

Lecture 2: Learning Open Science

Jeff Rouder

June 24, 2020





Contents lists available at ScienceDirect

Best Practice & Research Clinical Anaesthesiology

journal homepage: www.elsevier.com/locate/bean



3

High reliability organizations (HROs)

Kathleen M. Sutcliffe, PhD, Professor *

Department of Management and Organizations, Stephen M. Ross School of Business, University of Michigan, 701 Tappan St, Ann Arbor, MI 48109-1234, USA

Keywords:

high-reliability organisations (HROs)
high-reliability organising
system safety
accident prevention
resilience

Academic and professional disciplines, such as organisation and management theory, psychology, sociology and engineering, have, for years, grappled with the multidisciplinary issues of safety and accident prevention. However, these ideas are just beginning to enrich research on safety in medicine. This article examines a domain of research on system safety – the High Reliability Organization (HRO) paradigm. HROs operate in hazardous conditions, but have fewer than their fair share of adverse events. HROs are committed to safety at the highest level and adopt a special

}

Theory of High Reliability Organizations

- Aviation
- Acute Medical Care
- Nuclear Power Plants

Theory of High Reliability Organizations

(Four properties we are going to use)

1. Sensitivity to Operations
2. Preoccupation With Failure/Mistakes
3. Resilience In The Face of Failure/Mistakes
4. Deference to Expertise

Theory of High Reliability Organizations

- What do these four properties mean?
- How are they translated into a lab environment
- Application of Properties Is A Process
- Process Guides Organization-Specific Policy
- Not guidelines; ways or developing **your** guidelines

Property 1: Sensitivity to Operations

- We are in the knowledge production business
- We focus on the knowledge we produce.
- Sensitivity means putting emphasis on the mechanics of how we produce it.

Property 1: Sensitivity to Operations

- How can we insure the integrity and authenticity of our knowledge
- Example, how do we know we reported the right values in table.
- How do we know who did what where?
- How can we curate these processes or show provenance of knowledge

Property 2: Preoccupation With Mistakes

- Small mistakes must be taken as seriously as large ones
- Constantly quering systems with stress tests
- Anticipation of human failure in meticulousness

Property 3: Resilience In The Face of Mistakes

- Mistakes will happen; we must learn from them
- Mistakes will happen; we must deal with them transparently
- Mistakes result from complex causes, must not just call it “operator error.”

Property 4: Deference to Expertise

- Undergrads are on the front lines
- Grads and Post-Docs are closer to the production of data
- PIs have long-term vision and experience
- We can better improve our operation together

No Concrete Recommendations

- You practice high reliability theory
- The outcomes (policies, procedures) reflect your experiences, skills, problems, context.
- The following are things we have discovered by going through the process

1. A Lab Culture Focused on Learning From Mistakes



1. A Lab Culture Focused on Learning From Mistakes

- I have something to tell you



1. A Lab Culture Focused on Learning From Mistakes



- I have something to tell you
- And it has been making me sick; I have lost a lot of sleep over it

1. A Lab Culture Focused on Learning From Mistakes



- I have something to tell you
- And it has been making me sick; I have lost a lot of sleep over it
- All My RTs Are Too Slow

1. A Lab Culture Focused on Learning From Mistakes



- I have something to tell you
- And it has been making me sick; I have lost a lot of sleep over it
- All My RTs Are Too Slow
- We did not catch trials that were too quick

1. A Lab Culture Focused on Learning From Mistakes



- I have something to tell you
- And it has been making me sick; I have lost a lot of sleep over it
- All My RTs Are Too Slow
- We did not catch trials that were too quick
- This has been going on for quite a while

1. A Lab Culture Focused on Learning From Mistakes



1. A Lab Culture Focused on Learning From Mistakes

- Why? Read time stamps array from [1] not [0].



1. A Lab Culture Focused on Learning From Mistakes



- Why? Read time stamps array from [1] not [0].
- C is a mistake waiting to happen

1. A Lab Culture Focused on Learning From Mistakes



- Why? Read time stamps array from [1] not [0].
- C is a mistake waiting to happen
- Why are we using a language that promotes mistakes.

1. A Lab Culture Focused on Learning From Mistakes



- Why? Read time stamps array from [1] not [0].
- C is a mistake waiting to happen
- Why are we using a language that promotes mistakes.
- Grad students would rather use psychophysics toolbox in MatLab.

1. A Lab Culture Focused on Learning From Mistakes



- Why? Read time stamps array from [1] not [0].
- C is a mistake waiting to happen
- Why are we using a language that promotes mistakes.
- Grad students would rather use psychophysics toolbox in MatLab.
- Done!

1. A Lab Culture Focused on Learning From Mistakes}

We record mistakes!

Details

AdverseID	8
Member	Jeff
Problem	Jeff zipped and submitted wrong version of a paper to typesetter. Directories were named sub, sub2 and rev1. Jeff zipped and sent sub2 when rev1 was the publication version.
Solution	<p>Solution 1. In the old days, we used to have p1, p2, p3, ..., with the latest one being p. When one wanted to archive a new version, one simply cp p to px, where x is the version number.</p> <p>Solution 2. Let's be sure to use git tags. That way we can keep everything from a paper together and tag versions.</p>
Resolution	<p>Let's adopt the following directory naming standards for paper versions.</p> <ol style="list-style-type: none">1. Each paper has a single name. That name should be used for all versions. It should be short but descriptive. Not sub, rev, or current. Examples are labRules or AMPPS18.2 There are two acceptable procedures for major versions.<ol style="list-style-type: none">a. Use git tagb. Use directories with version numbers: labRules.1 and labRules.2
PolicyChange	Yes
LastModified	Sat Sep 1 08:17:57 2018

2. Audit Trails

- Audit trails refer to the easy ability to reconstruct who did what when.
- What are some ideas?

