

How To Build And Operate An MQ Center of Mediocrity (MQCOM)

Because excellence is hard work!

Change is the only constant

Best Practices always evolve over time. That's what keeps them "best."

This presentation reflects...

- My current opinions regarding MQ (Less Than) Best Practices
- The product itself continues to evolve (even in PTFs)
- This version of the presentation is based on WebSphere MQ v7.5 and up
- This content will be revised over time so please be sure to check for the latest version at <https://t-rob.net/links>
- Your thoughts and ideas are welcome

Meanwhile, mediocrity clings on with the tenacity of an alien face hugger.



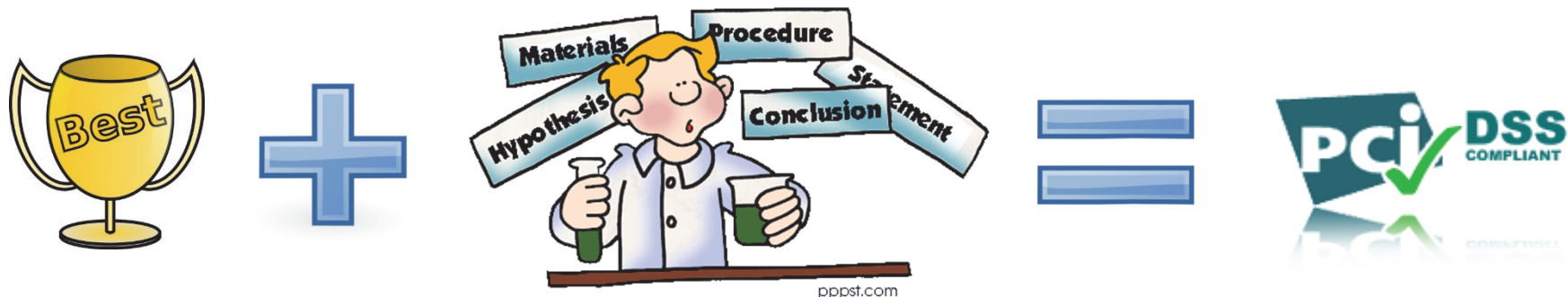
Agenda

- Excellence
- **Reverse-engineering excellence**
- Mediocrity as a management strategy
- Gaming the 80/20 rule
- Some example MQ Best, Second Best, and Lesser Practices
- Actively suppressing spontaneous excellence
- Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)

Analyzing excellence

There's a methodology for that!
(Coming soon to an app store near you?)

1. Identify the most successful shops.
2. Identify common processes and configuration patterns.
3. Distill these into a document.
4. Call it Maturity Model or a standard and sell services around it!



Case study in reproducing excellence

After extensive study of many excellent human specimens, all were found to have several traits in common:

- Head
- Well muscled core
- Strong arms
- Hands
- Strong legs
- Feet



Note: Some variation was observed in the in limbs, hands, and feet of study subjects. The head and torso appear to be mandatory in all cases, however. Start with those.

Achieving the same result

According to the theory, we can achieve excellence by implementing the same features that were common to the exemplars.

- ☒ Head
- ☒ Well muscled core
- ☒ Strong arms
- ☒ Hands
- ☒ Strong legs
- ☒ Feet

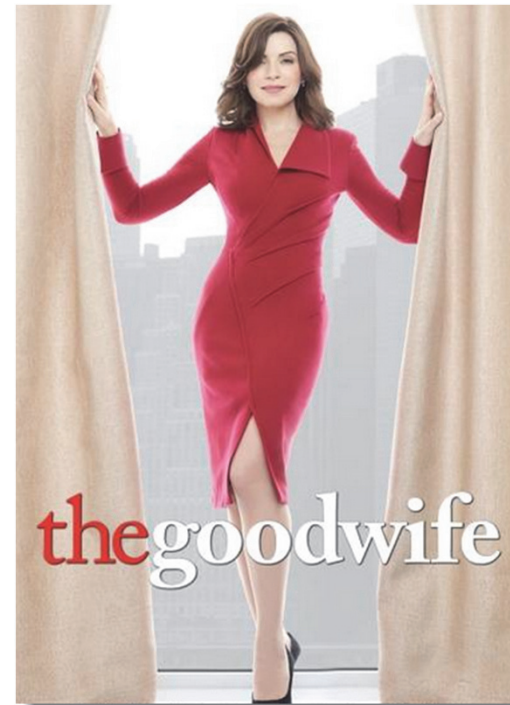


Mission accomplished!

