

Case Study: Northern Toronto Hospital Emergency Department (NTH-ED)

Instructor: Prof. Arik Senderovich

January 2026

1 Case Description: The Capacity Crisis at Northern Toronto Hospital

Northern Toronto Hospital (NTH) serves as a critical healthcare hub in the Greater Toronto Area, managing a high-velocity environment with over 16,000 unique patient visits annually. However, the Emergency Department (ED) is currently facing a crisis of confidence. In recent months, NTH has been the subject of several media reports highlighting the reality of “hallway medicine,” where the physical limits of the facility are tested by patients awaiting beds in corridors.

The Chief Operations Officer (COO) is focused on two deteriorating metrics that threaten the hospital’s reputation and clinical outcomes: the **Time to Physician’s Initial Assessment (PIA)** and the rate of patients who **Left Without Being Seen (LWBS)**. Initial data suggests a department under siege; patient volumes surge predictably between 10:00 AM and 10:00 PM, creating massive queues in the Yellow (YZ) and Green (GZ) zones. With nearly 20% of all encounters arriving via ambulance and approximately 18% of visits resulting in hospital admission, the ED has become a complex, high-stakes bottleneck for the entire healthcare system.

The COO has commissioned your team to analyze a high-fidelity event log containing 90,000+ milestones. Your goal is to move beyond aggregate statistics and use process mining to visualize the actual patient journeys, identifying where operational friction leads to clinical risk.

1.1 The Data

The provided event log (`event_log_ED_MMA_2026.csv`) contains detailed clinical and operational milestones for each patient visit. The key attributes include:

- **visit_id**: The unique identifier for each patient encounter (Case ID).
- **triage_code**: Canadian Triage and Acuity Scale (CTAS) levels (1: Resuscitation, 2: Emergent, 3: Urgent, 4: Less Urgent, 5: Non-Urgent).
- **initial_zone**: The physical area where the patient was first placed (e.g., YZ - Yellow Zone, EPZ - Emergency Psychiatric Zone, GZ - Green Zone).
- **Events**: Milestones such as *Ambulance Arrival*, *Triage*, *Registration*, *Assessment*, *Ambulance Transfer*, *Discharge*, and *Left ED*.

1.2 Operational Challenges

NTH’s ED operates as a complex queueing system with multiple priority classes (triage levels) and shared resources (physicians, nurses, and beds in specific zones). The administration is particularly concerned about:

1. **Bottlenecks:** Which stages of the process (e.g., Triage to Registration or Registration to Assessment) contribute most to the total length-of-stay (LOS)?
2. **Zone-Specific Performance:** Are certain zones (like the EPZ - Emergency Psychiatric Zone or the Resus - Resuscitation) experiencing systemic delays?
3. **Priority Inversion:** Do lower-acuity patients sometimes get treated faster than higher-acuity ones?
4. **Consultation Delays:** A significant number of cases involve *Consult Request* and *Consult Arrival*. How do these impact the overall flow?

1.3 Data Characteristics & Challenges

Preliminary analysis of the dataset reveals several nuances you must account for:

- **Simultaneous Events:** Events like *Triage* and *Ambulance Arrival* often share the exact same timestamp. Your tools must handle sub-second ordering or logical sequencing.
- **Missingness:** The `initial_zone` is missing for approximately 2% of events, and consultation details (`consult_desc`) are only present for patients requiring specialist intervention.
- **Disposition Outcomes:** Patients are either discharged (mostly to private homes) or admitted to the facility (approx. 18% of cases). Predicting this early in the process is of high clinical value.

1.4 A Note on Data Ethics

Although this dataset is provided for educational purposes, treat it with the professional care required for real clinical data. Your tool should demonstrate an awareness of data integrity and should not output identifiable patient information if it were to be used in a live hospital environment.

2 Part 1: Process Discovery and Analysis with DISCO (25% of Final Mark)

In this part, you will use **Fluxicon DISCO** to perform a professional-grade process mining analysis of the NTH-ED event log.

2.1 Objective

Visualize the actual "as-is" process, identify deviations from the standard clinical pathway, and quantify performance metrics.

2.2 Tasks

1. Data Import & Mapping:

- Case ID: `visit_id`
- Activity: `event`
- Timestamp: `timestamp`
- Resource: `initial_zone`
- Other attributes: Map `triage_desc`, `age`, and `disposition_desc` as Data attributes.

