- How do you know when something needs to change?
- Automatic degradation
- Alerts based on timing + response content with manual switchover

It all comes back to...balance

- In the end, everything really is insurance.

- *mitigation cost* ≤ (*occurrence probability* × *occurrence cost*)
- *cost* = *money* + *goodwill*
- What are the **likely** issues you will run into?

Are you 200?

- What **are** your dependencies? (Think of any new ones since the talk started?)
- What outages have you been hit with? How did that go? Tips for others?

Cute Picture of a Kitten

Pic: http://cuteoverload.ning.com/largePicture.php?DetailID=1357475