Case study in reproducing mediocrity

The guidelines for running a Center of Mediocrity are determined using the same methodology but based on study subjects that have not risen to a level that can be rightfully called “excellent.” Here we focus on the merely “good.”

- Head
- Torso
- Arms
- Hands
- Legs
- Feet



Achieving the same result

According to the theory, we can achieve *mediocrity* by implementing the same features that were common to the exemplars, but without the up-front cost!

- ☒ Head
- ☒ Torso
- ☒ Arms
- ☒ Hands
- ☒ Legs
- ☒ Feet



Mission accomplished!

Our takeaway?

Excellence is a process. It takes stamina.

Mediocrity is a purchase. Anyone can do it!

*(*And later we show you how to hide the line item cost!)*

Mediocrity ==



Excellence ==



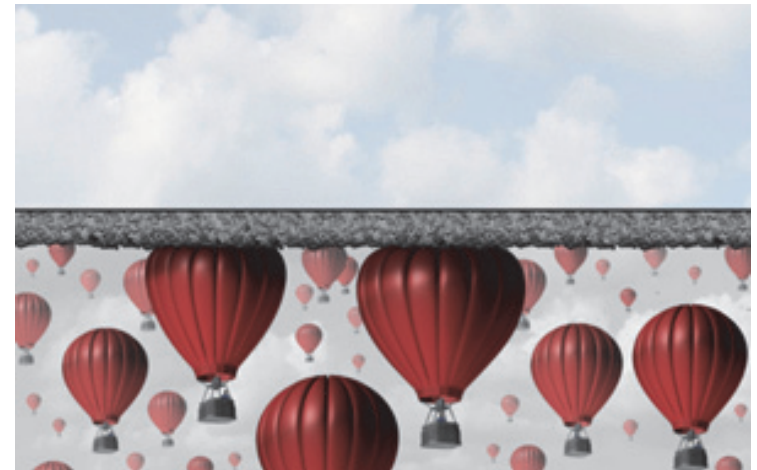
Agenda

- Reverse-engineering excellence
- **Mediocrity as a management strategy**
- Gaming the 80/20 rule
- Some example MQ Best, Second Best, and Lesser Practices
- Actively suppressing spontaneous excellence
- Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)

The excellence ceiling

Mediocrity in pursuit of excellence.

- Premise: Excellence is measurable.
- Premise: KPI's are universally applicable.
- Premise: The most important KPI is cost.
- Premise: The more we measure, the better.
- Premise: The more precisely we measure, the better
- Premise: That which is not easily quantifiable is not important.
- Conclusion (whether acknowledged or not): Quantity == Quality



Investing only in things where the benefits are easily quantified means giving up on results that are not easily measured, such as synergy or productivity.

Examples

Recognize any of these cases in which hard accountable cost is avoided at the expense of unaccountable and much greater soft costs?

- Lack of tools: cost realized as massive administrative burden.
- Lack of tools: admin-heavy tasks (i.e. cert management) infeasible.
- Lack of training: cost realized as extended and/or frequent outages, sluggish turnaround of support and project tasks.
- Infrequent patch management: costs deferred as technical debt with a steep vig and that comes payable mainly during Production outages.
- Security disabled in non-Prod environments: assures that any security defect is customer-facing, reputation-risking, cash hemorrhaging, and potentially business-ending. Assures security defects occur by not implementing controls prior to Production release.

Mediocrity is not cheap

Note: After the up-front implementation cost, the day-to-day operation cost of a Center of Mediocrity nets *higher* than that of a Center of Excellence.

Strategy: As long as it's not in your department it's OK.

- Push that elevated cost to another department as administrative overhead.
- Distribute it as an incrementally small burden on each of very many people.
- Declare things like patch management or security as unnecessary.