2. Process Discovery:

- Identify the "Happy Path" for a CTAS Level 2 (EMERGENCY) patient.
- Use the *Attribute Filter* to isolate cases that spent time in the **EPZ** zone. How does the process graph change?

3. Performance Analysis:

- Calculate the median time between *Registration* and *Assessment*. This is the **PIA (Physician's Initial Assessment)** time.
- Identify the **LWBS (Left Without Being Seen)** rate: Analyze patients with disposition descriptions such as *"Left After Triage"* or *"Left After Initial Assessment"*. What is the median time these patients spent in the ED before leaving?
- Identify the "Ping-pong" effect: How often do patients move back and forth between zones or repeat assessments?

4. Consultations: Filter for cases containing the *Consult Request* activity. What is the average delay added by waiting for a consultant to arrive?

2.3 Deliverables

Submit a concise PDF report (max 5 pages) containing:

- Annotated process maps for the main triage levels.
- A table summarizing the top 3 bottlenecks found.
- Strategic recommendations for the NTH COO based on your DISCO findings.

3 Part 2: Building an Intelligent Process Mining Tool (75% of Final Mark)

The final component of this course is to move beyond off-the-shelf software. You will use AI-assisted development tools (such as **Cursor**, **GitHub Copilot**, or **ChatGPT**) to build a generalized, intelligent process mining app that serves as a **Decision Support Tool for Emergency Rooms**.

3.1 Objective

Develop a tool that not only mines event logs but also integrates the advanced concepts we covered in the MMA program: Machine Learning, Optimization, and Causality. **Critical to this part is designing the tool with a specific user persona in mind.**

3.2 User-Centered Design & Personas

Before coding, you must define the **Persona** who will use your tool. For example:

- **The ED Flow Coordinator:** Needs a real-time dashboard to identify which zones are currently over-capacity.
- **The Hospital COO:** Needs a high-level performance overview to make long-term resource allocation decisions.
- **The Triage Lead:** Needs predictive insights to decide which patients are likely to be admitted vs. discharged.

Your tool's interface and features must be tailored to the specific needs, pain points, and technical proficiency of your chosen persona.

3.3 Core Functional Requirements

Your tool must implement the following "Foundational" and "Advanced" features:

1. **Discovery (Foundational):** Automatically generate a Directly-Follows Graph (DFG) from any uploaded CSV. The graph should be interactive, allowing users to drill down into specific activities.
2. **Conformance (Foundational):** Implement a basic conformance checking module. Allow the user to define a "Standard Protocol" (e.g., Triage → Registration → Assessment) and identify cases that deviate from this norm.
3. **Queue Mining (Foundational):** Calculate and visualize queue lengths and waiting times across different resources or zones (`initial_zone`). This is where your Queueing Theory knowledge must be "lifted" into the code.
4. **Machine Learning Integration (Advanced):**
 - *Predictive Analytics:* Predict the "Probability of Hospital Admission", the "Probability of LWBS", or the "Remaining Time to PIA".
 - *Anomaly Detection:* Highlight "Red Flag" cases (e.g., extremely long wait times or unusual event sequences).
5. **Advanced Analytics (Choose 1):**

- **Simulation:** Run Monte Carlo simulations to show the impact of adding capacity.
- **Causality:** Use causal inference to explain *why* certain delays occur.
- **Reinforcement Learning:** Suggest “Next Best Actions” for the user persona to optimize flow.

3.4 User Experience (UX) & Implementation Integrity

The tool should be intuitive and professional. Avoid “feature soup”: every widget should serve the chosen persona.

- **Consistency:** Ensure that ML insights are contextualized within the process view (e.g., predicting delays directly on the process edges).
- **AI-Assisted Development:** Document how you used tools like Cursor to implement complex algorithms (e.g., DFG generation or queue calculations).

3.5 Technical Recommendations

While you are free to choose your tech stack, we highly recommend using **Python** with a web framework such as **Streamlit** or **Plotly Dash** for Part 2. These tools allow for rapid UI development and have excellent libraries for DFG visualization (e.g., **graphviz**), queueing calculations, and ML integration.

3.6 Final Submission & Grading

Your grade for Part 2 will be based on:

1. **The Tool (50%):** Functionality, code quality, and successful integration of ML/Queueing modules.
2. **Video Demo (25%):** A 10-minute screen recording where you:
 - Demonstrate the tool using the `event_log_ED_MMA_2026.csv` file.
 - Explain the logic behind your ML or Simulation components.
 - Show how you used AI tools (like Cursor) to accelerate your development.