
Emergency Response Coordination System¹

Lee Wall

Keaton Lee

With
Apart Research

Abstract

Emergency response to catastrophic events is often bottlenecked by institutional coordination. The 2019 Crimson Contagion exercise revealed that unclear roles, inefficient resource request processes, and serial task execution significantly delayed pandemic response. As AI capabilities accelerate the pace of potential catastrophes, traditional coordination mechanisms become unacceptably slow. We present the Emergency Response Coordination System (ERCS), an AI-driven platform that centralizes incident information, explicitly tracks task dependencies, and surfaces opportunities for parallelization in emergency response. The system integrates natural language chat interfaces for coordinators with automated email notifications for distributed responders, reducing cognitive overhead during high-stress incidents. We outline two tracks of a novel pathogen detection scenario based on the Nucleic Acid Observatory's capabilities where we demonstrate how this system could potentially reduce the response time in a multi-institution response effort to a catastrophe by enabling parallel execution of non-dependent tasks and maintaining situational awareness across institutions. The system's event-sourced architecture provides full auditability and version control, critical for post-incident analysis and adversarial robustness. This serves as a proof-of-concept that coordination technology can meaningfully accelerate institutional response to AI-enabled catastrophic risks.

Keywords: Coordination tech, disaster response, catastrophic risks, biosecurity, pandemic preparedness, cybersecurity, warning shots

¹ Work conducted at the [Apart def/acc Hackathon](#), 2025

1. Introduction

Historically, responses to emergencies have been severely bottlenecked by institution response times. As AI capabilities increase, though, the *pace* of potential disasters will accelerate.

For instance, given the precedent set by the COVID-19 pandemic response, it appears likely that a pathogen engineered with the assistance of AI to spread as fast as possible or as stealthily as possible could infect so many people in so little time that traditional emergency response becomes unacceptably slow. Similarly, a large-scale cyber intrusion assisted through AI (which, notably, has already happened with current AI systems) could happen too fast for institutions to respond before it is too late. These are risks we cannot afford to take.

The good news is, as anyone experienced with bureaucracy can intuit, emergency response coordination is nowhere close to optimal. The time lag between someone noticing a suspicious string of nucleotides that's been getting exponentially more frequent in wastewater samples and countermeasures being developed or organizations preparing for the logistics of PPE distribution is probably much longer than we'd like to think in addition to being uncomfortably inefficient – reliant on phone calls, emails, the pace of human reading, understanding, and information transfer while the pathogen spreads unabated.

What actually slows these responses down? The 2019 Crimson Contagion exercise found that although some bottlenecks were physical or financial – insufficient production capacity for protective equipment and medical devices, insufficient federal funding – some were frustratingly informational: inefficient resource request processes for states, unclear roles in the crisis, and whether something had been formally rubber-stamped as a disaster or emergency.

These are not problems with a technological silver bullet, but they are amenable to some of the current strengths of AI systems. Information consolidation, compression, and dissemination are precisely where large language models excel, creating an opportunity for slow institutions to adopt tools today that will streamline their emergency response to future threats. These models can standardize forms, assign roles in a crisis and make these roles widely known and well-defined, and summarize lengthy reports to save time for decisionmakers.

Furthermore, emergency responses are implicitly bottlenecked by a familiar inefficiency: unnecessarily serial tasks that could be parallelized. In any given response, while one person is waiting on the go-ahead from someone else, waiting on a form or a lab result, it would be more optimal for them to *begin the tasks that are not conditional on these things*. At an individual level, there are likely variations in personal efficiency in triaging tasks according to how well they can be parallelized, but this creates an opportunity for AI tools to surface opportunities for parallelization that make these responses intrinsically more efficient. At a multi-institutional scale, it is at best unmanageable for one person to handle this

task management due to the mental overhead of complex coordination, and at worst impossible when the necessary information is scattered across institutions.

The goal of this project is to bite off a piece of this problem by creating an AI-driven system that manages all the information relevant to an incident, centralizing it for emergency responders and coordinators, assigning clear roles, defining tasks and their dependencies according to user input, highlighting the most action-relevant information, and surfacing opportunities to parallelize work to speed up a response.

The system will have two main components: a barebones UI with a chatbot and a dashboard that enables users to get up to speed on an incident, and an emailing system that serves to both summarize information about the incident and gather information via YES/NO questions (e.g. “Can you do task A before Organization X finishes task B?”).

