Agentic Simulacra for Synthetic Construction Management Data Generation

Vincil C. Bishop III
Amazon Web Services
vincilb@amazon.com

Nivedha Balakrishnan
Amazon Web Services
nivibkr@amazon.com

Saeideh Shahrokh Esfahani
Amazon Web Services
saeidesh@amazon.com

Abstract

Construction management systems require realistic test data capturing complex stakeholder interactions and temporal dependencies, yet accessing real project data remains challenging due to privacy constraints and proprietary information protection. This research addresses a critical systems engineering challenge by introducing agentic simulacra patterns that leverage multi-agent coordination to generate synthetic datasets maintaining authentic domain semantics and workflow dependencies. The approach employs specialized construction professional agents executing coordinated processes across project lifecycle phases, with each agent embodying distinct roles, expertise, and decision-making patterns observed in actual construction projects. Experimental validation compares foundation model capabilities across ten diverse construction scenarios spanning residential, commercial, industrial, and infrastructure project types. This work establishes reusable patterns for synthetic data generation in complex sociotechnical systems where traditional approaches fail to capture emergent complexity arising from stakeholder interactions, with applications extending to healthcare, financial services, and supply chain management domains requiring correlated synthetic data for systems validation.

1 Introduction

Systems engineering for construction management software requires comprehensive test data representing complex project interdependencies. Construction management systems must handle intricate workflows involving multiple stakeholders, temporal dependencies, regulatory compliance, and dynamic conditions. Traditional approaches rely on simplified test datasets lacking real-world complexity or anonymized production data that remains inaccessible due to privacy constraints. Current synthetic data generation produces statistically valid but relationally disconnected datasets inadequately representing authentic construction workflows. Random generation creates structurally correct records but lacks semantic coherence and temporal dependencies, while rule-based systems struggle to capture emergent complexity from stakeholder interactions. This research introduces agentic simulacra patterns leveraging multi-agent systems where specialized construction professional agents execute coordinated processes to generate synthetic datasets maintaining authentic relational patterns, temporal dependencies, and stakeholder interaction dynamics suitable for comprehensive systems engineering validation.

1.1 Problem Definition

Construction management systems engineering requires synthetic datasets exhibiting three critical characteristics. Semantic coherence demands authentic construction terminology, appropriate technical specifications, and domain-consistent relationships between project artifacts. Temporal dependencies require synthetic data respecting sequential construction phases with explicit handoff requirements and approval gates. Stakeholder interaction patterns must capture complex coordination between professional roles with distinct responsibilities, communication patterns, and decision-making authorities to enable realistic systems testing.

2 Background and Related Work

2.1 Synthetic Data Generation in Systems Engineering

Traditional synthetic data generation approaches in systems engineering focus primarily on statistical validity and structural correctness. Random data generation produces datasets matching specified schemas and distributions but lacks semantic coherence and relational consistency. Rule-based systems can enforce constraints and dependencies but require extensive domain knowledge encoding and struggle with emergent complexity arising from stakeholder interactions.

Recent advances in large language models (LLMs) have enabled more sophisticated synthetic data generation approaches. Studies demonstrate that LLM-generated data can effectively support model training and system testing, with performance influenced by task complexity and domain specificity [1]. The TarGEN framework extends these capabilities through multi-step prompting strategies and self-correction mechanisms, enabling high-quality synthetic data generation with improved diversity and reduced bias [2]. Similarly, TrueTeacher produces multilingual synthetic datasets through iterative refinement processes, demonstrating robustness across domain variations [3].

However, these approaches remain limited for complex sociotechnical domains requiring coordinated stakeholder interactions and temporal workflow dependencies. Construction management systems exemplify such domains, where data generation must capture not only structural correctness but also authentic professional coordination patterns and domain-specific semantic relationships.

2.2 Multi-Agent Systems for Data Generation

Advances in generative artificial intelligence, particularly foundation models and multi-agent systems, present new opportunities for addressing complex synthetic data generation challenges. Park et al. (2023) demonstrated that generative agents built on large language models can maintain consistent personas, execute goal-directed behaviors, and coordinate through multi-agent interactions in simulated environments [4]. Their architecture incorporates memory systems enabling agents to recall past experiences, reflection mechanisms supporting higher-level reasoning about observations, and planning capabilities allowing agents to translate reflections into coherent action sequences.

These capabilities suggest potential applications beyond social simulation to structured data generation for enterprise systems where authentic stakeholder coordination patterns and temporal dependencies are essential. Multi-agent approaches can model complex professional interactions, maintain consistency across extended workflows, and generate correlated datasets reflecting realistic organizational dynamics.

2.3 Agentic Approaches to Synthetic Data Generation

Recent work demonstrates the effectiveness of agentic workflows for synthetic data generation across diverse domains. OrchDAG introduces directed acyclic graph representations for tool orchestration in multi-turn interactions, employing iterative refinement through feedback loops to improve generation quality [5]. This approach validates that explicit dependency modeling and iterative refinement can enhance synthetic data generation, though OrchDAG focuses on tool coordination rather than domain-specific data generation. TAGAL extends agentic methods to tabular data generation, comparing training-free LLM approaches against trained generative models and demonstrating that agentic feedback loops achieve performance comparable to models requiring fine-tuning [6]. These findings support the viability of training-free agentic approaches for synthetic data generation, though TAGAL addresses structured tabular data rather than complex relational datasets with temporal dependencies.

