Introduction to Automated Controls

Jay Swaminathan

ISACA®
Serving IT Governance Professionals
San Francisco Chapter

Agenda

- Defining Automated Controls
- The Value of Automated Controls
- Common Testing Approaches
- ITGC considerations
- The Concept of ‘Benchmarking’
- Capabilities of GRC tools
- Increasing reliance on automated controls
- Questions / Comments

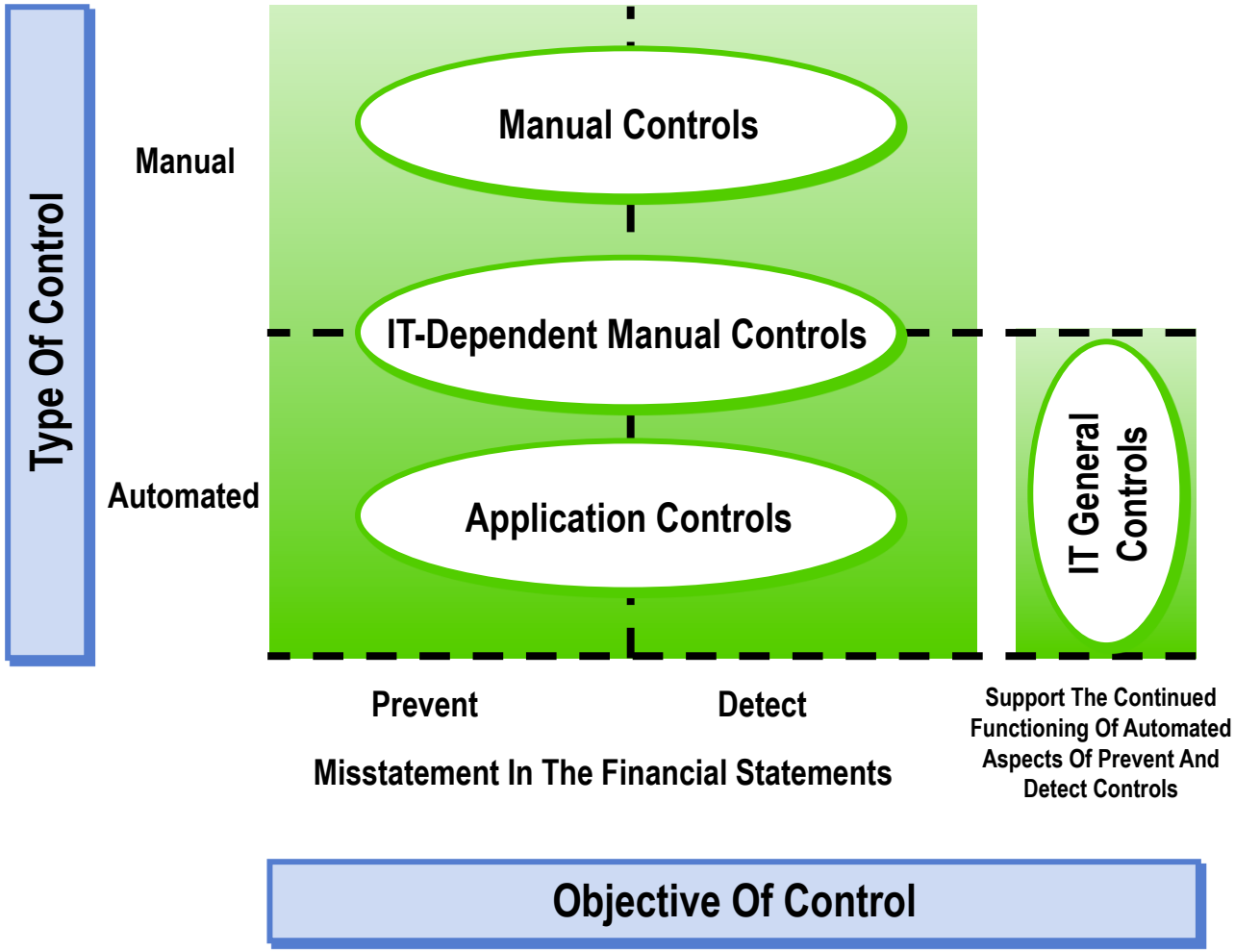
ISACA®
Serving IT Governance Professionals

