

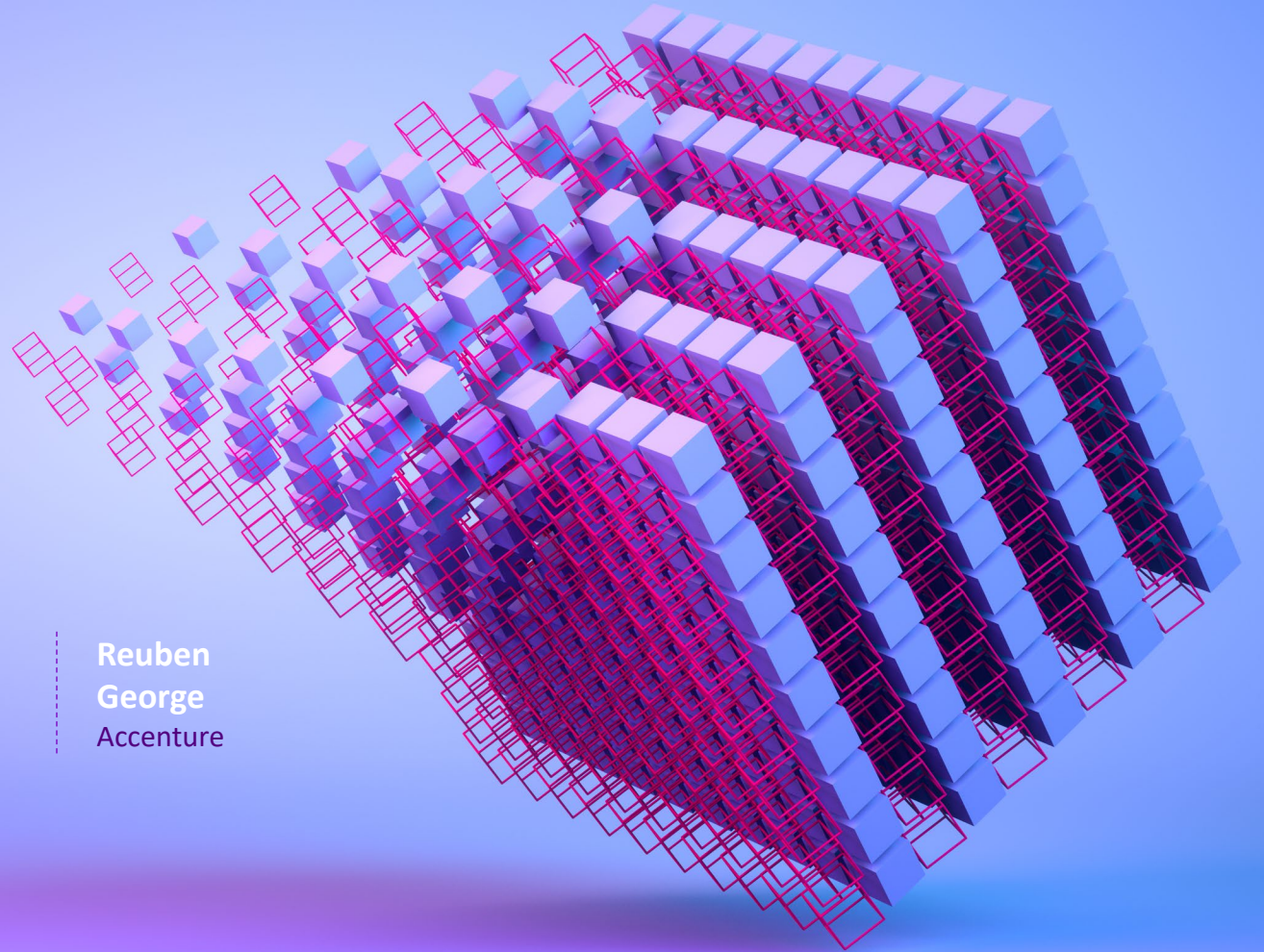
# Application of STPA in Distributed Digital Infrastructure (eCommerce Example)