2.4 Validation Challenges in Synthetic Data

Critical concerns regarding synthetic data validity have emerged from recent empirical studies. Ivetta et al. demonstrate that LLM-generated synthetic social survey data exhibits systematic biases exceeding sampling biases in human surveys, with 94.4% of LLM responses differing significantly from human benchmarks [7]. Their findings reveal that even small human samples (n=2) outperform LLMs in representing population characteristics, raising fundamental questions about synthetic data validity when ground truth is unavailable for validation. This contrasts with our approach where synthetic data serves systems engineering validation rather than population representation, though the findings underscore the importance of rigorous validation methodologies. Mannekote et al. investigate belief-behavior consistency in LLM-based simulations, revealing that foundation model capabilities substantially affect simulation fidelity, with correlation metrics varying from 0.48 to 0.96 depending on elicitation strategy [8]. These results validate our comparative evaluation of foundation models while highlighting that prompt engineering and elicitation strategies critically influence generation quality.

2.5 Domain-Specific Multi-Agent Simulation

Multi-agent simulation frameworks have demonstrated effectiveness in domain-specific contexts requiring professional coordination. Yue et al. present a legal consultation simulator employing supervisor-based quality control where a monitoring agent provides real-time refinement suggestions during lawyer-client interactions [9]. This supervisor-mediated approach achieves higher quality outputs than post-hoc evaluation, suggesting that real-time quality control mechanisms could enhance construction data generation. However, the legal framework addresses two-agent interactions with clear roles, while construction management involves substantially more complex multi-agent coordination. Jayashankar and Balan compare statistical methods, agent-based models, and GANs for employee behavior data generation, demonstrating that correlation preservation and distribution matching provide quantitative validation of synthetic data quality [10]. Their emphasis on statistical validation complements our qualitative evaluation approach, though employee behavior metrics exhibit simpler relational structures than construction management workflows.

3 Methodology

3.1 Agentic Simulacra Framework Design

Our methodology employs a hierarchical multi-agent architecture organized around sequential construction project phases reflecting industry-standard delivery methods. This phase-based structure establishes temporal boundaries and information flow patterns mirroring real construction lifecycles, ensuring synthetic data generation follows authentic sequencing and dependency relationships.

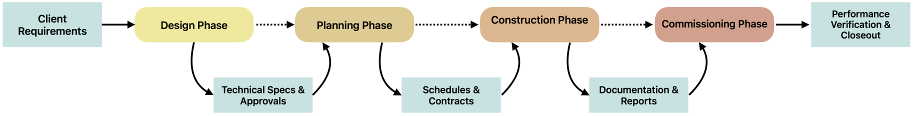

Figure 1: System Architecture Overview - Hierarchical multi-agent structure with four main phases (Design, Planning, Construction, Commissioning) showing agent interactions and data flow between phases.

The framework decomposes construction projects into four distinct phases based on comprehensive process mapping across disparate construction phases and processes:

Design Phase: Transforms client requirements into technical solutions through 9 processes including requirements analysis, site analysis, conceptual design, design development, specification writing, code compliance review, design coordination, cost estimation, and regulatory submission. Agents include architect, site engineer, structural engineer, MEP engineer, specification writer, code consultant, estimator, and regulatory specialist.

Planning Phase: Converts approved designs into executable strategies through 8 processes including construction scheduling, procurement planning, cost planning, contract development, site logistics planning, safety planning, quality planning, and risk management. Agents include scheduler, procurement specialist, contract manager, construction manager, safety manager, and quality manager.

Construction Phase: Coordinates daily field activities through 13 processes including daily operations management, material management, quality control, safety management, progress monitoring, change management, communication management, issue resolution, documentation management, time management, invoicing management, submittal management, and punch list management. Agents include construction supervisor, material coordinator, quality control inspector, safety officer, change manager, document controller, timekeeper, accounting specialist, submittal coordinator, and punch list coordinator.

Commissioning Phase: Verifies system performance and ensures operational readiness through 6 processes including system testing, documentation review, training delivery, performance verification, warranty processing, and project closeout. Agents include commissioning agent, documentation specialist, training coordinator, system technician, and warranty coordinator.

3.2 Agent Design and Specialization

Specialized construction professional agents populate the framework, embodying distinct roles, expertise, and responsibilities observed in actual projects. Each agent receives domain-specific system prompts defining professional knowledge, decision-making patterns, and communication styles characteristic of their roles, ensuring synthetic stakeholder interactions reflect authentic professional relationships and coordination dynamics. The complete framework employs 15 specialized agents across four construction phases, with detailed roles and responsibilities provided in Appendix A (Table 2).

3.3 Process Definition and Workflow Orchestration

Construction processes represent fundamental transformation functions within the framework. Each process consumes specific input objects and produces defined output objects, creating explicit dependency chains establishing information flow patterns. The framework implements 68 distinct construction processes mapped to specific agent actors and object types.

Process orchestration employs swarm-based coordination within each phase:

Phase-Based Swarms: Each project phase operates as an independent swarm with specialized agents executing coordinated processes. The framework employs 15 key agents distributed across phases with varying max_handoffs parameters to control coordination complexity.

Temporal Dependencies: Phase gates enforce sequential progression using expression-based conditions that verify completion criteria including agent execution success and minimum tool usage thresholds.

