

Table of Contents

Preface.....	xiii
--------------	------

Introduction: Birth of Chaos.....	xvii
-----------------------------------	------

Part I. Setting the Stage

1. Encountering Complex Systems.....	1
Contemplating Complexity	1
Encountering Complexity	3
Example 1: Mismatch Between Business Logic and Application Logic	3
Example 2: Customer-Induced Retry Storm	6
Example 3: Holiday Code Freeze	10
Confronting Complexity	13
Accidental Complexity	13
Essential Complexity	14
Embracing Complexity	16
2. Navigating Complex Systems.....	17
Dynamic Safety Model	17
Economics	18
Workload	18
Safety	19
Economic Pillars of Complexity	20
State	21
Relationships	21
Environment	21
Reversibility	22

Economic Pillars of Complexity Applied to Software	22
The Systemic Perspective	23

3. Overview of Principles.....	25
What Chaos Engineering Is	25
Experimentation Versus Testing	26
Verification Versus Validation	27
What Chaos Engineering Is Not	28
Breaking Stuff	28
Antifragility	28
Advanced Principles	29
Build a Hypothesis Around Steady-State Behavior	30
Vary Real-World Events	30
Run Experiments in Production	32
Automate Experiments to Run Continuously	32
Minimize Blast Radius	33
The Future of “The Principles”	35

Part II. Principles in Action

4. Slack’s Disasterpiece Theater.....	39
Retrofitting Chaos	40
Design Patterns Common in Older Systems	40
Design Patterns Common in Newer Systems	41
Getting to Basic Fault Tolerance	41
Disasterpiece Theater	42
Goals	42
Anti-Goals	43
The Process	44
Preparation	44
The Exercise	46
Debriefing	49
How the Process Has Evolved	50
Getting Management Buy-In	51
Results	51
Avoid Cache Inconsistency	51
Try, Try Again (for Safety)	52
Impossibility Result	52
Conclusion	53

5. Google DiRT: Disaster Recovery Testing.....	55
Life of a DiRT Test	57
The Rules of Engagement	58
What to Test	61
How to Test	69
Gathering Results	71
Scope of Tests at Google	71
Conclusion	75
6. Microsoft Variation and Prioritization of Experiments.....	77
Why Is Everything So Complicated?	77
An Example of Unexpected Complications	78
A Simple System Is the Tip of the Iceberg	79
Categories of Experiment Outcomes	80
Known Events/Unexpected Consequences	80
Unknown Events/Unexpected Consequences	82
Prioritization of Failures	83
Explore Dependencies	84
Degree of Variation	84
Varying Failures	85
Combining Variation and Prioritization	86
Expanding Variation to Dependencies	87
Deploying Experiments at Scale	87
Conclusion	89
7. LinkedIn Being Mindful of Members.....	91
Learning from Disaster	92
Granularly Targeting Experiments	93
Experimenting at Scale, Safely	95
In Practice: LinkedOut	96
Failure Modes	97
Using LiX to Target Experiments	98
Browser Extension for Rapid Experimentation	101
Automated Experimentation	104
Conclusion	105
8. Capital One Adoption and Evolution of Chaos Engineering.....	107
A Capital One Case Study	108
Blind Resiliency Testing	108
Transition to Chaos Engineering	109
Chaos Experiments in CI/CD	110
Things to Watch Out for While Designing the Experiment	111

Tooling	112
Team Structure	113
Evangelism	114
Conclusion	115

Part III. Human Factors

9. Creating Foresight..... 119

Chaos Engineering and Resilience	120
Steps of the Chaos Engineering Cycle	120
Designing the Experiment	121
Tool Support for Chaos Experiment Design	122
Effectively Partnering Internally	125
Understand Operating Procedures	125
Discuss Scope	127
Hypothesize	128
Conclusion	130