**Mahesh  
Venkataraman**  
Accenture

**Koushik  
Vijayaraghavan**  
Accenture

**Ram Ramalingam**  
Accenture

**Reuben  
George**  
Accenture



# Aviation & Distributed Architectures

Trade offs (Goal, Design & Runtime)

Envelope & Boundaries

Human Factor – Intent Vs Outcomes

# Reality Check: Major Cloud Outages & Degradations

Date	Service Provider	Duration	Impact
July 2024	CrowdStrike	~5 hours	Faulty update caused Windows machines to enter boot loops or recovery mode, affecting global operations across various sectors.
July 2024	Microsoft Azure AI	~20 hours	Azure's OpenAI services experienced a global outage due to a routine operation that inadvertently disabled critical code, impacting AI-driven tools.
July 2024	AWS Kinesis	~7 hours	APIs connecting AWS Kinesis experienced slowdowns and errors, particularly impacting logistics and financial services.
September 2024	Microsoft 365	~3 hours	Users worldwide were unable to access services like Outlook and Teams due to a third-party ISP change.
October 2024	Microsoft 365	~2 hours	Outlook, Teams, and Office 365 experienced outages due to a memory management issue.
December 2024	Microsoft Outlook	~2 hours	Global outage left thousands unable to access emails; cause identified and services restored.

Sources: Wikipedia [1], The Insurer [2], Reuters [3], NY Post [4], Business Insider [5]

[See reference slide for full URLs](#)

“ You can’t legislate against failure; focus on *fast detection & response* ”

↑  
Observability  
(Feedback)

↑  
Control

## Chris Pinkham

Co-led the Cape Town-based team that developed Amazon’s Elastic Compute Cloud (EC2)

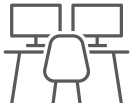
# STPA vs Modern Distributed Systems Techniques

Dimension	Chaos Eng.	Observability	Incidents Analysis	Resilience Testing	Perf. Eng.	STPA
Purpose & Focus	Fault Injection	Telemetry Signals	Postmortem	Availability	Latency & Load	Safety & Intent <input checked="" type="checkbox"/>
When Applied	Runtime	Runtime	Post-incident	Pre-deploy	Pre/Post	Design-time <input checked="" type="checkbox"/>
What It Discovers	Fault Patterns	Symptoms	Known Failures	Recovery Gaps	Bottlenecks	Systemic Hazards & Risks <input checked="" type="checkbox"/>
Systemic Insight	Limited	Fragmented	Partial	Limited	Partial	Holistic, Multi-layer <input checked="" type="checkbox"/>
Human Factor Coverage	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Some	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Explicit Modeling <input checked="" type="checkbox"/>
Emergent Property Handling	Runtime Validation	Not Focused	After-the-fact	Simulated Scenarios	Measured Outcomes	Modeled & Anticipated Early <input checked="" type="checkbox"/>

# Losses

- Financial Loss (L1)
- Customer Attrition (L2)
- Damaged Reputation (L3)
- Legal & Regulatory Penalties (L4)
- Operational Inefficiencies (L5)
- Missed Growth Opportunities (L6)

# Hazards



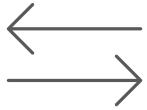
Unavailability of  
Critical Functions  
**(H1)**



Data  
Inconsistency  
**(H2)**



Unauthorized  
Access  
**(H3)**



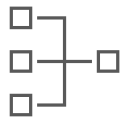
Unprocessed  
Transactions  
**(H4)**



Security  
Breach  
**(H5)**



Compliance  
Failure  
**(H6)**



Degraded System  
Performance  
**(H7)**

# Loss – Hazard Mapping

Hazard (H)	L1 Financial Losses	L2 Customer Attrition	L3 Damage to Reputation	L4 Legal & Regulatory Penalties	L5 Operational Inefficiencies
H1: The system is unable to provide essential services during peak traffic periods.	X	X	X		
H2: Data across the system becomes inconsistent, leading to mismatches in information.	X	X			X
H3: The system allows unauthorized access to sensitive data or functionalities.			X	X	
H4: Transactions are not processed correctly or fail to complete.	X	X	X		
H5: Security Breach.		X	X	X	
H6: The system is unable to comply with regulatory or legal requirements.			X	X	
H7: The system experiences delays or degraded performance in critical processes.	X	X			X