Input-Output Chains: Each agent process specifies required input objects and produces defined output objects, creating dependency graphs that mirror authentic construction information flow. For example, architects consume project specifications and produce documents, forms, meetings, drawings, and photo albums.

Execution Timeouts: Individual swarms operate with 900-1200 second timeouts, while the overall graph execution allows up to 7200 seconds with maximum 100 node executions to ensure completion within reasonable time bounds.

3.4 Data Structure and Correlation Mechanisms

The framework targets comprehensive coverage of construction management object types, generating 38 distinct data structures based on industry-standard construction management systems. Each object type adheres to JSON Schema definitions ensuring structural consistency while agents inject realistic construction terminology and contextually appropriate technical detail. The complete inventory of object types organized by functional category is provided in Appendix B (Table 3).

4 Implementation

4.1 Technical Architecture

The experimental implementation employs multi-agent orchestration frameworks supporting hierarchical coordination patterns and temporal dependency management. While our implementation uses the AWS Strands Agent SDK, the methodology is framework-agnostic and can be implemented using alternative multi-agent platforms, or custom orchestration systems.

Key technical components include:

Agent Orchestration: AWS Strands SDK multi-agent graph of swarms configuration manages hierarchical agent coordination across construction phases. Each phase operates as an independent swarm with specialized agents executing coordinated processes. Phase gates enforce sequential progression through expression-based conditions verifying completion criteria. Agents maintain persistent context enabling memory retention across process executions, with inter-agent communication coordinated through message queues.

Foundation Model Integration: Amazon Bedrock APIs provide access to Nova Lite and Nova Pro models with automatic retry mechanisms and rate limiting. The system employs prompt templating with role-specific instructions, few-shot examples, and dynamic context injection tailored to construction professional personas.

Data Persistence: Amazon S3 stores generated objects with correlation metadata enabling dependency tracking and lineage analysis. Object versioning supports audit requirements and temporal analysis of generated datasets.

4.2 Synthetic Project Scenarios

The experiment generates synthetic construction management datasets across ten diverse construction projects representing varied building types, scales, and complexity levels. Each project specification provides comprehensive details including physical characteristics, site conditions, budget parameters, schedule constraints, regulatory requirements, technical systems, stakeholder profiles, and risk factors. The complete list of project scenarios is provided in Appendix B (Table 4).

5 Experimental Design

5.1 Research Questions

This research addresses three primary research questions relevant to systems engineering practice:

RQ1: Can agentic simulacra generate synthetic construction management datasets that maintain semantic coherence, temporal dependencies, and stakeholder interaction patterns required for effective systems testing?

RQ2: How do foundation model capabilities influence synthetic data generation robustness, comprehensiveness, and quality for complex sociotechnical domains?

RQ3: What are the practical trade-offs between cost efficiency and generation reliability for production deployment of agentic synthetic data generation systems?

5.2 Experimental Variables

The experiment employs a controlled comparative design isolating foundation model performance as the primary experimental variable while maintaining identical orchestration frameworks and evaluation methodologies:

Independent Variable: Foundation model selection (Amazon Nova Lite vs. Nova Pro)

Controlled Variables: - Project specifications (10 identical projects) - Agent configurations (15 agents with identical system prompts) - Process definitions (68 construction processes) - Orchestration parameters (AWS Strands Agent SDK configuration) - Object schemas (51 JSON Schema definitions) - Evaluation criteria (structural, content, workflow, and domain metrics)

Dependent Variables: - Project completion rates and failure analysis - Data volume metrics (total records, object types, phase coverage) - Object generation patterns (distribution across categories) - Semantic quality scores (consistency, realism, relevance) - Temporal dependency maintenance and correlation integrity

5.3 Foundation Model Selection

The experiment compares two Amazon Nova foundation models representing different capability and cost tiers, though the framework supports any foundation model through configuration:

Amazon Nova Lite (us.amazon.nova-lite-v1:0): Cost-efficient baseline model optimized for computational efficiency with input tokens at $0.00006 per thousand and output tokens at $0.00024 per thousand. Designed for high-throughput applications prioritizing cost control.

Amazon Nova Pro (us.amazon.nova-pro-v1:0): Advanced model providing enhanced reasoning capabilities with input tokens at $0.0008 per thousand and output tokens at $0.0032 per thousand (13.3x cost difference). Designed for applications requiring sophisticated reasoning and high-quality outputs.

Both models receive identical project specifications, agent system prompts, and orchestration parameters, isolating model capability differences for comparative analysis. The framework's model-agnostic design supports alternative foundation models including through simple configuration changes.

5.4 Evaluation Methodology

Evaluation employs LLM-as-Judge methodology using Claude Sonnet 3.7 (us.anthropic.claude-3-7-sonnet-20250219-v1:0) to assess synthetic data quality across three critical dimensions using structured prompts and five-point Likert scales. Each generated construction object receives three independent evaluations (consistency, realism, relevance) against the project scenario context through domain-expert prompts that examine specific quality attributes relevant to construction management systems validation, producing 29,304 total evaluation records (9,768 objects $\times$ 3 dimensions).

Consistency: Evaluates internal coherence of object elements, examining whether dates and timelines follow proper sequences, costs and quantities align with scope descriptions, technical specifications match described work requirements, references to other objects remain valid, and all data elements support a coherent narrative without contradictions. Scores range from completely inconsistent with major contradictions (1) to fully consistent with all elements aligning logically (5).

