

# The Finer Art of Being a Senior Sysadmin



<http://bit.ly/seniorSA>

**Sheeri Cabral**

Senior DB Admin/Architect, Mozilla  
@sheeri [www.sheeri.com](http://www.sheeri.com)



**It should not be a length of time**



**Experience is a requirement**



**The Battle Against Any Guess**  
**[www.battleagainstanypguess.com](http://www.battleagainstanypguess.com)**



Which brings us to Tom Limoncelli...



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
understands the internal workings  
of the systems he/she administers**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
understands the internal workings  
of the systems he/she administers  
and debugs issues from a place of science,**





<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
understands the internal workings  
of the systems he/she administers  
and debugs issues from a place of science,  
not guessing or rote memorization.**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
understands the internal workings  
of the systems he/she administers  
and debugs issues from a place of science,  
not guessing or rote memorization.**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person has enough experience  
to know a problem's solution**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person has enough experience  
to know a problem's solution  
because he or she has seen and fixed it before



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person has enough experience  
to know a problem's solution  
because he or she has seen and fixed it before  
(but is smart enough to check that assumption



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person has enough experience to know a problem's solution because he or she has seen and fixed it before (but is smart enough to check that assumption since superficial symptoms can be deceiving).



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
automates their way out of problems**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**A senior person  
automates their way out of problems  
rather than "working harder".**





<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person  
automates their way out of problems  
rather than "working harder".  
They automate themselves  
out of a job constantly



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person  
automates their way out of problems  
rather than "working harder".  
They automate themselves  
out of a job constantly  
so they can be re-assigned  
to more interesting projects.



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person  
automates their way out of problems  
rather than "working harder".  
They ~~automate~~ themselves  
out of a job constantly  
so they can be re-assigned  
to more interesting projects.



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

A senior person  
automates their way out of problems  
rather than "working harder".  
They **document** themselves  
out of a job constantly  
so they can be re-assigned  
to more interesting projects.



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**By creating processes others can follow**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**By creating processes others can follow  
you enable delegation**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**By creating processes others can follow  
you enable delegation  
and multiply effectiveness**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**What's better?**





<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

**What's better?  
Knowing a lot?**



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

What's better?  
Knowing a lot?  
Or being good yourself



<http://everythingsysadmin.com/2012/09/seniorsysadmins.html>

What's better?  
Knowing a lot?  
Or being good yourself  
and helping others learn too?



But how?



wikis

But how?



wikis

checklists

But how?



[www.checkmarkable.com](http://www.checkmarkable.com)



wikis

checklists

But how?

blog posts





wikis

checklists

But how?

blog posts

speaking



wikis

checklists

podcasts

But how?

blog posts

speaking



wikis

checklists

podcasts

webinars

But how?

blog posts

speaking



wikis

checklists

podcasts

webinars

But how?

blog posts

speaking

writing a book



wikis

checklists

podcasts

webinars

But how?

blog posts

speaking

writing a book

teaching



# Ops School

<https://ops-school.readthedocs.org/en/latest>



# Ops School

<https://ops-school.readthedocs.org/en/latest>

**Virtual Self-Study Groups:  
PHP Percolate**



# Ops School

<https://ops-school.readthedocs.org/en/latest>

Virtual Self-Study Groups:  
PHP Percolate  
HTML5 Brunch





# Ops School

<https://ops-school.readthedocs.org/en/latest>

Virtual Self-Study Groups:

PHP Percolate

HTML5 Brunch

jQuery Jam



# Ops School

<https://ops-school.readthedocs.org/en/latest>

Virtual Self-Study Groups:

PHP Percolate

HTML5 Brunch

jQuery Jam

MySQL Marinate



# Ops School

<https://ops-school.readthedocs.org/en/latest>

Virtual Self-Study Groups:

PHP Percolate

HTML5 Brunch

jQuery Jam

MySQL Marinate

[www.mysqlmarinate.com](http://www.mysqlmarinate.com)



wikis

checklists

podcasts

webinars

**Keep learning from others!**

blog posts

speaking

writing a book

teaching



**Level up!**



**Let's turn to John Allspaw**



<http://www.kitchensoap.com/2012/10/25/on-being-a-senior-engineer/>

**Experience aka “Engineering maturity”**



Can there be engineering maturity in DevOps?





<http://www.kitchensoap.com/2012/10/25/on-being-a-senior-engineer/>

# **“The Unwritten Laws of Engineering” written in 1944**

*The Unwritten*  
**LAWS OF ENGINEERING**



with revisions and additions  
by  
**James G. Skakoon**

and original  
by  
**W. J. King**



**USD**  
**\$14.95**



*The Unwritten*  
**LAWS OF ENGINEERING**



with revisions and additions  
by  
**James G. Skakoon**

and original  
by  
**W. J. King**



**USD**  
**\$14.95**

**69**  
**pages**

*The Unwritten*  
**LAWS OF ENGINEERING**

with revisions and additions  
by  
**James G. Skakoon**

and original  
by  
**W. J. King**





**Mature engineers  
seek out constructive criticism  
of their designs.**



It's nice to have people say  
“yes” and “you're a genius!”



It's nice to have people say  
“yes” and “you're a genius!”  
but it's not optimal.



**Mature engineers  
understand the non-technical areas  
of how they are perceived.**





It does not matter how good you are



It does not matter how good you are  
if nobody wants to work with you



**Level up  
on communication**



Critique tech  
not people



**Mature engineers understand  
that not all of their projects  
are filled with rockstar-on-stage work.**



Having the power to accept commits  
is like being a janitor



**See also: documentation**



**Mature engineers make their trade-offs explicit when making judgements and decisions.**





That means knowing what your  
tradeoffs are



**Mature engineers know  
the importance of  
feelings people have.**



**Mature engineers know  
the importance of  
(sometimes irrational)  
feelings people have.**



<http://www.kitchensoap.com/2012/10/25/on-being-a-senior-engineer/>

**Allspaw's article is worth reading:  
It ends with The Ten Commandments of  
Egoless Programming**



# The Peter Principle and the myth of the corporate ladder



**Do not contribute to the problem!**



Feedback?