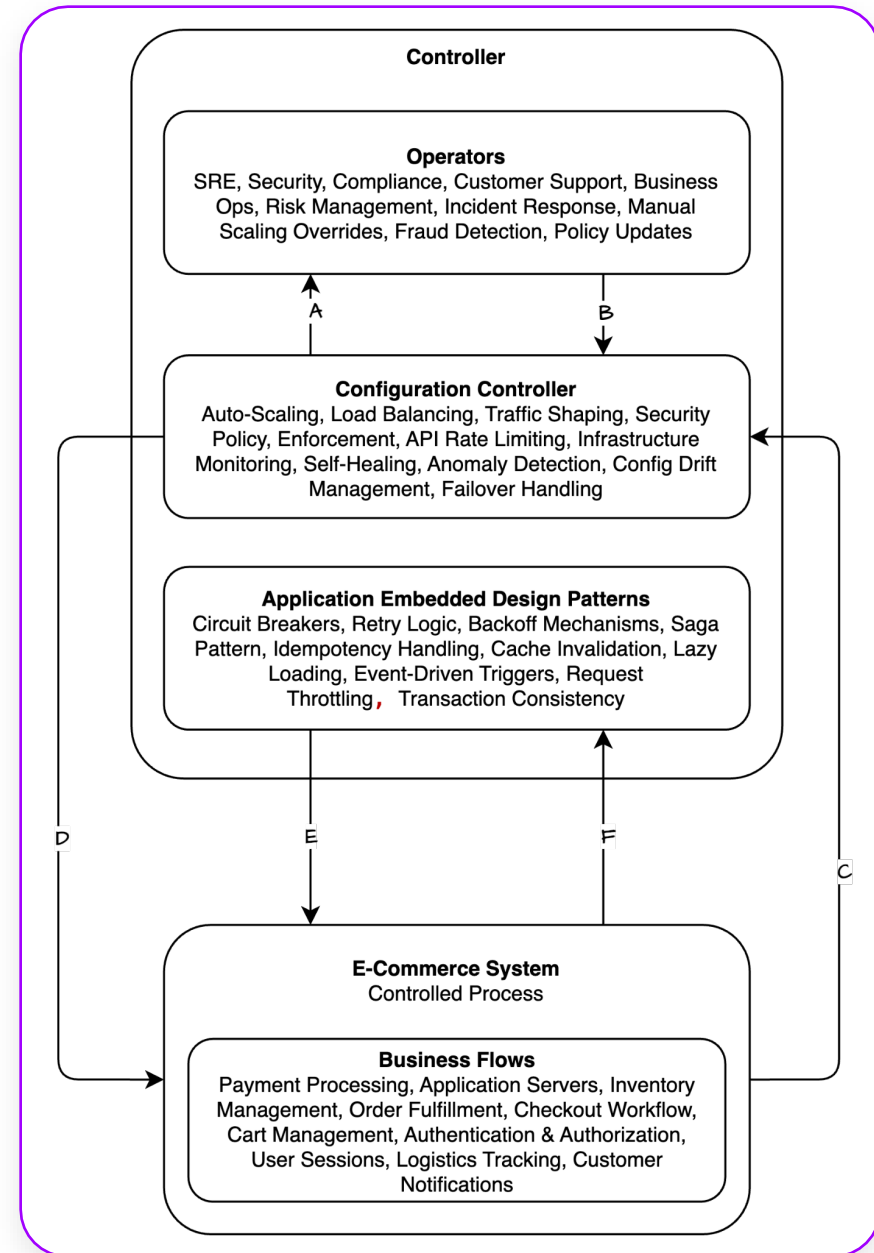
# System Constraints - 1

Hazard (H)	L1	L2	L3	L4	L5	System Constraints
H1: The system is unable to provide essential services during peak traffic periods.	X	X	X			<p>SC1: The system shall sustain 99.99% uptime during forecasted peak traffic, with no single point of failure.</p> <p>SC2: The system shall automatically recover disrupted services within 2 minutes of detecting unavailability.</p>
H2: Data across the system becomes inconsistent, leading to mismatches in information.	X	X			X	<p>SC3: For all critical records (e.g., inventory, orders), any update must be reflected across all replicas within 1 second, and no stale reads shall be served for more than 2 seconds.</p> <p>SC4: Data inconsistencies must be detected within 1 minute and resolved within 5 minutes through automated reconciliation.</p>
H3: The system allows unauthorized access to sensitive data or functionalities.			X	X		<p>SC5: Access control shall enforce role-based access policies on 100% of sensitive endpoints.</p> <p>SC6: Unauthorized access attempts must trigger alerts within 10 seconds and block further access automatically.</p>

## System Constraints – 2

Hazard (H)	L1	L2	L3	L4	L5	System Constraints
H4: Transactions are not processed correctly or fail to complete.	X	X	X			SC7: Transaction success rate shall remain $\geq 99.5\%$ across all payment flows. SC8: Duplicate or failed transactions must be auto-rolled back or flagged within 60 seconds.
H5: Security Breach		X	X	X		SC9: All security anomalies (e.g., failed login bursts, token reuse) must be flagged within 30 seconds. SC10: Breach attempts must trigger containment (e.g., user session kill) within 2 minutes.
H6: The system is unable to comply with regulatory or legal requirements.			X	X		SC11: All PII fields must be stored and transmitted only using AES-256 encryption and TLS 1.3. SC12: All audit logs of regulated transactions must be retained for 7 years and accessible to compliance teams within 5 minutes of request.
H7: The system experiences delays or degraded performance in critical processes.	X	X			X	SC13: 95% of customer-facing APIs must respond within 300ms under normal load. SC14: The system shall auto-scale within 30 seconds of crossing 70% CPU or memory utilization.

# Setting up control structures for an eCommerce Business



# Control Structure

Aspect	Application Embedded Design Patterns	Configuration Controller	Operator
