ERP CONTROLS OPTIMIZATION FOR FRAUD PREVENTION AND CONTINUOUS CONTROLS MONITORING

(EMBEDDED PROCESS-LEVEL CONTROLS IN SAP, ORACLE, AND PEOPLESOFT ETC.)

COURSE DESCRIPTION

In a time long ago, in-house COBOL apps … custom code, were loaded with process-level (e.g., transaction-level) controls. These controls were called "edits" and "validations" and were designed to stop "garbage in/garbage out" in its tracks. A great programming rule-of-thumb was to have the first 40% of your code to act as data "triage." Ah, life was good.

Then ERPs were born, and things went a little south. Don't blame the ERP … the controls are in it (in fact, they are embedded in it … "native"), these to-die-for controls just never seem to get invoked. You realize that we are talking about CONTROLS THAT NEVER SLEEP. No more sample size "issues." Well, it's time to optimize these ERPs (and make auditing easer to boot!)

World-class business runs at the speed of technology; shouldn't the controls do the same? If applications run in real-time/run time), doesn't it follow that controls need to be in real-time run-time too? What good are detective (after-the-fact) controls, when the dastardly deed is done?

Savvy business managers know the value, and competitive advantage, of information. Embedded controls provide intelligent modeling and monitoring, taking the business of business to "the next level" of operational excellence.

Remember the goal of embedded controls, and CCM, is to monitor *AND* to monitor the monitors. (What audit can do is to monitor the monitors who monitor the monitors.) Also, *if it can measure, it can be modeled, and if it can be modeled, it can be monitored*. This is cerebral business, business control, risk management (and audit) founded in management science.

It is ironic that you can audit, MORE, by doing LESS, as "pinging" embedded controls is a real "test of control operation effectiveness," and less time consuming than after-the-fact file sweeping. This paradox is true if you take an engineering approach (e.g., process focus) to audit. **Audit shops have reported a 90% reduction in audit time AND stronger testing!** By continuously monitoring core business processes, via embedded controls and mathematical modeling, you may locate material errors in real-time/run-time, strengthen the control environment, and manage down business risks. Additionally, CCM has been shown to reduce fraud/waste/abuse and identify best practices. ("What gets measured gets done".)

CCM is a 4-pronged approach. Central to this approach is the invoking of the vast number of process-level embedded controls … NOT controls *OVER* the process (think access controls); but rather controls *IN* the process. ERPs, like SAP and Oracle, are loaded with embedded control opportunities. Unfortunately, because it takes forever to "get the ERP up and running," these control OPPORTUNITIES … get lost. That is, your ERP is "up" … but it is NOT OPTIMIZED from a control perspective. These embedded controls exist 100% of the time, so they are TO-DIE-FOR!

WHO SHOULD ATTEND

- All auditors: Program, Compliance, Fraud, Operational, Efficiency, SOX, IT, Process and Financial
- Senior and Line Management (IT, Business Units, Budgeting, Forecasting, Supply-Chain)

Detailed Course Outline

1. The Professional Literature, and Stakeholders, "Call" for CCM
   a. The COSO, CobiT and the PCAOB
   b. Examples of monitoring in "the real (business) world"
2. What is CCM
   a. CCM from 10,000 feet
   b. The 4 necessary components of CCM
3. The Audit Risk Model
   a. The staggering cost of audit evidence
4. Internal Controls Make a "Comeback"
   a. CobiT for Application Controls and PCAOB AS 5
   b. Tests of Controls: TOEs and TODs
   c. Are you really "testing" the control … or just "reviewing" it
5. Application Controls and Opportunities Lost
   a. Controls embedded in ERPs (Oracle, SAP … whatever)
   b. Triggers and Configurables
   c. Application Controls and their Relationship to General Controls
6. Macro-Level Design Issues
   a. Who builds the CCM?
   b. Who hosts the CCM?
   c. Who monitors the Monitors?
   d. What is real time? Run-time? Near real time?
   e. Options for the CCM User
   f. What do you monitor?
   g. Would you REALLY stop a transaction?
   h. Why CCM requires both halves of the 1-2 punch.
7. Tactical and practical: Micro-Level Design and Implementation Issues
   a. Implementation methodology-Macro
   b. Implementation methodology-Micro
   c. Push-backs from I.T., and How to Dispel Them
   d. Capturing the concerns of the Business Unit Manager (focus groups)
   e. Critical Success factors and Key Performance Indicators
   f. Modeling the business rules (regression and configurables)
   g. SAP code: ABAP, Oracle code: PL/SQL and legacy COBOL
   h. Filtering data … or data overload
   i. Guardrails
   j. Recalibration of embeddeds and guardrails
   k. Constructing Dashboards and Cockpits
   l. Backfilling the cockpit: data pipes
   m. Total Costs for CCM
   n. Obtaining traction: 3 different levels of initial installation (go live)
8. Actual Installations of CCM
   a. Lessons learned from over 20 years of installing CCM
   b. Business implementations (WINS!)
9. Question and Answer Session
10. Fitting CCM to the Your Company

Course Objectives

Upon completion of this course, students will be able to:

- Cite and discuss the current literature regarding CCM.
- Review the new COSO for Monitoring (2009) and new CobiT for Application Controls (2009).
- Explain the necessary components of CCM.
- Define and operationalize CSFs and KPIs.
- Recognize the vast opportunities of embedded controls.
- Differentiate between controls "placed in operation" and controls "operating effectively."
- Identify the embedded hooks in major applications (SAP, Oracle, PeopleSoft).
- Realize the importance of "asking the right questions" to Business Unit Managers as regards to computerized controls.
- Understand the resource requirements to get CCM off the ground.
- Anticipate the potential obstacles to CCM.
- Design continuous control monitors.
- Determine that WE ALREADY HAVE most of the capabilities we need to facilitate CCM.
- Explain why the necessary components of CCM are a 1-2 punch.
- Discuss business cases of the 1-2 punch.
- Consider CCM tactical and practical implementation options.
- Review actual CCM "wins" and determine the low hanging fruit to go after.

Course Length

- 1 day

Dr. Dan Kneer Advisory Group
3639 Midway Drive, Suite B370
San Diego, CA 92110 USA
+1.619.223.1521

www.dankneer.com