MEDICAL PHYSICS COMMUNITY · INSPIRATION · INNOVATION · IMPACT
2026 JULY 19–22
VANCOUVER, BC
JOINT AAPM | COMP MEETING

QA Vibes: Rapid Deployment of an MPC Auditor Via LLM-Augmented Vibe Coding

Jason Paisley, MS, DABR[1,2];
[1]Novant Health Cancer Institute[2]New Hanover Regional Medical Center

ABSTRACT

Purpose: Transitioning Linac QA from time based testing to data-driven quality management is hindered by the high technical barrier of developing custom clinical tools. This work demonstrates "vibe coding" as a viable solution, leveraging a Large Language Model (LLM) to rapidly architect, debug, and deploy a production-ready auditing platform for Varian's Machine Performance Check (MPC). The resulting platform utilizes Statistical Process Control (SPC) to identify risks obscured by traditional binary pass/fail reviews.
Methods: A clinical physicist with limited software engineering experience utilized an LLM (Gemini 3.0) to co-develop a Python-based auditing ecosystem. The LLM was used to architect a recursive scraper to parse longitudinal MPC data from network drives. The LLM further assisted in implementing a 5-sigma outlier filter, calculating Process Capability (Cp) and 30-day drift velocity for over 120 parameters. Total development time, from conceptualization to an interactive HTML "Traffic Light" dashboard, was quantified to assess the "speed-to-clinic" advantage of LLM-assisted coding. The LLM-generated logic was manually validated against a benchmark dataset.
Results: Total development time from conception to a working prototype was 3.7 hours. Automated analysis of the data revealed that ~90% of daily QA metrics were process capable (Cp > 1.33), suggesting that manual daily review for these parameters may be redundant. Conversely, the output and beam unitformity exhibited a Cp of <1 (Incapable). This statistical failure was detected despite the system passing Varian tolerances, providing a proactive trigger for intervention before clinical thresholds were breached.
Conclusion: LLM-augmented custom coding development represents a paradigm shift in medical physics with the potential to move the field closer to real time predictive machine maintenance. This study demonstrates that "vibe coding" bridges the gap between complex clinical data and actionable quality management, allowing physicists to rapidly deploy tools that optimize QA workflows.

CONTACT

Jason Paisley
Novant Health Cancer Institute- New Hanover Regional Medical Center
1988 A S. 16th St. Wilmington, NC, 28401
Jason.Paisley@novanthealth.org

INTRODUCTION

The Problem with Current QA Methods

- Linear accelerator (linac) Quality Assurance (QA) is primarily time-based. This approach ensures regular testing intervals but fails to recognize improvements in modern linac construction and frequently requires performing tests on highly stable parameters.
- Performing tests on stable parameters can cause QA fatigue and increases the risk of critical alerts being overlooked.
- Vendor tools like Varian's Machine Performance Check attempt to automate and streamline QA but frequently lack tools for meaningful statistical analysis. There is too much data to effectively sift through the "noise".

The Solution and Paradigm Shift

- This work proposes a shift toward data-driven quality management. We developed a fully automated, recursive clinical auditor that handles the entire data lifecycle of a linac in minutes. Using statistical process control, the auditor evaluates the process capability Cp of each MPC test.

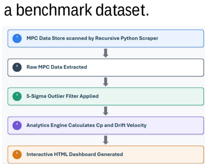
$$C_p = \frac{Threshold}{6\sigma} \quad (1)$$

- A process is considered capable or very stable if the $C_p > 1.33$. If $C_p < 1.0$, the process is considered incapable as the routine variation is too wide. This was used to develop a traffic light dashboard.

Condition	Status	Meaning
Cp≥1.33	🟢	Stable
1≤Cp<1.33	🟡	Variation Acceptable
Cp<1.0	🔴	Incapable. Variation too wide

METHODS AND MATERIALS

- A clinical physicist utilized an LLM (Gemini 3.0) for AI assisted development (vibe coding) of a Python-based auditing ecosystem.
- The Python based recursive scraper was used to parse the entire MPC dataset for the linac.
- The data was processed by implementing a 5-sigma outlier filter and calculating the Process Capability (Cp) and 30-day drift velocity for over 120 parameters.
- Total development time, from conceptualization to an interactive HTML "Traffic Light". dashboard, was quantified to assess the "speed-to-clinic" advantage of LLM-assisted coding.
- The auditor calculations were manually validated against a benchmark dataset.