Scope	Governs application behavior dynamically through embedded runtime logic.	Manages system-wide configurations dynamically (e.g., auto-scaling, traffic routing, security enforcement).	Oversees strategic interventions, compliance enforcement, and critical operational decisions.
Level of Execution	Operates within the application codebase, ensuring runtime adaptability (e.g., retries, circuit breakers, consistency mechanisms).	Modifies infrastructure settings dynamically based on system metrics and policies.	Executes manual overrides and governance decisions that impact both application and system runtime behavior.
Decision-Making	Autonomous, making decisions based on internal application signals (e.g., request failures, timeout thresholds).	Adaptive, responding to system-wide conditions and monitoring feedback for configuration enforcement.	Human-driven, analyzing trends, adjusting policies, and intervening during major incidents.
Examples	Circuit breakers, saga pattern, lazy loading, event-driven cache invalidation, request throttling, transactional consistency mechanisms.	Auto-scaling adjustments, security enforcement updates, API rate limiting, traffic routing, anomaly detection-based failover handling.	SREs override scaling policies, security teams handle breaches, compliance teams adjust regulatory rules.
Control Type	Code-Embedded Runtime Control	Dynamic Infrastructure & System Control	Manual Oversight & Policy-Based Control
Primary Role	Ensures non-functional properties such as performance, consistency, and security within the codebase.	Ensures availability, scalability, and security by dynamically adjusting infrastructure configurations.	Ensures business continuity, regulatory compliance, and governance when automation is insufficient.
Feedback Loops	Reads application-specific signals (e.g., failed transactions, retry limit breaches, latency anomalies).	Reads system-wide metrics (e.g., CPU/memory usage, service health, security anomaly detection).	Reads business and operational insights (e.g., customer complaints, regulatory audits, financial reports).
Emergent Properties Controlled	Performance, fault tolerance, consistency, security, efficiency.	Availability, scalability, resilience, security, regulatory compliance.	Compliance, risk management, business continuity, major service recovery.