San Francisco Chapter

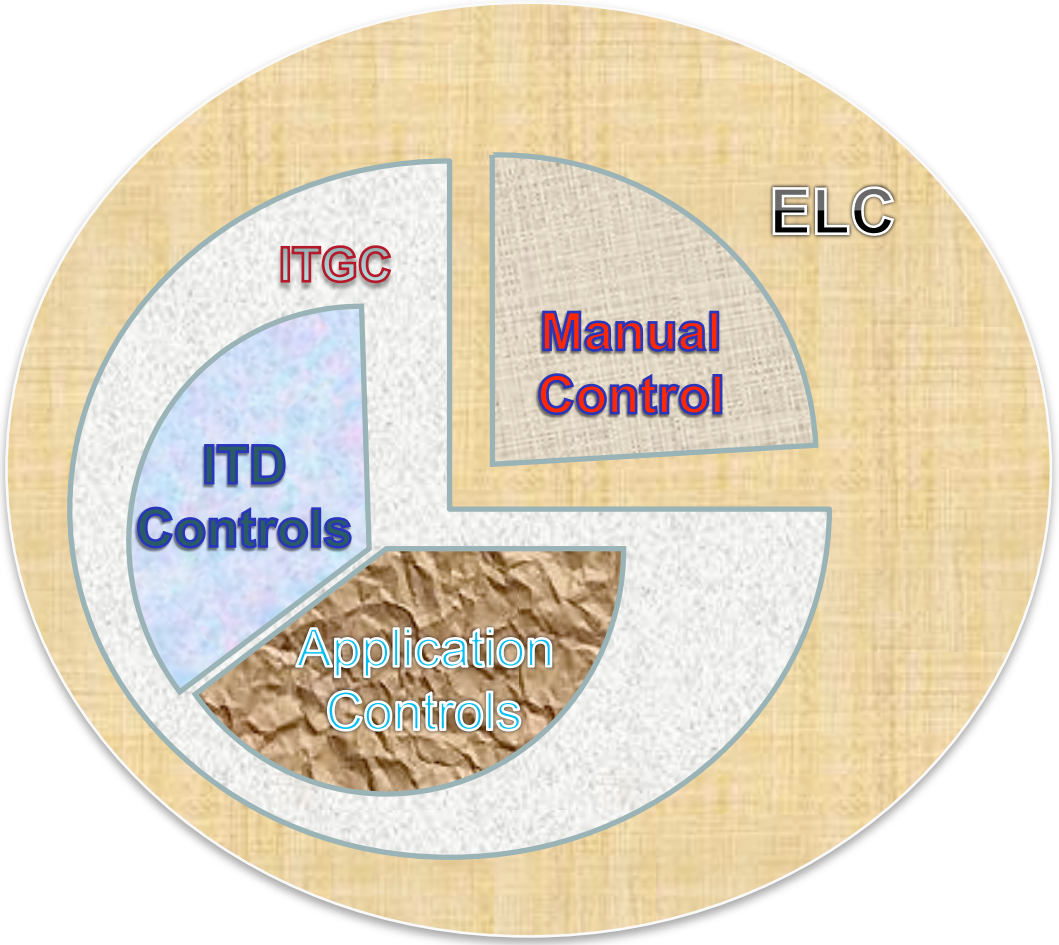
Examples

	Example 1	Example 2	Example 3	Example 4
	System calculates Depreciation based on setting	Three way match	System enforced journal approval based on limits	Custom logic to enforce sales order limits by sales person
Type	Inherent	Configurable	Configurable	Customized
Nature	Calculation	Validation	Authorization	Authorization
Key Change Manage considerations	Program changes	Program and Configuration changes	Program changes	Program changes
Key Logical Access considerations	None	Access to configuration	Access to configuration	None
Walkthrough	Positive	Negative	Positive	Positive

Categories of Controls



Control Layout



Type of Controls

- Inherent processing and controls
  - Built into the application
  - Examples: DR = CR, Depreciation calculation, etc.
- Programmed controls (custom coded)
  - Custom functionality – Based on specific business requirement
- Configurable controls
  - Customized and can be disabled or set up to operate in different ways
  - Examples: three-way matching, auto-accounting

Nature of Application Controls

- Validations
- Calculations
- Authorizations

Application Controls –
If it works once, will work consistently

Examples

	Example 1	Example 2	Example 3	Example 4
	System calculates Depreciation based on setting	Three way match	System enforced journal approval based on limits	Custom logic to enforce sales order limits by sales person
Type	Inherent	Configurable	Configurable	Customized
Nature	Calculation	Validation	Authorization	Authorization
Key Change Manage considerations	Program changes	Program and Configuration changes	Program changes	Program changes
Key Logical Access considerations	None	Access to configuration	Access to configuration	None
Walkthrough	Positive	Negative	Positive	Positive