Agenda

- Reverse-engineering excellence
- Mediocrity as a management strategy
- **Gaming the 80/20 rule**
- Some example MQ Best, Second Best, and Lesser Practices
- Actively suppressing spontaneous excellence
- Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)

Mediocrity master class

**Premise: 80% of the benefit comes
from 20% of the investment.**

**Hmmm...if that's true, we
can get 64% of the benefit
from only 10% of the
investment.**

Let's do that!





*A little mediocrity
goes a long way.*

*Once you commit to it,
less is more.*

* Also, if you put stuff in *Script* it seems more authoritative and mediocrity is about nothing if not appearances.

Agenda

- Reverse-engineering excellence
- Mediocrity as a management strategy
- Gaming the 80/20 rule
- **Some example MQ Best, Second Best, and Lesser Practices**
- Actively suppressing spontaneous excellence
- Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)

Pro Tip:



As we go through this next section, look around from time to time to see if anyone is looking horrified and shrinking down in their seat because they just recognized something they do.

If you do not see anyone looking horrified and shrinking down in their seat, it's probably your turn.

Guide to mediocrity



- **Best:**

- **Best:**

- **Best:**

- **Second Best:**

- **Second Best:**

- **Second Best:**

- **Lesser:**

- **Lesser:**

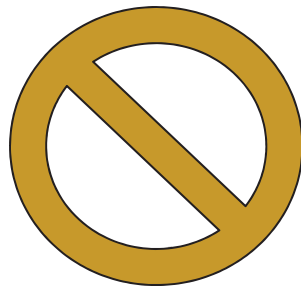
- **Lesser:**



No!

As we study the examples, keep in mind that you are aiming low.

If you shoot for the middle, you might accidentally overshoot and achieve excellence by accident.



Maybe

That would set a dangerous precedent so, in the immortal words of Larry David, curb your enthusiasm.



You want to be here

Patch Management



- **Best:** We have one or two people whose job includes subscribing to IBM announcements for patch management, security, new releases, etc.
- **Best:** We are making plans now to migrate from v7.5 to v8.0 or v9.0
- **Best:** We have capacity and a process to apply a new patch within [x] days.
- **Second Best:** We plan to buy extended support.
- **Second Best:** We plan to patch everything on Extended Support to the latest or second-latest patch so we can actually *get* that support.
- **Lesser:** Wait, what? V6.0 is out of support?
- **Lesser:** If it aint broke, don't patch it.
Alternative: If it aint Broke, don't patch it.



Patch Management



- One customer paying extended support on about 3k MQ servers was ineligible for support on about 2.5k of them due to patches 3 or more releases out of date.
- Multiple shops have failed compliance audits due to unpatched MQ. In at least one case I know of, the customer was given 90 days to patch all of Prod or lose their payment card processing privileges.
- More than one shop's idea of client patch management was to copy individual jar files from the new release. In one case we found jars from at least 2 major versions of MQ and 5 different releases. They had been having frequent outages in customer-facing systems and taken major monetary and reputation losses.



Training



- **Best:** I attended this presentation live. (And was extremely impressed.)
- **Best:** Every developer, architect, and admin has MQ on their workstation.
- **Best:** We have a sandbox for infrastructure testing and POC, training.
- **Second Best:** We have no travel budget but we allow time for OJT training.
- **Second Best:** We make room in dev for infrastructure testing, training.
- **Second Best:** When we hire temp help, we include skills transfer in scope.
- **Lesser:** The MQ person took the training back in '01 or '02.
- **Lesser:** There's only a handful of verbs. How hard can it be?
- **Lesser:** We hired JMS people and MQ is compliant so no training needed.
- **Lesser:** [Stakeholder role] doesn't need to know about MQ, really.
(Where stakeholder role == [developer | architect | tester | NOC | etc.])

Training



- One customer on MQ v8.0 was automating configurations and workarounds dating back to MQ v5.3 and that broke v8.0 features.
- A customer implemented several MQ clusters consisting of two mainframe LPARS, believing individual messages would be highly available on both.
- A high-level law enforcement agency (not USA) gave out a p12 file with their public and private key to lower agencies connecting in.
- Multiple shops from government, public sector, banking, insurance, healthcare, and more have discovered business partners connecting with MQ Explorer to their edge QMgrs, often with administrator privileges.
- I once saw a network of about 20 QMgrs that had something like 40 overlapping clusters defined, none of which were healthy. In the pre-engagement negotiations I had naively responded "MQ clusters? Sure. Any size or shape. Bring it on."