# Emergent properties

Emergent Property	Description	Emergence	STAMP Perspective
<b>System Availability and Reliability</b>	Ability of the platform to remain operational under varying loads.	Interaction between infrastructure scaling, load balancing, and recovery mechanisms.	Hazards: Service unavailability due to insufficient scaling or unbalanced traffic. Constraints: Dynamic resource allocation and traffic distribution.
<b>Data Consistency</b>	Accuracy and uniformity of data across distributed systems (e.g., inventory, pricing).	Synchronization across databases, caches, and APIs under varying operational conditions.	Hazards: Inconsistent inventory data causing over-selling or underselling. Constraints: Synchronization mechanisms must ensure consistent updates.
<b>Security and Privacy</b>	Protection of sensitive customer and business data from unauthorized access or misuse.	Integration of authentication systems, access controls, and encryption mechanisms.	Hazards: Unauthorized access due to weak or misconfigured security policies. Constraints: Enforce least privilege access and detect unauthorized attempts.

# Emergent properties

Emergent Property	Description	Emergence	STAMP Perspective
<b>Customer Experience</b>	Responsiveness, speed, and quality of interactions for customers on the platform.	Backend processing, frontend rendering, and network latency interactions.	Hazards: High latency or downtime leading to abandoned carts and dissatisfaction. Constraints: Meet acceptable response times and prioritize key actions.
<b>Compliance with Regulations</b>	Adherence to legal and regulatory requirements (e.g., GDPR, PCI-DSS).	Interaction of data handling processes, audit trails, and reporting mechanisms.	Hazards: Non-compliance due to incomplete audit trails or data mishandling. Constraints: Enforce data retention policies and maintain complete audit logs.
<b>Scalability</b>	System's ability to handle increased workloads by scaling resources.	Coordination of load balancers, infrastructure monitoring, and resource provisioning.	Hazards: Insufficient scaling during peak events causing service degradation. Constraints: Detect load increases early and scale resources accordingly.

# Top 5 Findings from STPA

Finding (via STPA)	Impact If STPA Were Not Applied	Why Conventional Techniques Failed
<b>Unsafe control logic during availability zone failover</b>	Prolonged outages due to incorrect failover sequencing; longer MTTR	Chaos testing validated recovery behavior but missed control path flaws
<b>Config drift across environments created unsafe states</b>	Persistent config-related incidents; risk of cascading failures during deployment	Monitoring lacked insight into timing/version mismatches
<b>Operator actions occasionally worsened system state under pressure</b>	Higher error rate during incidents; inconsistent or incorrect manual recovery	Postmortems focused on outcome, not flawed decision points
<b>Broken or misaligned feedback loops between components</b>	Delayed detection of issues; increased customer-visible degradation	Observability showed metrics, not missing or delayed signal paths
<b>Flawed assumptions in auto-scaling thresholds under load</b>	Overloaded services, customer churn during spikes; increased cart abandonment	Performance tests didn't simulate coupled feedback across tiers
<b>Fraud detection model failing to adapt to new fraud patterns.</b>	Financial	Feedback loops not covered

# Benefits

Metric	% Improvement
MTTR (Mean Time to Recovery)	32% faster
Incident Rate (Config-Linked)	66% reduction
Detection Latency (Control Gaps)	50% faster
Operator Recovery Error Rate	66% reduction
Customer Drop-off During Failures	51% lower

## Sources & References

- [1] *2024 CrowdStrike-related IT Outages*, Wikipedia – [https://en.wikipedia.org/wiki/2024\\_CrowdStrike-related\\_IT\\_outages](https://en.wikipedia.org/wiki/2024_CrowdStrike-related_IT_outages)
- [2] *Top 10 IT Outages in 2024–2025*, The Insurer – <https://www.theinsurer.com/ti/viewpoint/top-10-it-outages-in-2024-2025-01-24/>
- [3] *Microsoft 365 Down*, Reuters – <https://www.reuters.com/technology/microsoft-365-down-thousands-users-downdetector-shows-2024-09-12/>
- [4] *Microsoft Outage Hits Outlook, Teams and Office 365*, NY Post – <https://nypost.com/2024/10/10/business/microsoft-outage-knocks-out-outlook-teams-and-365/>
- [5] *Microsoft Outlook Suffers Global Outage*, Business Insider – <https://www.businessinsider.com/microsoft-outlook-suffers-global-outage-2025-3>