Realism: Assesses authenticity of construction content, evaluating whether terminology and language reflect industry-appropriate usage, described processes follow realistic construction workflows and sequences, object details and specifications appear feasible, roles and stakeholders mentioned are appropriate for the project type, and technical requirements align with actual construction practices. Scores range from completely unrealistic with impossible scenarios (1) to fully realistic mirroring real-world project management practices (5).

Relevance: Measures alignment with project scenario context, examining whether objects relate to correct project type and construction context, scope and purpose align with project requirements, content matches project trades and location characteristics, objects would logically exist in the specified construction project type, and generated artifacts appropriately support the project scenario. Scores range from completely irrelevant and unrelated to project context (1) to fully relevant and perfectly aligned with project type and scope (5).

6 Results and Analysis

6.1 Experimental Data Overview

The experimental framework generated 9,768 construction management objects across 10 iterations of 10 diverse project scenarios, producing 29,304 evaluation records assessing consistency, realism, and relevance dimensions. The dataset encompasses 38 distinct object types spanning planning artifacts, financial records, communication documents, quality control records, technical specifications, operational logs, and stakeholder directories, distributed across 199 project phases and 921 construction processes. Detailed experimental data generation metrics are provided in Appendix C (Table 5).

Nova Pro generated substantially more comprehensive datasets, producing 7,394 objects compared to Nova Lite's 2,374 objects, representing a 211% increase in data volume. Both models successfully generated all 38 object types and covered the complete range of project phases and construction processes, indicating framework robustness across model capabilities. Object distribution patterns remained consistent between models, with documents, reports, and drawings comprising the highest volume categories, reflecting authentic construction management documentation hierarchies.

6.2 Quality Evaluation Results

LLM-as-Judge evaluation across 29,304 assessments reveals unexpected quality patterns, with Nova Lite achieving higher overall scores despite generating fewer objects. The evaluation employed five-point Likert scales assessing consistency, realism, and relevance dimensions for each generated object against project scenario contexts.

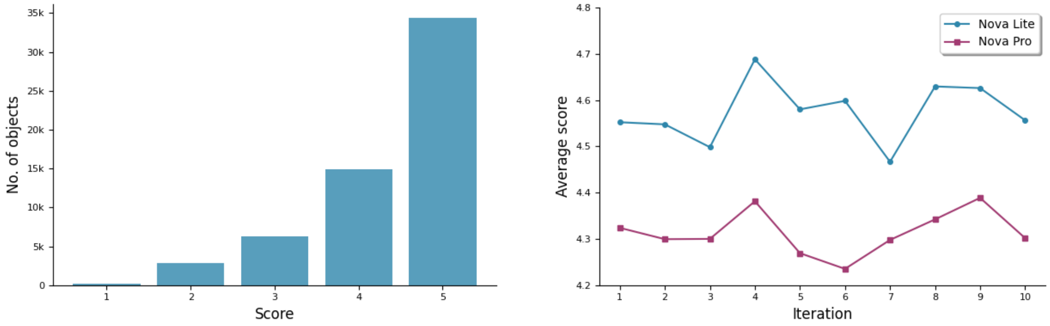

Figure 2: Quality Evaluation Results by Foundation Model. Left: Comparative quality scores across consistency, realism, and relevance dimensions. Right: Object generation volume comparison revealing quality-quantity tradeoffs in foundation model selection.

Dimension	Nova Lite	Nova Pro	Overall	Difference
Relevance	4.77	4.70	4.71	-0.07
Consistency	4.54	4.29	4.35	-0.25
Realism	4.39	3.95	4.05	-0.44
Overall Mean	**4.57**	**4.31**	**4.37**	**-0.26**

Table 1: Quality evaluation scores by dimension and model. Nova Lite achieved higher scores across consistency and realism dimensions despite lower data volume, suggesting quality-quantity tradeoffs in foundation model selection.

Nova Lite achieved a mean score of 4.57 compared to Nova Pro's 4.31, with particularly notable advantages in realism (4.39 vs 3.95) and consistency (4.54 vs 4.29) dimensions. Both models performed similarly on relevance (4.72), indicating comparable capability in generating contextually appropriate objects for specified project scenarios. The overall mean score of 4.37 across all evaluations suggests the framework produces high-quality synthetic data, though significant variation exists across dimensions and models.

Scenario-level analysis reveals residential projects achieved highest quality scores (4.67 mean), while highway bridge projects scored lowest (4.21 mean), suggesting framework performance varies with project complexity and domain specificity. Parking garage, skyscraper, and distribution warehouse scenarios achieved comparable scores (4.43-4.44), indicating consistent performance across commercial and industrial project types.

6.3 Low Score Analysis

Analysis of 1,535 low-scoring objects (scores ≤ 2, representing 5.2% of evaluations) reveals systematic quality challenges. Nova Pro generated 1,289 low-scoring objects (5.8% of its evaluations) compared to Nova Lite's 246 (3.5%), indicating higher-volume models produce more quality outliers. Realism accounts for 69.7% of low scores, consistency for 25.2%, and relevance for only 5.1%, suggesting the framework maintains strong contextual alignment while struggling with authentic construction terminology and internal coherence for certain object types.

