Low-Context DevOps:
A new way of improving SRE team culture

Thomas A. Limoncelli
SRE Manager

stackoverflow.com/teams
Introduction
Who is Tom Limoncelli?

- Pronouns: he/him
- SRE Manager at Stack Overflow, Inc (NYC)
- Sysadmin since 1988 (Google, AT&T/Bell Labs, Mentor)
- Blog: EverythingSysadmin.com  Twitter: @YesThatTom
- Author
10 billion
Page views in 2019

100+ million
Unique visitors in 2019

stack overflow
Foreshadowing
1. High and low context cultures

2. Low context DevOps

3. Leadership
Our common challenge
1. High and Low Context Cultures
High Context Cultures

A party with friends, family gatherings, expensive gourmet restaurants with a regular clientele, undergraduate on-campus friendships, hosting a friend in your home overnight

- Communication is implicit. Less written/formal information; more collective history
- People have to “read between the lines” to understand what’s going on
- Relies on long term relationships
- Decisions and activities focus around personal face-to-face relationships, often around a central person who has authority
Low Context Cultures

Large airports, a chain supermarket, a cafeteria, a convenience store, sports where rules are clearly laid out, a motel

- Communication is explicit
- There are rules, you are told the rules
- Knowledge tends to be codified, public, external, and accessible
- More interpersonal connections of shorter duration
- Knowledge is more often transferable

http://www.culture-at-work.com/highlow.html
High Context
All meaning is not conveyed in the language

Low Context
Most meaning is expressed by the specific words

Chinese languages
Japanese
Arabic
Indian languages
Greek
Spanish
Italian
English
French
American
Scandinavian languages
German/Swiss
Dutch

https://www.slideshare.net/trompenaars/tht-iap-certification-presentation-day1-6dec2010
2. Low Context DevOps
A DevOps environment should strive to be low-context
Three ways to reduce the required context of your DevOps env

- Carefully Constructed Defaults
- Make Right Easy
- Ubiquitous Documentation
New Employee Environment

- PC
- Software required to do your job
- Access and Permissions
Why this stays broken?

- New employees can't fix it.
- Experienced employees don't feel the pain.
- Requires working across silos: IT, InfoSec, Eng, HR
- Who will do this if you don't?
Make Right
Easy
The lazy path guides you to the right way
Good

LibreSSL:
The default is “timelessly correct”

Our CI/CD pipeline embodies our recommended practices

Base library institutes telemetry collection, standard flags, etc. Not being standard is difficult!

Bad

OpenSSL:
Requires PhD to use right, stale settings
Foundational Tools

Your foundational tools and infrastructure make or break your ability to provide a low context environment.

- Ticket system
- Bug tracking system
- Monitoring / observability
- Config management
- OS Installation and Patching
- CI/CD Pipeline system
- Container / artifact repository
- Documentation repository
- Source code control
- Chat & Collaboration Infrastructure
Ubiquitous Documentation
Docs when you need them

A deep link / URL:

- In error messages
- In CI/CD control panel descriptions
- In alert messages
- Everywhere people might want info!
Last login: Fri Feb 14 07:53:59 on ttys004

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208650.
tlimwork2:~ tlimonce11$  

Create a culture of ubiquitous documentation
Management sets expectations

- It’s not done until it’s documented
- Not just big projects: Tickets, chatops, releases

File bugs about docs just like software

- Record tech debt or it won’t be fixed
- Stale documentation is dangerous

Culture of always updating as you work

- “Manual Work is a Bug”
- https://queue.acm.org/detail.cfm?id=3197520

Fight the “my code is the documentation”

- At a minimum, you need a doc to point to the code! (or how to get started)
Need inspiration?
Document so you can relax while on vacation.
Docs?

What docs?
404 Not Found
Raise your hand if you love to write documentation
Why people avoid writing documentation
Uncertain Scope

Uncertain Audience

And ...
Blank VIM document
Blank dark mode
Blank screen syndrome
“While not a recognized medical condition, Blank Screen Syndrome is a very real problem.”
Really awesome templates

**Service Doc Template:**
- Overview
- Useful links
- High-level diagram
- History/Background
- Desired State
- Common Issues

**Alert Doc Template:**
- Failure condition
- Business impact
- Escalation chain
- Suggested resolution (checklist)

The Practice of System and Network Administration, 3rd Edition, Chapter 31

The Practice of Cloud System Administration, Chapter 17
Write in small batches!
Include doc updates in work estimates
Where do engineers already write?

When someone writes a great description of something, tell them to paste it into something more permanent!

- Email
- Chat
- Instant message
How do I exit the Vim editor?

Answered 7 years, 8 months ago  Active 12 days ago  Viewed 2.1m times

I'm stuck and cannot escape. It says:

```
3695 type :quit<Enter> to quit VIM
```

But when I type that it simply appears in the object body.

```
901
```

- Share
- Edit
- Follow
- Tag

37 Answers

Hit the (Esc) key to enter "Normal mode". Then you can type :q to enter "Command-line mode". A colon (:) will appear at the bottom of the screen and you can type in one of the following commands. To execute a command, press the Enter key.

- :q to quit (short for quit)
- :q! to quit without saving (short for :quit!)
- :wq to write and quit
- :wq! to write and quit even if file has only read permission (if file does not have write permission: force write)
- :wx to write and quit (similar to :wq, but only write if there are changes)
- :exit to write and exit (same as :wx)
- :qa to quit all (short for :quitall)
- :q: to quit without saving and make Vim return non-zero error (i.e. exit with error)
Foreshadowing
New employees can use it.

 Experienced employees feel/see the pain.

 Works across silos: IT, InfoSec, Eng, Marketing
3. Leadership
But who will make it happen?
But who will make it happen?
Management

vs.

Leadership
Management is:

- Set priorities
- Provide resources
- Clear roadblocks
Leadership is:

- Going first
- Making it easy for others to follow
In summary...

**DevOps environments should strive to be low context**

**Tip #1: Smart Defaults**
- New Employee Env: Accelerate new hires being productive

**Tip #2: Make Right Easy**
- The lazy path should fulfill recommended practices
- Foundational Tools (CI/CD, Git, etc.)

**Tip #3: Ubiquitous Documentation**
- A culture of documentation
- Management support
- Small batches: Write as you work, not after
- Inspiration: Helps the team; reduces the likelihood you'll get paged while on vacation
In summary...

**Why we don’t write docs?**

- The burden of “audience” and “scope”
- Scary blank screen syndrome

**How to fix:**

- Templates, templates, templates
- Repurpose text from where they already write (Email, Chat rooms, IM, StackOverflow)
- Central source: [stackoverflow.com/teams](http://stackoverflow.com/teams)
Thomas A. Limoncelli
SRE Manager

Twitter: YesThatTom
Blog: EverythingSysadmin.com

stackoverflow.com/teams
Thomas A. Limoncelli
SRE Manager

Twitter: YesThatTom
Blog: EverythingSysadmin.com

stackoverflow.com/teams