

# PROQA 7

---

Modernizing Emergency Dispatch Software  
for the Next Generation

Jordan Plank | Group Product Manager

# EXECUTIVE SUMMARY

## OBJECTIVE

Create a modernized, unified platform for emergency dispatch across all three disciplines (EFD, EPD, EMD) with a browser-based UI, simplified installation, and standardized third-party integrations.



### KEY OUTCOMES

- Unified three separate discipline applications into one platform
- Modernized UI with React and Tailwind while preserving familiar workflows
- Simplified installation and update processes for agencies
- Plan for hybrid On-Prem/SaaS architecture for flexible deployment
- Established foundational platform for future product integrations



### CORE TRADEOFFS

- Paradigm redesign vs. UI refresh approach
- Browser-based SaaS vs. On-Prem first deployment
- Navigating multiple leadership transitions mid-project
- App Menu vs. Discipline Selector
- Custom CAD integrations vs. standardized pub/sub framework

# INDUSTRY CONTEXT



## PLATFORM CHALLENGES

- Outdated interface lacking modern UX standards
- Complex installation process agencies
- Separate applications for each dispatch discipline



## MARKET PRESSURE

- Growing demand for modern, unified dispatch solutions
- Competitors advancing with cloud-based architectures
- Customers expected seamless updates and SaaS delivery



# PROBLEM STATEMENT

## CUSTOMER PROBLEMS

- Outdated interface hinder dispatcher efficiency
- Painful installation process for multi-discipline agencies
- Licensing and update processes are cumbersome
- Separate apps for EFD, EPD, and EMD created friction in protocol fluidity
- No standardized approach for CAD integrations

## BUSINESS PROBLEMS

- Original “Paradigm” plan was off-target and not customer-validated
- Internal bias towards existing design choices
- Multiple leadership transitions created instability
- Blackbox product documentation and requirements
- No in-app analytics to inform product decisions
- Way off-target delivery schedules with legacy project plan

# MY ROLE

*Product Owner → Product Manager → Group Product Manager*



## PRODUCT VISION

- Evolved product vision through leadership transitions
- Defined long-term platform approach and roadmap
- Championed UI refresh over Paradigm redesign



## CUSTOMER DISCOVERY

- Focus groups and agency shadowing sessions
- Salesforce and implementation data analysis
- Conference demos and SME collaboration



## TEAM LEADERSHIP

- Maintained team stability through leadership turnover
- Built comprehensive documentation and requirements
- Aligned stakeholders on transparent project plans

# USER RESEARCH & DISCOVERY



## CUSTOMER FEEDBACK

- Focus groups and conference demonstrations
- Agency shadowing to observe real workflows
- Revealed Paradigm plans were not accepted by users via statements like “Training to the new design will be a challenge” and flat out “I don’t like it”



## DATA ANALYSIS

- Salesforce reporting for customer support patterns
- Implementation data and 1:1 customer feedback
- No in-app analytics, relied on qualitative data
- Partnership with data science team for multi-disc. scenarios



## STAKEHOLDER ALIGNMENT

- Made research visible and transparent to leadership
- Aligned SME experts with customer-driven insights



# TRADEOFF #1: PARADIGM VS. UI REFRESH

## PARADIGM REDESIGN

- Completely new UI/UX design
- New color scheme and layouts
- Users confused by unfamiliar workflows
- Clashed with intentional formatting
- Not validated by customer feedback

## UI REFRESH

- Preserved familiar ProQA workflows
- Modernized with React and Tailwind
- Aligned with customer expectations
- Required re-evaluation after leadership change
- More incremental change vs. dramatic shift

## DECISION

Championed the UI refresh approach to maintain workflow familiarity while modernizing the technology stack with React and Tailwind.

**GAVE UP:** A bold, completely new visual experience that could have differentiated ProQA 7 more dramatically, but risked alienating dispatchers who depend on established workflows.