Infrastructure projects generated disproportionate low-score concentrations: highway bridge (16.5%), gas station (13.7%), and restaurant outparcel (12.2%), while residential projects produced minimal low scores (1.8%). Object type analysis identifies specifications (13.7%), directory companies (13.1%), reports (11.9%), documents (10.0%), and drawings (9.3%) as primary contributors, suggesting agent prompts lack sufficient domain expertise for specialized technical documentation. Conversely, operational objects like timesheets, daily logs, and equipment records generated minimal low scores.

Representative failures reveal systematic patterns. Highway bridge projects frequently generated HVAC system drawings and MEP specifications inappropriate for open-air infrastructure, demonstrating fundamental project type confusion. Directory company objects exhibited inconsistent contact information failing to align with stated locations. Specification documents referenced building codes inappropriate for project types, while drawing objects showed scale inconsistencies revealing template reuse across incompatible project types.

These patterns suggest three primary challenges: insufficient project type differentiation causing agents to apply building construction templates to infrastructure projects, limited domain knowledge producing technically implausible specifications, and weak cross-object validation allowing internally inconsistent data to propagate. Addressing these requires enhanced agent specialization, expanded domain knowledge bases, and strengthened validation mechanisms ensuring technical plausibility and internal consistency. Cross-object validation mechanisms would address internal consistency failures where generated objects reference non-existent systems or contradict project specifications. Implementing validation layers that verify object references against project scope, check technical specifications against project type requirements, and ensure temporal sequencing consistency would catch quality issues before evaluation. Post-generation validation passes could identify and regenerate objects exhibiting common failure patterns, such as HVAC references in infrastructure projects or inappropriate building codes for project types.

Iterative refinement processes employing LLM-as-Judge evaluation during generation rather than post-hoc assessment could enable real-time quality improvement. Agents receiving immediate feedback on generated objects could adjust subsequent outputs, learning from quality assessment patterns to avoid recurring failures. This approach transforms evaluation from passive assessment to active quality control, potentially reducing low-score incidence while maintaining generation throughput. Implementing quality thresholds requiring regeneration for objects scoring below acceptable levels would ensure only high-quality synthetic data enters final datasets, though at potential computational cost increases.

6.4 Research Questions

RQ1: Can agentic simulacra generate synthetic construction management datasets that maintain semantic coherence, temporal dependencies, and stakeholder interaction patterns required for effective systems testing?

Yes, with qualifications. The framework generated 9,768 objects across 38 types, 199 project phases, and 921 construction processes, demonstrating capability to produce diverse, structured datasets. Overall quality scores averaged 4.37 on five-point scales, with relevance scoring 4.71, indicating strong contextual alignment. However, 5.2% of objects scored ≤ 2, revealing systematic failures in semantic coherence for technical documentation. The framework excels at structured transactional data (timesheets, daily logs, equipment records) but struggles with technical narrative content (specifications, drawings, reports) requiring domain-specific knowledge. Temporal dependencies and stakeholder interaction patterns remain implicit in object metadata rather than explicitly validated, limiting effectiveness for comprehensive systems testing.

RQ2: How do foundation model capabilities influence synthetic data generation robustness, comprehensiveness, and quality for complex sociotechnical domains?

Foundation model capabilities create quality-quantity tradeoffs. Nova Pro generated 211% more objects (7,394 vs 2,374) but scored lower on quality (4.31 vs 4.57 mean), particularly on realism (3.95 vs 4.39) and consistency (4.29 vs 4.54). Nova Pro produced 5.8% low-scoring objects versus Nova Lite's 3.5%, indicating higher-volume models generate more quality outliers. Both models achieved similar relevance scores (4.70 and 4.77) and covered all 38 object types, suggesting equivalent contextual understanding but different execution quality. Advanced models prioritize comprehensiveness over individual object quality, while baseline models produce fewer but more consistent outputs. Domain complexity matters: residential projects scored 4.67 mean while highway bridges scored 4.21, revealing performance variation across project types regardless of model capability.

RQ3: What are the practical trade-offs between cost efficiency and generation reliability for production deployment of agentic synthetic data generation systems?

Cost-quality-volume tradeoffs require careful consideration. Nova Lite costs 13.3x less per token but generates 68% fewer objects, resulting in higher per-object costs when volume requirements exist. Nova Lite's superior quality scores (4.57 vs 4.31) and lower failure rates (3.5% vs 5.8%) suggest better reliability for quality-critical applications. However, Nova Pro's 211% volume advantage provides comprehensive coverage essential for systems testing requiring diverse scenarios. Production deployment decisions depend on requirements: quality-critical applications with modest volume needs favor Nova Lite, while comprehensive testing scenarios requiring extensive coverage justify Nova Pro despite quality tradeoffs. Organizations must balance token costs, quality requirements, volume needs, and post-processing effort when selecting foundation models for production synthetic data generation.

7 Discussion and Conclusion

7.1 Systems Engineering Implications

Agentic simulacra address critical construction management systems engineering challenges. Generated datasets maintain authentic stakeholder interaction patterns and temporal dependencies essential for realistic validation. The framework generates diverse project scenarios at scale, supporting comprehensive testing across varied operational conditions. Synthetic data generation enables systems development without accessing sensitive customer data, addressing privacy and competitive constraints while supporting continuous integration workflows.