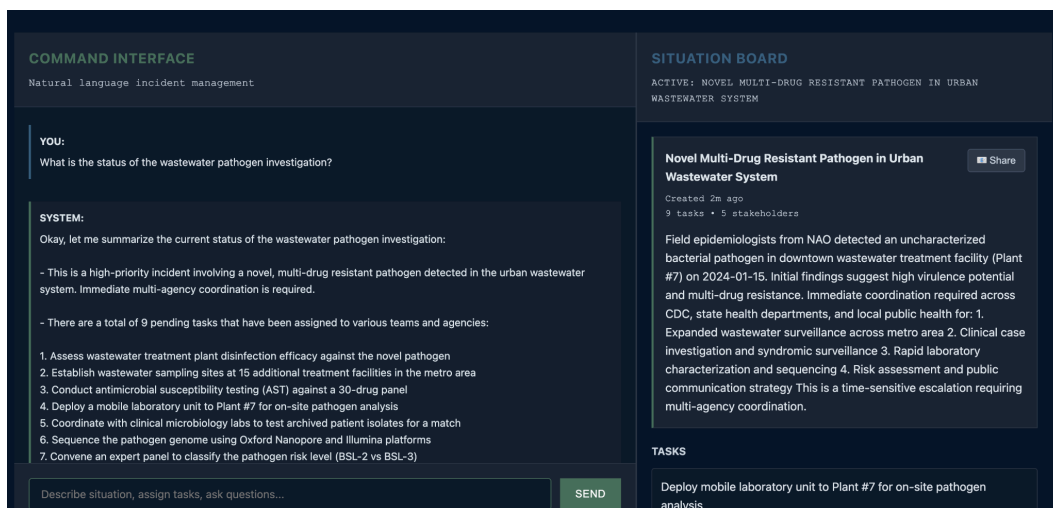
2. Results

System Design

We implemented a working prototype of ERCS with the following capabilities:

- Event-sourced incident state management with full audit trail
- Natural language chat interface for incident coordinators
- Automated email digest and YES/NO question system for distributed responders
- Explicit task dependency tracking with automatic blocker detection
- Real-time status updates and notifications

Mock Scenario To evaluate ERCS's potential impact, we mocked up a novel pathogen detection scenario with Claude Sonnet 3.5, based on the Nucleic Acid Observatory's surveillance capabilities and the coordination challenges identified in the Crimson Contagion exercise.



Mock Scenario: Engineered Pandemic Pathogen

Scenario Setup

T=0 (Monday, 3:47 AM EST): NAO's automated analysis flags an anomaly in wastewater samples from San Francisco, Los Angeles, and Seattle. A novel RNA sequence is appearing at exponentially increasing concentrations (doubling every ~18 hours). BLAST search returns no matches to known pathogens. The sequence has features consistent with respiratory viruses.

The junior bioinformatician on night shift, Maria Chen, sees the alert.

Scenario A: Traditional Response (No ERCS)

Hour 0-8: Detection and Internal Deliberation

3:47 AM: Maria reviews the data. It looks real, but she's junior and doesn't want to cause a false alarm. She emails her supervisor, Dr. James Park.

7:30 AM: Dr. Park arrives at work, sees the email buried under 40 others. He asks Maria to re-run the analysis and check for contamination.

9:45 AM: Maria confirms the finding. Dr. Park decides this needs to go up the chain and calls NAO's director, Dr. Sarah Williams. She's in a meeting; he leaves a voicemail.

11:30 AM: Dr. Williams calls back. They discuss for 20 minutes. She agrees it's concerning but wants independent confirmation before alerting CDC. "We can't cry wolf again after the false positive last year."

12:00 PM: She emails a university lab contact to request verification. The professor is teaching and doesn't see the email until later. His lab can do it, but needs samples shipped overnight—they can start Wednesday morning.

2:30 PM: Dr. Williams makes a judgment call: this is spreading too fast to wait 36 hours. She decides to alert CDC while continuing verification in parallel.

Hour 8-24: First CDC Contact and Bureaucratic Activation

3:00 PM: Dr. Williams calls the CDC Emergency Operations Center. Gets a duty officer who takes notes and says someone will call back.

4:30 PM: CDC epidemiologist Dr. Karen Martinez calls back, takes detailed notes, agrees it sounds serious but needs to brief her supervisor.

