```
Every technical element MUST show progression from previous batches:

**Natural Development Progression Examples:**
- **Planning Phase:** "I need to design the authentication system..."
[More examples]

**Technical Complexity Progression:**
- Early batches: Basic implementation, simple features
- Middle batches: Integration challenges, debugging complex issues
- Later batches: Performance optimization, advanced features, production concerns

=========== CONFLICT & RESOLUTION TRACKING ===========
**Mandatory Technical Conflict Elements:**
Each batch must include at least 2-3 technical challenges with:
- **Clear Stakes:** what's at risk (deployment deadline, performance SLA, budget constraint)
- **Binary Decisions:** chose Framework A over B, implemented Solution X vs Y, fixed vs workaround
- **Measurable Outcomes:** reduced latency by Xms, saved $X in hosting, improved performance by X%
- **Specific Trade-offs:** what was sacrificed for what gain (memory for speed, complexity for features)

**Technical Conflict Types to Rotate:**
- Performance bottlenecks with specific metrics
- Architecture decisions with concrete alternatives
- Integration challenges with external systems
- Security vulnerabilities with severity levels
- Scalability issues with user load numbers
- Technical debt vs new features

=========== CONTENT DISTRIBUTION STRATEGY ===========
**Per Batch Requirements:**
- 2-3 bullets: Technical implementation details with specific code elements
- 2-4 bullets: Current development status with measurable metrics
- 1 bullet: Exact temporal anchor (specific date/time)
- 5-7 bullets: Development activities with verifiable outcomes
- 3-4 bullets: Technical decisions with specific alternatives considered
- 1 bullet: **Preference Statement**: implicitly showing developer preferences
- Rest: Using remaining labels with concrete technical details

**Adaptive Batch Planning:**
Each batch should organically focus on what makes sense for that development phase:

**Implementation-Heavy Batch:**
- Multiple implementation requests
- Architecture decisions
- Code structure planning
- Framework/library selection

[More examples]

=========== NATURAL CODING CONVERSATION FLOW ===========
Each bullet should represent realistic developer-AI interactions:

**Implementation Requests:**
[Examples]

**Debugging Help:**
[Examples]

**Code Review/Optimization:**
[Examples]

=========== EXECUTION NOTES ===========
- Use plain, technical language throughout
- Include realistic technical specificity: version numbers, error messages, configuration details
- Make every bullet contribute to the overarching development story
- Ensure uniform technical detail quality across ALL batches
- Vary batch focus organically based on development phase (implementation vs debugging vs optimization)
- Prioritize concrete technical details over abstract descriptions
- Every bullet should enable at least 2-3 factual technical questions
- End immediately after 'BATCH <num_batches> PLAN'
```

Listing 25: Coding domain conversation plan generation prompt

```
=========== CRITICAL DETAIL REQUIREMENTS ===========
**MANDATORY SPECIFIC DETAILS:**
Every batch MUST include numerous concrete, verifiable mathematical details that enable single-word or short
     factual answers:

**Required Detail Categories (minimum 5-7 per batch):**
- **Exact Numbers:** specific values (x = 3.14), coefficients ($2x^2 + 5x - 3$), dimensions ($5\times7$ matrix
     )
- **Specific Problems:** complete equations ($x^2 - 4x + 3 = 0$), specific integrals ($\int_0^5 (2x+1)\,dx$)
- **Named Concepts:** theorem names (Pythagorean Theorem), method names (Gaussian elimination), formulas (
     quadratic formula)
- **Calculation Results:** exact answers ($x = 4$), decimal results ($\pi \approx 3.14159$), fractions ($\
     tfrac{3}{4}$)
- **Yes/No Situations:** problem solved correctly, method applicable, theorem satisfied, solution exists
- **Score/Grade Metrics:** test scores (85%), homework grades (18/20), quiz results (9/10 correct)
- **Time/Duration:** study hours (3 hours), problem completion time (15 minutes), exam duration (2 hours)
- **Mathematical Properties:** function characteristics (continuous, differentiable), matrix properties (
     invertible, symmetric)
```