2. Audit Trails

- Standardized filenames and directories help
- Versioning
- Automatization
- Databasing
- Coded Analysis
- Expanded Documents

2a. Standardization

- Care to share how you name directories and files?

2a. Standardization

Which is the final copy?

- final.docx
- reallyFinal.docx
- myFinal.docx
- thisIsTheFinalVersion.docx

2a. Standardization

Experiments

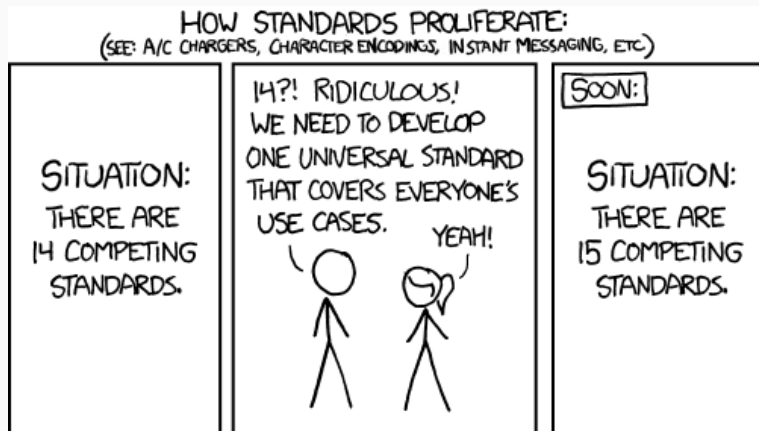
- Organized in series of related experiments, “baserate” might contain “br1”, “br2”, ...
- data files are br1.dat.001, br1.dat.002,...
- Session information files are br1.ses.001, br2.ses.002,... -Also br1.ins.txt, br1.debrief.txt,br1.info.txt (or br1.txt), and br1.all (all data in one file)
- No cleaning allowed. Only raw data.
- No other files

2a. Standardization

Projects:

- A self-contained collection of files on a common idea.
- Must be stored on approved site.
- Project titles have an organizing prefix, e.g., bf-,ctx-,irt-,stat-,wm- and more. “bf-aovlite” contains Rouder et al., 2017, Psych. Methods.
- Projects have zero, one, or more outputs: Papers, Presentations, Grants.
- Directories in a project are “dev”, “papers”, “presentations”, “grants”, “share”

2a. Standarization (With Care)



2b. Versioning and Logs

- I am not a fan of manual version control
- SKILL: git
- Has a log function that helps with audit trail.

2c. Computer Automation

- Computers Do Everything That They Can
- SKILL: shell scripting

2c. Computer Automation

- **Experimenter** and **Participant** log in
- **Computer** scans directory, determines filename,
- **Computer** queries database to see if this participant is there
- **Computer** runs mini-survey for demographics if needed
- **Computer** records all trial level information in one file
- **Computer** records all session level information in two files (identified, not identified)
- **Computer** updates lab database
- **Computer** uploads data to the cloud nightly (trial-level + deidentified)

2d. Databasing

- You have a collection of 80s wonder songs



2d. Databasing

- You have a collection of 80s wonder songs
- iTunes



2d. Databasing

- You have a collection of 80s wonder songs
- iTunes
- Filenames and directories are inferior



2d. Databasing

- You have a collection of 80s wonder songs
- iTunes
- Filenames and directories are inferior
- Friendly choices include Microsoft Access, Apple Filemaker Pro, **Open Office Base**



2d. Databasing

```
\includegraphics<1>[width=15cm]{figs/dataBase}  
\includegraphics<2>[width=11cm]{figs/dataBase-Exp}  
\includegraphics<3>[width=11cm]{figs/dataBase-Session}
```


2e. Databasing

- SKILLS: mysql
- HELPER: phpmyadmin

2e. Coded Analysis

- Menu driven vs. text-driven (coded)
- Coded inherently contains its audit trail
- Rule 1: Must be coded in a computer language (R, Matlab, SAS,
- Rule 2: Must download data from URL (web), URL coded too
- Resultant: Code is sufficient to replicate the analysis.
- SKILL: R

2f. Expanded Documents

- We tend to think of the document as a thing
 - An output
 - A file
 - Describes knowledge production process

2f. Expanded Documents

- Expand the document to include the knowledge production process itself!
- Execute a document:
 - downloads data
 - does analysis
 - typesets figures, tables, equations
 - organizes bibliography
 - launches webapps
 - surrounds these elements with your words

2f. Expanded Documents

- SKILL: Rmarkdown

Highly Reliable vs. Open Science

- High Reliability should be uncontroversial
- High Reliability is a precursor to Open Science
- Open Science is far more scary

We Are Born Open on the Data Side

- If we collect it, the you can see it immediately.

“Jeff, should I delete it before it uploads?”

- Our data have warts: machines crash, people quit early, bad settings, etc.
- Data that should not be used in a proper analysis will be uploaded
- Let it upload. We will exclude it in the analysis code.
- We may be vulnerable to critique, but we did our best, so have self compassion

“Jeff, there are so many long RTs”

- Because it is open, the discussion has come up quickly.
- We may document our explorations in the analysis code
- We may be vulnerable to critique, but we did our best, so have self compassion

“Jeff, the stimulus durations were too short. We need to rerun the experiment with longer durations. Should we just delete this experiment from the server?”

- No.
- We may be vulnerable to critique, but we did our best, so have self compassion

Vulnerability and Self Compassion

- Open Science requires some vulnerability and self compassion
- Vulnerability and self compassion are needed to write
- Vulnerability and self compassion are life skills