5:45 PM: Dr. Martinez is pulled into an existing flu meeting. The NAO issue will be discussed after.

7:00 PM: Meeting ends. Dr. Martinez briefs the Emergency Response Coordinator, Dr. Robert Chen. He wants to see the raw data and needs to determine if this meets criteria for activating the emergency response framework.

Tuesday, 8:00 AM: Dr. Chen reviews the data and convenes a working group for 2 PM.

2:00 PM: Working group meets (8 people). Consensus: activate preliminary investigation, but no formal emergency declaration yet—insufficient data. They decide on next steps: CDC will send a team to collect independent samples, assign genomic sequencing (24-48 hour turnaround), draft communication plan (but don't execute until confirmed).

Hour 24-60: Sequential Task Execution

Wednesday, 10:00 AM: CDC team arrives in San Francisco, collects samples, ships them to Atlanta.

Thursday, 8:00 AM: CDC lab receives samples and begins sequencing.

Friday, 2:00 PM (T+60 hours): Sequencing complete. Matches NAO findings. Novel pathogen confirmed.

Friday, 3:00 PM: Dr. Chen briefs HHS leadership. They activate emergency response framework.

Friday, 4:00 PM: First formal coordination meeting with 30 people on Zoom from CDC, HHS, ASPR, NIH, FDA. The meeting spends 20 minutes just figuring out who's in charge of what, what the priorities are, and what resources are available.

Friday, 5:30 PM: Decisions made: Epidemiology team starts case investigation (but it's the weekend), lab team begins characterization, medical countermeasures team starts thinking about therapeutics/vaccines (but needs more pathogen data first), ASPR will assess national stockpile (but it's Friday evening).

Weekend: Skeleton crews. Most work waits until Monday.

Monday, 9:00 AM (T+102 hours): Full teams back at work. The pathogen has been doubling every 18 hours for 102 hours—that's 5.7 doublings. **If it started at detectable levels, it's now ~50x more prevalent.**

Key Bottlenecks

- **Verification paralysis:** 12 hours lost to internal NAO deliberation about false alarm risk
- **Communication latency:** Voicemails, email chains, phone tag = ~8 hours
- **Bureaucratic activation:** Determining if this meets emergency criteria = 24 hours
- **Sequential dependencies:** Couldn't start countermeasures until sequencing confirmed, couldn't mobilize PPE until emergency declared
- **Weekend effect:** 48 hours of reduced capacity at critical moment
- **Unclear ownership:** First coordination meeting spent significant time on role clarification

By Monday, T+102 hours: Pathogen confirmed and characterized, epidemiological investigation beginning, countermeasures work beginning, PPE mobilization beginning. **~4 days passed with pathogen spreading unchecked.**

Scenario B: With ERCS

Hour 0-4: Instant Incident Creation and Parallel Awareness

3:47 AM: Maria opens ERCS chat and types: "Novel RNA sequence in SF/LA/Seattle wastewater. Exponential growth (~18hr doubling). No BLAST matches. Respiratory virus features."

ERCS creates Incident #2024-0847 and responds: "Created incident. I've notified your supervisor and CDC EOC (monitoring mode). Can you verify this isn't contamination?"

3:49 AM: Automated emails go to Dr. Park ("Your team flagged potential novel pathogen—status: unconfirmed") and CDC EOC ("FYI - NAO reported possible pathogen, unconfirmed").

9:45 AM: Maria confirms verification in chat. ERCS updates incident status and notifies Dr. Williams: "Verification complete. Should we notify CDC immediately or wait for independent confirmation?"

Hour 4-6: Rapid Decision-Making and Parallelization

10:00 AM: Dr. Williams opens the ERCS chat: "What's the cost of waiting?"

ERCS: "Each 18-hour delay = 2x more pathogen. Independent confirmation takes ~36 hours = 4x more spread. However, early notification risks false alarm."

Williams: "Notify CDC immediately. Get independent confirmation in parallel."

10:01 AM: Dr. Chen (CDC Emergency Response Coordinator) receives email: "YOU ARE INCIDENT COMMANDER - Novel pathogen alert." He opens the chat.

Chen: "Show me what we know."

ERCS provides complete summary: timeline, facts, current status, independent confirmation in progress (ETA Wed 2 PM).

Chen: "What can we start right now without waiting for confirmation?"