# TRADEOFF #2: APP MENU VS. DISCIPLINE SELECTOR

## APPLICATION MENU

- Separate apps for EFD, EPD, and EMD
- Toggle between disciplines at app level
- Users must exit workflow to switch disciplines
- Still feels like three separate applications
- Adds unnecessary navigation complexity

## DISCIPLINE SELECTOR

- Unified app with in-context discipline selection
- Core Service Engine houses all three logic files
- Hidden for single-discipline; shown for multi-access
- Embedded in Case Entry for seamless workflow
- Requires all discipline logic in one engine

## DECISION

Chose the discipline selector within Case Entry over a separate app menu, leveraging the Core Service Engine to unify all three disciplines into one seamless application experience.

**GAVE UP:** The simplicity of independent applications per discipline, accepting the engineering complexity of housing all three discipline logic files within a single Core Service Engine.

# CUSTOMER-FOCUSED UX DECISIONS

## UX PRINCIPLES



Preserve familiar ProQA 5 workflows



Modernize without disrupting muscle memory



Align with voice-of-the-customer insights



Remove personal bias from design decisions

## KEY UX FEATURES

- Unified multi-discipline interface
- Simplified licensing and update workflows
- Browser-based access for any device
- Transparent project plan visible to stakeholders
- Standardized CAD integration experience

VALIDATED THROUGH: Focus groups • Agency shadowing • Conference demos • Salesforce data analysis

# TRADEOFF #3: SAAS VS. ON-PREM FIRST

## ON-PREM FIRST

- Serves majority of existing customer base
- Offline-first architecture essential for high uptime
- Critical for dispatch reliability in emergencies
- Does not address growing SaaS market demand
- Slower iteration cycles for testing and updates

## BROWSER-BASED SAAS

### Addresses growing SaaS market demand

- Simplified user testing without local installs
- Faster update and deployment cycles
- ProQA 5 already serves On-Prem customers
- Requires reliable cloud infrastructure

## DECISION

I really pushed for On-Prem and offline-first architecture as a main priority. A large portion of our customer base depends on it, and offline architecture is essential for dispatch reliability. Ultimately chose SaaS-first to address market demand while ProQA 5 bridged the gap.

**MOST PROMISING APPROACH I DIDN'T CHOOSE:** On-Prem with offline-first architecture — it serves our largest customer segment and guarantees the uptime dispatch centers need. We mitigated this by maintaining ProQA 5 as the On-Prem bridge.

# IMPLEMENTATION CHALLENGES

## LEADERSHIP TURNOVER & PROJECT STABILITY

Multiple leadership changes threatened to derail the project. The engineering director left with undocumented knowledge, followed by a CTO departure due to financial issues.

## APPROACH

- Stepped up as a stabilizing leader during transitions
- Built comprehensive documentation to prevent knowledge loss
- Maintained transparent project plans and requirements
- Worked with each new leader to preserve validated plans

**Maintained team morale and project momentum through three leadership transitions**

# TRADEOFF #4: CAD INTEGRATION APPROACH

## CUSTOM PER PARTNER

- Tailored to each CAD partner's needs
- Maintains existing integration patterns
- Each integration takes 6-12 months to update
- Not scalable across growing partner base
- Unsustainable engineering overhead

## STANDARDIZED FRAMEWORK

- Azure Pub/Sub provides reliable messaging
- Simplified developer-friendly web socket API
- Enables agency-wide or user-specific messaging
- Requires CAD partners to adopt new approach
- Initial investment in new infrastructure

## DECISION

Moved toward a standardized Azure Pub/Sub framework with a developer-friendly web socket API, reducing integration time and enabling scalable partner onboarding.

**GAVE UP:** Tailored, partner-specific integrations that accommodated unique CAD workflows, requiring partners to adopt and transition to the new standardized framework.