Application Controls Benefits

- Implement once (cost depending on type of control)
- Lower cost in operation of control
  - Less dependence on humans
  - Fewer errors
  - Less paper
- Control assessment usually more efficient
  - Test of One
  - Benchmarking

Random Application Control points

- Control nomenclature
  - SOAP

  Subject, Object, Action and Purpose
- Control where system defaults information are not strong
- Logical Access controls as Application controls
- Restricted Access & SOD Controls
- Ignorance is not a control

Testing Approach

- Test of Design (Test of one)
  - Inquiry and observation procedures.
  - Review of configurations for configurable control
  - Examination of one or more transactions to confirm the design.
- Test of Effectiveness
  - Rely on underlying IT General Controls

Questions / Discussion:

- When is a ‘negative test’ appropriate?
- How to confirm whether setup is same across the whole organization
- What additional considerations for configurable controls
- Do we review code for customizable controls?

Testing Examples

- Inspect configuration
  - Inspect 2/3/4-way match configuration
  - Inspect tolerance levels configured
- Re-performance via a walkthrough of each significant flow of transactions
  - Demonstrate the operating effectiveness of the control via positive and negative confirmation
- Inspect the authorizations and re-perform controls to confirm the design
  - Inspect privileges assigned to all users
- Determine how overrides are possible throughout testing and how they are monitored

ITGC Considerations

- IT General Controls must be effective
- ITGC must cover automated controls (e.g., configuration changes)
- Configuration not made at entity/instance level (customer, supplier, item, etc.)

ITGC Considerations continued…

- SOD between access to configuration vs. transaction

- Emphasis could shift between change management and logical access
  - Authorizations, configurations – Logical Access
  - Calculation, customization, Inherent – Change Management

ITGC Concerns

Change Management

- Ability to make code changes is not limited to programmers
- Standard change management process not followed for configuration settings

Logical Access

- End users have ability to change configuration settings (Users Vs. Super Users Vs. System administrator)
- Override of the control by super users or system/database administrators
- Improper Segregation of Duties

  (Create document Vs. release holds)

Testing – Ineffective ITGC

- Sample based Application control testing
- WCGW never went wrong
  - E.g. configuration not changed although change management around configurations not effective.
- Professional judgment on inherent controls

Examples

	Example 1	Example 2	Example 3	Example 4
	System calculates Depreciation based on setting	Three way match	System enforced journal approval based on limits	Custom logic to enforce sales order limits by sales person
Type	Inherent	Configurable	Configurable	Customized
Nature	Calculation	Validation	Authorization	Authorization
Key Change Manage considerations	Program changes	Program and Configuration changes	Program changes	Program changes
Key Logical Access considerations	None	Access to configuration	Access to configuration	None
Walkthrough	Positive	Negative	Positive	Positive

Benchmarking

- Benchmarking is the ability to roll forward prior conclusions on control effectiveness based on the ability to demonstrate a static and controlled environment.
- Historically very difficult to achieve due to complexities within the ERP environment and the dynamic (regularly changing) nature.
- GRC Software packages now making true benchmarking feasible.

Question / Discussion: Does benchmarking become irrelevant if continuous monitoring (via GRC tools, etc.) can be achieved?

Benchmark Testing Approach

- Monitoring
- Rotational Testing

GRC Capabilities

- Monitor SOD real time
- Efficiently Implement SOD prevent controls
- Monitor configuration changes real time
- Document risks, audit, findings, etc

Case Study

Expanding Reliance on Automated Controls

ISACA®
Serving IT Governance Professionals
San Francisco Chapter

Objective

- Identification of unmitigated risks or redundant controls
- Identify additional automated controls
- Increase the efficiency of testing the controls

Rationale

- Once implemented, application controls are significantly cheaper to operate.
- Application controls are more consistent and secure than manual controls.
- Application controls are very often the only controls within an automated process.
- It could be more efficient to rely on application controls rather than doing substantive testing.

Process

1. Meetings with Process Owners to understand the process
2. Working session to determine control set and testing approach
3. Draft implementation plan
4. Confirm changes and discuss the plan to implement

Result

- Identified controls that were already implemented and contributed to the mitigation of risk
- Implemented new application controls that reduced the need for manual controls
- Used benchmarking for some application controls to increase the efficiency of the controls assessment

Control mix **prior** to review – 90% manual, 10% automated

Control mix **after** review – 50% manual, 50% automated

Questions?

ISACA®
Serving IT Governance Professionals
San Francisco Chapter