ERCS: "Three tasks: (1) Preliminary epi investigation, (2) PPE stockpile assessment, (3) Draft public communication. All are low-cost and reversible if false alarm."

Chen: "Start epi and PPE. Karen Martinez for epi, Jamie Liu for PPE."

10:16 AM: Automated task assignment emails go out with "ACCEPTED/BLOCKED/explain" response options.

Hour 6-34: True Parallel Execution

10:30 AM: Dr. Martinez accepts and opens chat: "Do we know attack rate yet?"

ERCS: "No. That comes from pathogen characterization (Thursday). Start with broad syndromic surveillance—you can narrow later."

11:00 AM: Jamie Liu completes stockpile assessment via chat: "150M N95s, sufficient for 3-metro response. If national, need manufacturers (3-week lead time). Should I put them on standby?"

ERCS routes to Dr. Chen, who decides: "Wait for confirmation Wednesday, but prepare paperwork."

Tuesday-Wednesday: Multiple work streams progress in parallel. Dr. Martinez identifies 101 suspicious cases. Jamie prepares distribution plans. CDC lab works on confirmation. NIH team is briefed and ready to start countermeasures.

Wednesday, 2:00 PM (T+34 hours): CDC lab director posts in chat: "Sequencing complete. Confirms NAO findings. Starting characterization now."

2:01 PM: ERCS notifies all actors via email: "CONFIRMED: Novel pathogen verified." In Dr. Chen's chat: "Three decisions needed: (1) Approve PPE distribution? (2) Assign countermeasures to NIH? (3) Release public communication?"

Chen: "Approve PPE and countermeasures. Show me the comms draft first. Also schedule coordination meeting for 3 PM—include CDC, ASPR, NIH, HHS."

2:02 PM: Automated emails execute decisions and send meeting invites.

Hour 35-53: Coordination with Shared Context

3:00 PM: Coordination meeting. Dr. Chen shares his screen showing ERCS and types: "Give me a summary of where we are."

ERCS provides complete status: what's completed (detection, verification, confirmation, preliminary epi, PPE assessment), what's in progress (characterization ETA Thu 9 AM, PPE distribution, countermeasures starting), critical path (waiting on characterization), no blockers.

The meeting takes 15 minutes instead of an hour because everyone has context.

Thursday, 9:00 AM (T+53 hours): Lab director posts: "Characterization complete. R0 ~3.5 (highly transmissible), 5% hospitalization rate, 3-5 day incubation."

9:01 AM: ERCS sends targeted email questions to task owners: "Dr. Martinez: Given R0=3.5, expand surveillance to neighboring regions? Y/N" and "Jamie Liu: Given R0=3.5, activate manufacturers now? Y/N"

Both respond affirmatively. Dr. Chen approves via chat. ERCS releases public communication.

The pathogen has been doubling every 18 hours for 53 hours. That's 2.9 doublings. If it started at detectable levels, it's now ~7.5x more prevalent (compared to 50x in traditional response).

Key Improvements

By Thursday morning, T+53 hours: Pathogen confirmed and fully characterized, PPE already distributing, countermeasures work well underway (2-day head start), public notified with accurate information, epi team refining case counts. **~2 days passed vs. 4+ days in traditional response.**

Bottleneck	Traditional	With ERCS
Internal verification	12 hours	6 hours
CDC notification	8 hours	2 hours
Awareness propagation	24+ hours	Minutes
Task parallelization	Almost none	Extensive

Coordination overhead	High (20min meetings on roles)	Low (clear from template)
Decision latency	Hours (need meetings)	Minutes (full context immediately)

Where ERCS Helped

- **Explicit task dependencies** → people know what can start immediately vs. what must wait
- **Centralized state** → no "what do we know?" questions
- **Automated notifications** → information reaches right people immediately
- **Clear ownership** → templates pre-assign roles
- **Audit trail** → every decision logged, reducing false alarm fears
- **Parallelization visibility** → agents surface opportunities humans miss

ERCS cut response time from ~100 hours to ~50 hours, buying the world critical time when pathogen spread is exponential.

3. References

- U.S. Department of Health and Human Services. (2020). Crimson Contagion 2019 https://www.governmentattic.org/docs/HHS-CrimsonContAfterActionRpt_2020.pdf
- Nucleic Acid Observatory Consortium. (2021). A Global Nucleic Acid Observatory for Biodefense and Planetary Health. arXiv. <https://arxiv.org/abs/2108.02678>