7.2 Domain Applicability

The methodology applies to complex sociotechnical systems requiring synthetic data with authentic stakeholder interactions and temporal dependencies. Healthcare patient care workflows, financial transaction processing, supply chain procurement, and software development agile methodologies all involve multiple professional roles coordinating through structured processes with explicit handoff requirements. The approach proves most effective in domains exhibiting well-defined process structures, inaccessible real-world data due to privacy constraints, established professional roles, and structured data conforming to defined schemas.

7.3 Foundation Model Selection

Experimental results reveal unexpected quality-quantity tradeoffs. Nova Pro generated 211% more objects but scored lower on quality metrics (4.31 vs 4.57 mean), particularly on realism (3.95 vs 4.39) and consistency (4.29 vs 4.54) dimensions. Nova Lite produced fewer but higher-quality objects with lower failure rates (3.5% vs 5.8% low scores). Organizations must balance volume requirements against quality needs, with Nova Lite suitable for scenarios prioritizing data quality and Nova Pro appropriate when comprehensive coverage outweighs individual object quality concerns.

7.4 Research Contributions

This research makes several contributions to systems engineering practice. The work introduces agentic simulacra patterns that leverage multi-agent coordination to generate correlated synthetic datasets maintaining authentic domain semantics and workflow dependencies. The research demonstrates practical application of generative agents to construction domain challenges, establishing reusable patterns for complex sociotechnical system data generation. Additionally, the work provides empirical analysis of foundation model performance trade-offs for production synthetic data generation, offering guidance for systems engineering tool selection. Finally, the methodology establishes patterns applicable to other complex domains requiring synthetic data with stakeholder interactions and temporal dependencies.

7.5 Limitations and Future Work

Current experiments involve ten projects with 9,768 objects; production scenarios may require thousands of objects per project. Generated datasets could benefit from more extensive cross-phase handoff patterns demonstrating complete temporal dependency chains. Framework adaptation to other sociotechnical domains requires validation of agent design patterns and process orchestration approaches. Extended project lifecycles may reveal consistency challenges not apparent in current experimental timeframes. Future research should expand experimental validation across complete project lifecycles, implement quantitative correlation metrics complementing qualitative assessment, scale generation volume to production requirements, and adapt the framework to additional sociotechnical domains.

7.6 Conclusion

Agentic simulacra employing multi-agent orchestration generate synthetic construction management datasets exhibiting semantic coherence, temporal dependencies, and stakeholder interaction patterns required for systems engineering validation. Experimental results across 10 project scenarios producing 9,768 objects with 29,304 quality evaluations demonstrate framework viability while revealing quality-quantity tradeoffs between foundation models. The hierarchical multi-agent architecture with explicit correlation injection mechanisms establishes reusable patterns applicable to healthcare, financial services, supply chain management, and other domains requiring synthetic data with authentic stakeholder interactions. The methodology's framework-agnostic design ensures broad applicability across diverse technical environments, establishing agentic simulacra as a viable approach for addressing synthetic data challenges in systems engineering.

Acknowledgments

The authors thank the AWS Generative AI Innovation Center for supporting this work.

During preparation of this work, artificial intelligence was used in research and writing processes. After using these services, output was reviewed and edited as needed, and authors take full responsibility for content.

References

[1] Mihai Nadăş, Laura Dioşan, and Andreea Tomescu. Synthetic data generation using large language models: Advances in text and code, 2025. ISSN 2169-3536. URL `http://dx.doi.org/10.1109/ACCESS.2025.3589503`.

[2] Himanshu Gupta, Kevin Scaria, Ujjwala Anantheswaran, Shreyas Verma, Mihir Parmar, Saurabh Arjun Sawant, Chitta Baral, and Swaroop Mishra. Targen: Targeted data generation with large language models, 2024. URL `https://arxiv.org/abs/2310.17876`.

[3] Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. Trueteacher: Learning factual consistency evaluation with large language models, 2023. URL `https://arxiv.org/abs/2305.11171`.

[4] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior, 2023. URL `https://arxiv.org/abs/2304.03442`.

[5] Yifu Lu, Shengjie Liu, and Li Dong. Orchdag: Complex tool orchestration in multi-turn interactions with plan dags, 2025. URL `https://arxiv.org/abs/2510.24663`.

[6] Benoît Ronval, Pierre Dupont, and Siegfried Nijssen. Tagal: Tabular data generation using agentic llm methods, 2025. URL `https://arxiv.org/abs/2509.04152`.

[7] Guido Ivetta, Laura Moradbakhti, and Rafael A. Calvo. Synthetic social data: trials and tribulations, 2025. URL `https://arxiv.org/abs/2510.19952`.

[8] Amogh Mannekote, Adam Davies, Guohao Li, Kristy Elizabeth Boyer, ChengXiang Zhai, Bonnie J Dorr, and Francesco Pinto. Do role-playing agents practice what they preach? belief-behavior consistency in llm-based simulations of human trust, 2025. URL `https://arxiv.org/abs/2507.02197`.

[9] Shengbin Yue, Ting Huang, Zheng Jia, Siyuan Wang, Shujun Liu, Yun Song, Xuanjing Huang, and Zhongyu Wei. Multi-agent simulator drives language models for legal intensive interaction, 2025. URL `https://arxiv.org/abs/2502.06882`.

[10] Rakshitha Jayashankar and Mahesh Balan. Advancing employee behavior analysis through synthetic data: Leveraging abms, gans, and statistical models for enhanced organizational efficiency, 2024. URL `https://arxiv.org/abs/2409.14197`.