# CAD INTEGRATION: TECHNICAL MODERNIZATION

## REST API TRANSITION

- RESTful APIs with JSON payloads replacing legacy XML
- Scalable, cloud-compatible integrations

## SOFTWARE DEVELOPER KIT

- API docs, sample code, and payload examples
- In-house prototype CAD app and reference workflows



## MODERNIZED SPECIFICATIONS

Cleaner format • Clear field requirements • Defined data structures • Version compatibility • Faster internal updates

# CAD INTEGRATION: CONFIGURATION & BENEFITS



## SIMPLIFIED PARTNER CONFIG

- ProQA centrally owns formatting and styling
- Eliminates partner hardcoding of display rules

## EXPLICIT FORMATTING & METADATA

- Styling directives delivered directly in payload
- Predictable rendering across all CAD systems

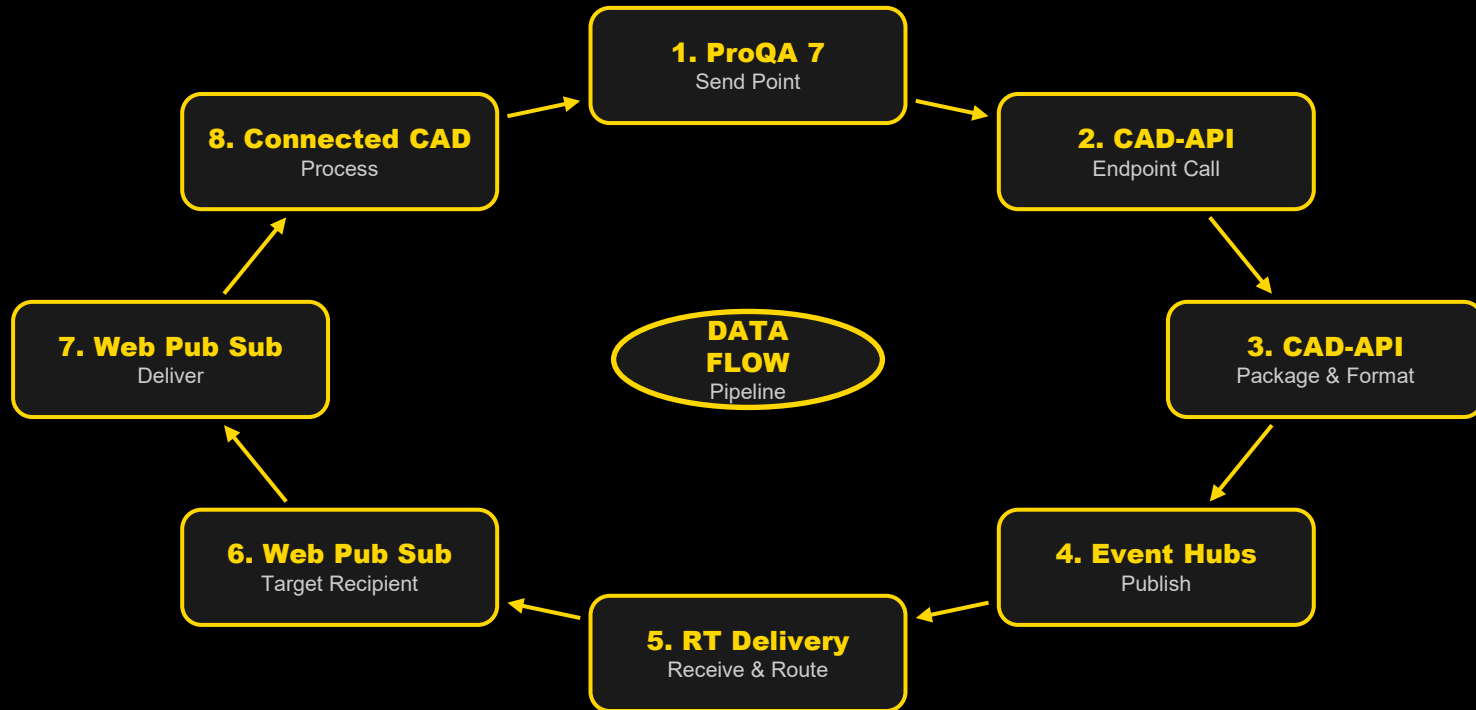
## STRATEGIC BENEFITS

**CAD PARTNERS:** Faster onboarding • Less custom work • Better Developer Experience

**PROQA / PDC:** Faster deploys • Display control • Interoperability

# CAD ARCHITECTURE

Data Flow Pipeline — Send Point to CAD Delivery



## REST APIs • Request/Response

Creating incidents, fetching records, updating unit status, authentication, config retrieval, historical queries, admin actions