Security



- **Best:** We implemented intrusion prevention, intrusion detection, strong authentication, compliance monitoring, forensic logging, least-privilege access model, and strong accountability tracking.
- **Best:** We use TLS for authentication and integrity as well as privacy.
- **Best:** We periodically get an external review and/or pen test.
- **Second Best:** All admin access is strongly authenticated and restricted.
- **Second Best:** We don't use SYSTEM.* channels for live connections.
- **Second Best:** We have intrusion prevention and some monitoring.
- **Lesser:** MCAUSER('mqm'), CHLAUTH(DISABLED)
- **Lesser:** Enabled TLS, CHLAUTH, exit for partner channels but not internal.
- **Lesser:** We had a big security project a few years back so we're good.

Security



- Many shops that do not turn security on in lower environments eventually get to Prod, have problems, disable security and release anyway. One such customer had no security whatsoever in Prod, said they'd been trying for years but it just never worked and could I get IBM to fix it?
- Did a pen test, broke in 5 minutes into the gig. Provided explicit instructions how to fix. Followed up a year later, broke in exact same way. Customer complained they got the same advice they'd already paid for.
- Department manager noticed I'd specified TLS for their cluster channels, told me it was out of scope. He claimed it was too hard to implement and operate. He relented when I explained I'd automated the cert management some time ago and the MQ team had been building out TLS-enabled queue managers for several months without even knowing.

Process



- **Best:** Day-to-day Admin tasks mostly on strategic, not as much tactical.
- **Best:** Tools and automation make it faster, cheaper, easier to consistently do the right thing than to rush out quick and dirty builds.
- **Best:** Team capacity and bench in reserve to handle emergencies.
- **Second Best:** Strategic tasks not majority of time spent but still high.
- **Second Best:** Rigorous procedures maintain quality, consistency.
- **Second Best:** Push back deadlines to make capacity for emergencies.
- **Lesser:** Scheduling entirely based on triage. Only urgent tasks worked.
- **Lesser:** Performance primarily based on task throughput.
- **Lesser:** Capacity, if needed , comes from working past the nominal 50 hours normally expected of all team members.

Process



- The shop that had the best server capacity management I've ever seen worked their people a nominal 60+ hours/week, had massive turnover, high absenteeism, and frequent outages.
- Raise your hand if you remember when “*the MQ person*” quit and you suddenly became “*the new MQ person*.”
- A manager once told me that if they bought a web-based admin tool, everyone would forget how to administer MQ by hand. They had a small MQ team, 3,000+ queue managers, 100,000 objects, and 2+ million access control list entries (not counting the mqm group entries). Nothing was touched unless it was brand new or broken. Everyone was intimately familiar with rumqsc, though.



Agenda

- Excellence
- Reverse-engineering excellence
- Mediocrity as a management strategy
- Gaming the 80/20 rule
- Some example MQ Best, Second Best, and Lesser Practices
- **Actively suppressing spontaneous excellence**
- Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)

Mediocrity Master Class

Chronic mediocrity will eventually break the will of your team. Until that happens though you may have to deal with unexpected spontaneous outbreaks of excellence.

This is because people are essentially lazy. They will come up with ways of working faster, better, repeatably, consistently, and collaboratively to benefit themselves, not the company. They readily admit to their laziness but claim the benefits qualify this as lazy-as-virtue. Your job is to impress on them that this is actually lazy-as-sin.

Strategies:

- Do not recognize people for taking initiative. Unless it fails, in which case single them out and publically shame them. Keep them around as an example to others.
- Home-grown tools and process improvements must always be found to conflict with some internal standard. There are enough internal standards that anything can be declared non-compliant.
- Anyone persisting in pursuit of excellence must be rated “not a team player” in their performance review and become ineligible for bonus.

** Yes, sadly, all of these are real-life examples and far too common.*

Agenda

- Reverse-engineering excellence
- Mediocrity as a management strategy
- Gaming the 80/20 rule
- Some example MQ Best, Second Best, and Lesser Practices
- Actively suppressing spontaneous excellence
- **Mediocrity means never having to use the question mark
(But we'll make an exception for those who have yet to make the transition)**

Questions & Answers