Appendix

A Methodology Details

Agent Role	Phase	Primary Outputs	Key Interactions
Architect	Design	Drawings, Specifications	Structural Eng., MEP Eng.
Structural Engineer	Design	Structural Drawings	Architect, Project Manager
MEP Engineer	Design	System Designs	Architect, Contractor
Regulatory Specialist	Design	Permits, Approvals	All Design Agents
Project Manager	Planning	Schedules, Budgets	All Agents
Estimator	Planning	Cost Estimates	Architect, Procurement
Scheduler	Planning	Project Schedules	Project Manager, Contractor
Procurement Specialist	Planning	Purchase Orders	Estimator, Suppliers
Construction Manager	Construction	Progress Reports	Superintendent, PM
Superintendent	Construction	Daily Logs	Construction Manager
Quality Inspector	Construction	Inspection Reports	Superintendent, CM
Safety Officer	Construction	Safety Reports	All Construction Agents
Commissioning Specialist	Commissioning	Test Procedures	Testing Technician
Testing Technician	Commissioning	Test Results	Commissioning Specialist
Closeout Coordinator	Commissioning	Final Documentation	All Agents

Table 2: Key Agent Roles and Responsibilities within the multi-agent framework. The complete framework employs 15 specialized agents across four construction phases, with each agent maintaining domain expertise and coordination patterns consistent with construction industry practices.

B Implementation Details

Category	Object Type	Description	Schema Fields
Planning	Schedule Activities	Project timeline tasks	id, name, duration
	Schedule Calendar Items	Calendar events	id, date, event
	Tasks	Work assignments	id, title, assignee
	Meetings	Project meetings	id, title, attendees
Financial	Direct Costs	Cost tracking	id, category, amount
	Invoicing	Payment processing	id, vendor, amount
	Change Orders	Scope modifications	id, description, cost
	Prime Contracts	Main agreements	id, contractor, value
	Subcontracts	Trade agreements	id, trade, scope
	Purchase Orders	Procurement orders	id, supplier, items
Communication	Correspondence	Formal communications	id, sender, recipient
	Emails	Electronic communications	id, from, to, subject
	RFIs	Requests for information	id, question, response
	Instructions	Work directives	id, instruction, recipient
	Transmittals	Document delivery	id, documents, recipient
Quality	Inspections	Quality assessments	id, type, results
	Punch List	Deficiency items	id, item, location
	Observations	Quality observations	id, observation, action
	Incidents	Safety reports	id, type, description
Technical	Documents	Project documentation	id, name, path
	Drawings	Technical drawings	id, title, discipline
	Specifications	Technical requirements	id, section, description
	Forms	Data collection	id, type, data
	Reports	Status reports	id, title, content
Operational	Daily Construction Report	Daily progress	id, date, weather
	Daily Log - Accidents	Incident tracking	id, date, type
	Daily Log - Delays	Schedule impacts	id, date, cause
	Daily Log - Deliveries	Material tracking	id, date, supplier
	Daily Log - Equipment	Equipment logs	id, date, equipment
	Daily Log - Inspections	Inspection activities	id, date, type
	Daily Log - Manpower	Labor tracking	id, date, crew
	Daily Log - Productivity	Work output	id, date, activity
	Daily Log - Safety Violations	Compliance issues	id, date, violation
	Daily Log - Scheduled Work	Work execution	id, date, work
	Daily Log - Visitors	Visitor tracking	id, date, visitor
	Crews	Crew assignments	id, crew, members
	Equipment	Equipment records	id, type, model
	Timesheets	Time tracking	id, worker, hours
	T&M Tickets	Time and material	id, work, time
Stakeholder	Directory Companies	Vendor database	id, company, contact
	Directory Users	Team directory	id, name, role
	Bidding	Contractor selection	id, project, bidders
	Submittals	Product approval	id, product, vendor

Table 3: Construction Management Object Types - 38 object types organized by functional category.

Project	Description
Metropolitan Financial Tower	52-story office tower in Seattle (1.85M SF)
Meridian Heights Apartments	324-unit multi-family in Denver (285K SF)
I-85 Catawba River Bridge	1,240-foot highway bridge in North Carolina
Metro Transit Parking Structure	6-level, 1,200-space garage in Portland
Westfield Gateway Mall	1.25M SF shopping center in Texas
Hillcrest Estate Residence	8,500 SF custom home in Washington
QuickStop Travel Center	5,850 SF gas station in Georgia
Riverside Crossing Restaurant	4,850 SF fast-casual restaurant in North Carolina
Midwest Regional Distribution Center	625K SF warehouse in Indiana
Riverside Regional Water Treatment Plant	15 MGD water treatment expansion in Iowa

Table 4: Synthetic Project Scenarios representing diverse construction types, scales, and complexity levels.

C Results & Analysis

Metric	Nova Lite	Nova Pro
Total objects generated	2,374	7,394
Objects per scenario (mean)	237	739
Unique object types	38	38
Project phases covered	199	199
Construction processes	920	920
Top object types		
Documents	18.7%	18.7%
Reports	10.7%	10.7%
Drawings	9.6%	9.6%
Schedule Activities	6.0%	6.0%
Forms	5.3%	5.3%

Table 5: Experimental data generation summary showing Nova Pro generated 211% more objects than Nova Lite while maintaining identical object type coverage and process diversity.