*Reliable • Stateless • Easy to debug • Legacy vendor friendly*

## WebSockets • Real-Time Streaming

Live call updates, dispatch events, unit movements, queue state, ProQA determinant updates, active incident sync, dashboard feeds

*Persistent connection • Low latency • Event-driven • Bi-directional*

# PLATFORM ARCHITECTURE



## CORE ARCHITECTURE

- Browser-based React + Tailwind UI
- Core Services Logic Engine for dispatch workflows
- Hybrid On-Prem/SaaS deployment model
- Azure Event Hubs for real-time messaging
- Azure Web Pub/Sub for CAD communication
- Unified application for EFD, EPD, and EMD



## DESIGN PRINCIPLES

- UI refresh preserving familiar ProQA workflows
- Offline-first architecture for high uptime
- Simplified installation and update processes
- Standardized third-party integration framework



**FOCUS:** Building a foundational platform that is iterative and can be built upon for future product integrations.

# OUTCOMES & BUSINESS IMPACT

*ProQA 7 Modernization — Platform Transformation*

**3-in-1**

Unified  
Discipline Platform

**End  
2026**

Core Pilot  
Target Date

**3**

Leadership  
Transitions Navigated

**Majority**

Stakeholder  
Alignment

## PROGRESS

Stakeholders and users have full transparency into project plans  
Increased trust earned through navigating leadership changes

## FUTURE VISION

Foundation for QA and AI Training app integrations  
2–5-year plan for unified platform across all products

*ProQA 7 will serve as the base architecture for future products to integrate with*

# MEASURING AN IN-FLIGHT MODERNIZATION

| What I Can Measure Today    | Current Evidence                                                   |
|-----------------------------|--------------------------------------------------------------------|
| Stakeholder Alignment       | <b>Majority buy-in maintained through 3 leadership transitions</b> |
| Pilot Agency Engagement     | <b>Core pilot on track for end 2026 launch</b>                     |
| Demo & Focus Group Feedback | <b>Positive reception at conferences and agency sessions</b>       |
| Support Ticket Patterns     | <b>Salesforce data driving ProQA 7 design priorities</b>           |

## HOW I KNOW WE'RE ON TRACK

ProQA 7 hasn't shipped yet, but these are the real signals I'm tracking today to validate we're building the right thing before launch metrics exist.



Agency shadowing insights



Conference demo reception



Pilot agency commitment



Support ticket trend analysis

# KEY LESSONS LEARNED



## NAVIGATING TURNOVER

Stepping up as a leader during instability kept morale and momentum alive through three leadership transitions.



## DOCUMENTATION MATTERS

Solid requirements, transparent project plans, and thorough documentation prevented critical knowledge loss.



## CUSTOMER ALIGNMENT

Removing personal bias and aligning with the voice of the customer earned trust with stakeholders and users.



## GRIT & RESILIENCE

Determination to protect validated plans through adversity ensured the project survived and thrived.

# MY PRODUCT PHILOSOPHY



## **Start with Customer Reality**

Remove personal bias and align with the voice of the customer before defining solutions



## **Lead Through Uncertainty**

Step up during instability and keep teams aligned through transparent communication



## **Build for Iteration**

Create foundational platforms that are iterative and can be built upon for future growth



## **Balance Vision with Execution**

Navigate both long-term platform strategy and immediate business needs with grit and resilience

---

*ProQA 7 is more than a software modernization — it represents a strategic transformation of emergency dispatch software into a unified, scalable platform that will serve as the foundation for the next generation of products.*