10. Humanistic Chaos..... 133

Humans in the System	133
Putting the “Socio” in Sociotechnical Systems	134
Organizations Are a System of Systems	134
Engineering Adaptive Capacity	135
Spotting Weak Signals	135
Failure and Success, Two Sides of the Same Coin	136
Putting the Principles into Practice	137
Build a Hypothesis	137
Vary Real-World Events	138
Minimize the Blast Radius	138
Case Study 1: Gaming Your Game Days	139
Communication: The Network Latency of Any Organization	141
Case Study 2: Connecting the Dots	142
Leadership Is an Emergent Property of the System	145
Case Study 3: Changing a Basic Assumption	146
Safely Organizing the Chaos	148
All You Need Is Altitude and a Direction	149
Close the Loops	149
If You’re Not Failing, You’re Not Learning	150

11. People in the Loop..... 151

The Why, How, and When of Experiments	152
---------------------------------------	-----

The Why	152
The How	153
The When	154
Functional Allocation, or Humans-Are-Better-At/Machines-Are-Better-At	155
The Substitution Myth	157
Conclusion	159
12. The Experiment Selection Problem (and a Solution).....	161
Choosing Experiments	161
Random Search	162
The Age of the Experts	163
Observability: The Opportunity	168
Observability for Intuition Engineering	169
Conclusion	171

Part IV. Business Factors

13. ROI of Chaos Engineering.....	175
Ephemeral Nature of Incident Reduction	175
Kirkpatrick Model	176
Level 1: Reaction	176
Level 2: Learning	177
Level 3: Transfer	177
Level 4: Results	178
Alternative ROI Example	178
Collateral ROI	179
Conclusion	181
14. Open Minds, Open Science, and Open Chaos.....	183
Collaborative Mindsets	183
Open Science; Open Source	185
Open Chaos Experiments	186
Experiment Findings, Shareable Results	188
Conclusion	188
15. Chaos Maturity Model.....	191
Adoption	192
Who Bought into Chaos Engineering	192
How Much of the Organization Participates in Chaos Engineering	193
Prerequisites	194
Obstacles to Adoption	195

Sophistication	196
Putting It All Together	201

Part V. Evolution

16. Continuous Verification.....	205
Where CV Comes From	205
Types of CV Systems	207
CV in the Wild: ChAP	209
ChAP: Selecting Experiments	209
ChAP: Running Experiments	210
The Advanced Principles in ChAP	210
ChAP as Continuous Verification	211
CV Coming Soon to a System Near You	211
Performance Testing	212
Data Artifacts	212
Correctness	212
17. Let's Get Cyber-Physical.....	215
The Rise of Cyber-Physical Systems	216
Functional Safety Meets Chaos Engineering	217
FMEA and Chaos Engineering	219
Software in Cyber-Physical Systems	219
Chaos Engineering as a Step Beyond FMEA	221
Probe Effect	224
Addressing the Probe Effect	225
Conclusion	226
18. HOP Meets Chaos Engineering.....	229
What Is Human and Organizational Performance (HOP)?	229
Key Principles of HOP	230
Principle 1: Error Is Normal	230
Principle 2: Blame Fixes Nothing	231
Principle 3: Context Drives Behavior	231
Principle 4: Learning and Improving Is Vital	232
Principle 5: Intentional Response Matters	232
HOP Meets Chaos Engineering	232
Chaos Engineering and HOP in Practice	234
Conclusion	235

19. Chaos Engineering on a Database.....	237
Why Do We Need Chaos Engineering?	237
Robustness and Stability	237
A Real-World Example	238
Applying Chaos Engineering	240
Our Way of Embracing Chaos	240
Fault Injection	241
Fault Injection in Applications	241
Fault Injection in CPU and Memory	242
Fault Injection in the Network	242
Fault Injection in the Filesystem	243
Detecting Failures	244
Automating Chaos	245
Automated Experimentation Platform: Schrodinger	245
Schrodinger Workflow	247
Conclusion	247
20. The Case for Security Chaos Engineering.....	249
A Modern Approach to Security	250
Human Factors and Failure	250
Remove the Low-Hanging Fruit	252
Feedback Loops	253
Security Chaos Engineering and Current Methods	254
Problems with Red Teaming	255
Problems with Purple Teaming	255
Benefits of Security Chaos Engineering	255
Security Game Days	256
Example Security Chaos Engineering Tool: ChaoSlingr	257
The Story of ChaoSlingr	257
Conclusion	260
Contributors/Reviewers	260
21. Conclusion.....	263
Index.....	267