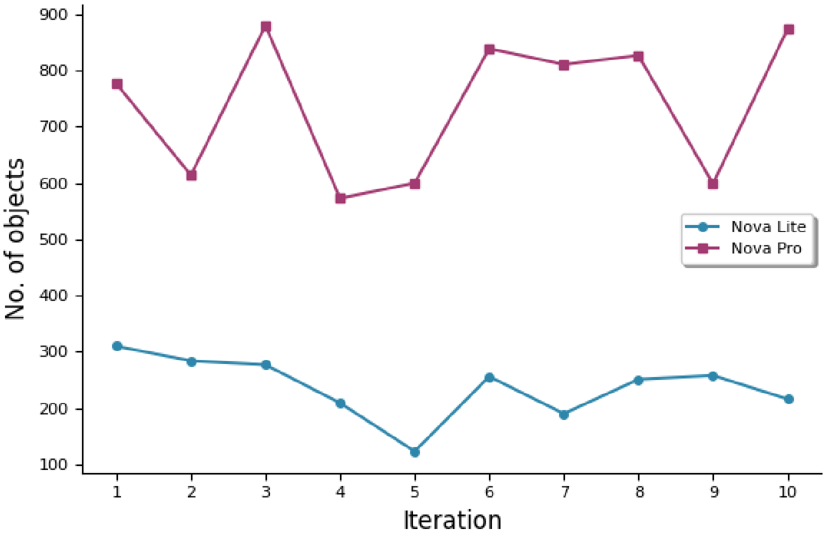

Figure 3: Object Type Distribution Across Construction Management Categories. Generated synthetic datasets span 38 distinct object types organized into seven functional categories including planning artifacts, financial records, communication documents, quality control records, technical specifications, operational logs, and stakeholder directories, demonstrating comprehensive coverage of construction management system requirements.

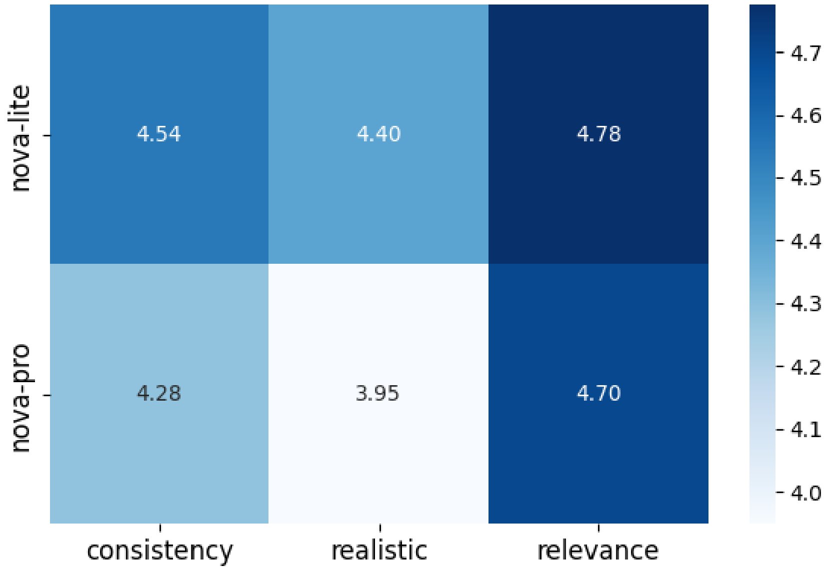

Figure 4: Score Distribution by Project Scenario. Scenario-level analysis reveals performance variation across project types, with residential projects achieving highest quality scores and infrastructure projects presenting greater complexity challenges, indicating framework effectiveness correlates with domain familiarity and project type characteristics.

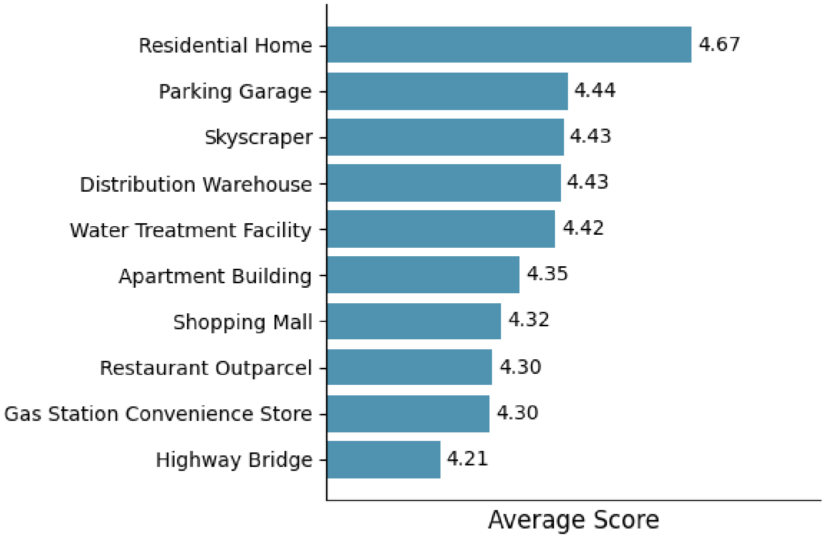

Figure 5: Average Quality Scores by Construction Project Type. Comparative analysis of mean evaluation scores across ten diverse construction scenarios demonstrates consistent framework performance, with scores ranging from 4.21 to 4.67, revealing that project complexity and domain specificity influence synthetic data quality.