16×9

CONTROLS ORGANIZATION

BOOTSTRAP, QUALITY

Prototype Document

A convergence of twisty dependencies.
0b-verge-proto-doc

First Last

May 21, 2024

Prior:

RACI
Responsible: First Last
Authority: Management, Senior
Contact: Leadership
Inform: Staff

A framework for the design of operational controls, incorporating authorization, and iterative optimization, as a structured development process, is not easily phrased, and the implementation is equally complex. Yet, simultaneous framework and operational development, with authorized implementation, maximum value, least time, and lowest cost is fundamental underpinning of nearly every project; regardless of technicality. Many years ago, I learned two valuable lessons from the pharma industry: "If it isn't documented, it didn't happen," and what I will characterize as, "Repeatability is the mother of improvement."

What begins as a system process development template, an initiative I've revisited many times, quickly collides with a dilemma: should I abandon the generic framework with abstract inputs and outputs—applicable to any project—in favor of real-world integration examples? The former is inherently inaccessible for education, lacking the elucidation, and verification an example provides, and which is also a great asset for development; while the latter introduces an application barrier, the example data must be filtered out, with every new template application. So, why not both; a process framework, with examples!

The challenge is that framework development and example-populated framework development are separate, yet interdependent, projects. Their integration essentially forms a third project. As details are refined and applied, the time to iterate the development cycles is compounded. Although using examples is the best way to uncover optimization opportunities, populating frameworks with examples becomes a distraction from refining the framework itself.

The simple yet crucial answer is that you must provide examples with templates—especially for process templates—because without input and output examples, process understanding is elusive. Even generative AI struggles with application of example free abstract process; prompt for a novel application, of multiple preexisting activities or objects, and that usually results in difficulty. If the idea is truly new, the AI system has difficulty identifying the solution (without bootstrapping a randomized simulation), even when additional prompts converge on an assembly of the solution, the step of *inventing the application of components* and *providing the novel solution,* is illusive.

Take a moment to prompt AI about the pros-and-cons of coconut oil as a lamp fuel, and heat source. A solution using this fuel has critical barriers, yet comes with significant benefits, all about which generative AI will pontificate, if the solution is provided. Yet identifying the solution is missed by generative AI, because examples are not significantly represented, in the training corpus. (At least at the time of this writting.) Google doesn't fare much better, try getting beyond a paraffin, canola, and olive oil comparisons for lamp fuel; and, the health aspect of using the various oils on your skin. These are popular topics and comprise the first 50,000 or so search results. The critical challenge of using coconut oil as a lamp fuel is the burn temperature. It is so high that cotton wicks are quickly vaporized, causing the flame to descend into the fuel, and become smothered. Glass fiber wicks are no more viable. The heat causes the glass to melt, and mushroom like structures form on the tips, preventing fuel from wickking up, in a few hours, the flame is out, and the wick is ruined. Half of the solution is to use carbon felt, a material with the consistency of regular felt, and a burn temperature suitable to provide protection from acetylene welding. The relatively nominal challenge of crafting a coconut oil lamp from carbon felt can be a life saver, in a winter emergency power-loss situation. There is one other hurdle, the energy density of coconut oil is so great that it requires a flue, like that caused by the glass bulb of a hurricane lamp. Convection currents draw air from the bottom to feed oxygen to the flame. While generative AI may pontificate the pros-and-cons, data sheets, physical properties, accuracy, and viability of this solution; try and prompt for it. Describe the goal,

and barriers, but without actually presenting the assembly of the components, and characterization of the solution details. I'll give a reward of some sort to the first person who can show me prompts that lead a commodity AI platform to identify this (or a better solution!) without actually describing how the components solve the challenges, of using coconut oil as a fuel. Bonus points for also prompting AI to identify coconut oil as a superior domestic lamp fuel. As for the health aspects, I've given up searching for answers, but it stands to reason that coconut oil is at least better than any other option.

Generative AI demonstrates *what we call understanding,* only after a novel solution is provided (or ironically, after we are presented with solutions that are new to us), then it can explain every aspect of, how, or why the solution works, in great detail. Much like a juvenile "know-it-all" personality exhibits an amazing degree of knowledge yet with little understanding of application, an inability to innovate, and probably acute symptoms of the Dunning-Kruger effect. The difficulty of discussing process without specific examples led to the creation of a place marker for a guidance document called "Governance, Risk, and Compliance Sourcebook," and eventually the subtitle that better describes the series, *"A Controlled Operating Documents Approach."* The document was intended to catalog various aspects operational control, bookmarking them for later development, as needed, so focus could remain on developing the operation that inspired the most recent framework initiative effort.

So, where are we? Is the next step documentation of an operations framework, or the application of the framework to document the implementation of my pet project? As it happens, it turns out doing both would become my new project. Fortunately, for anyone reading this far, that means the guidance won't simply be a GRC template sourcebook, but an actual book, outlining the sequential roll out of foundational documents, their authorization, and development, for a fictitious organization, called the *Controls Organization.* An initial document (or three) will define documents, their cataloging, and authorization; subsequent documents will develop other aspects of process, quality, compliance, and controls. And, of course no framework would be complete without a demonstration application example!

The pet project is an implementation of PKGSRC under a hybrid of Quality frameworks including NIST RMF, Multi-Vee, Iceburg, DevOps, GxP, and DevSecOps. In practice, there are significant challenges maintaining Platform, OS, and Software qualification controls, through revision and release cycles. I will attempt to develop the guidance to address, and resolve, those challenges, in the most generic, and optimal way possible!

If you are interested in that sort of thing, and especially as it applies to development of controlled compute platforms for scientific applications, read on! And, if you know a thing or two, please reach out, this is a work in progress, and I would love to expand the list of co-authors and collaborators!

Very Respectfully,
George Georgalis
Principal Author
george@galis.org