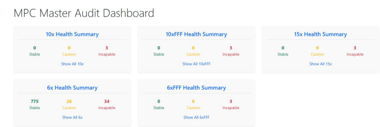


RESULTS

Development Cycle and Performance

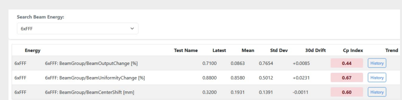
- Total development time from conception to a working prototype was ~ 3.7 hours
- The audit tool successfully processed 5 years of data with over 840 parameters. The run time for the audit tool was 4 minutes 30 seconds.

The interactive HTML dashboard output is shown below



Significant Findings

- An analysis of the MPC checks showed that over 90% of the tests were process capable C_p>1.33. This demonstrates that many parameters are stable year over year.
- The largest variation was seen in output, beam uniformity and beam center shift
- MLC leaf position also showed low (<1) C_p values
- The 30-day drift velocity metric for process incapable tests were reviewed to determine if they could be used to predict impeding failures.



Search Beam Energy:
6xFFF

Energy	Test Name	Latest	Mean	Std Dev	30d Drift	Cp Index	Trend
6xFFF	6xFFF: BeamGroup/BeamOutputChange [%]	0.7100	0.0863	0.7654	+0.0085	0.44	History
6xFFF	6xFFF: BeamGroup/BeamUniformityChange [%]	0.8800	0.8580	0.5012	+0.0231	0.67	History
6xFFF	6xFFF: BeamGroup/BeamCenterShift [mm]	0.3200	0.1931	0.1391	-0.0011	0.60	History

Figure 1. Dashboard output .

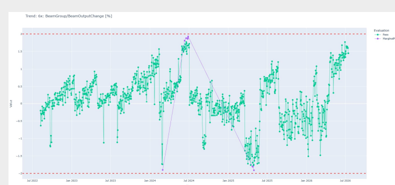


Figure 2. Longitudinal Plot of Beam Output Check .

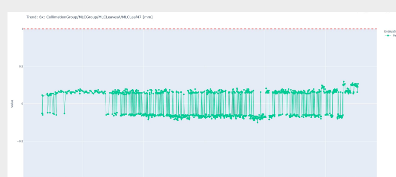


Figure 3. Longitudinal Plot of MLC position .

DISCUSSION

Physicist Review

- A key finding of this work is the decoupling of mathematical process capability from clinical tolerance. For example, mechanical metrics like MLC leaf position (Figure 3) exhibit extremely tight baseline precision. Consequently, small changes that are in fact clinically negligible trigger a sub-threshold C_p < 1.0, suggesting the process is incapable, even though the absolute physical deviation remains well within safe operational limits.

Possible Improvement to Prevent QA Fatigue

- To prevent the dashboard from flagging stable parameters that possess very high baseline precision, the next iteration of the software must implement a physical tolerance floor (e.g., aligning with TG-142 criteria) alongside the C_p calculation.

The Predictive Power of Drift Velocity

- While the C_p index (Equation 1) acts as a powerful alert, the 30-day drift velocity provides a predictive early-warning mechanism. This allows physicists to monitor linear trends and intervene prior to a parameter exceeding clinical thresholds.
- During this data collection period, the MV panel was replaced and drift velocity for the beam uniformity parameter before and after the MV panel replacement was noticeably reduced.

Efficiency

- Manually parsing, filtering , and plotting 5 years of data across 840+ parameters would take hours. Executing this entire pipeline automatically in 4.5 minutes via an LLM-co-developed architecture (~3.7 hours build time) highlights a massive breakthrough in clinical operational efficiency.

CONCLUSIONS

Transition to Data Driven Quality Management

- Quantitatively proving that over 90% of daily MPC parameters are highly capable provides strong evidence that clinics can safely move away from, time-based QA schedules to reduce clinical QA fatigue.

Proactive Machine Maintenance

- Combining a reactive metric C_p with a predictive metric (30-day drift velocity) establishes a dual-layered audit system. This framework successfully catches progressive machine degradation (such as beam deviations or changes surrounding a component replacement) weeks before clinical thresholds are breached.

Rapid Production and Prototyping

- Collaborating with an LLM enables clinical physicists without formal software engineering backgrounds to architect, validate, and deploy complex automation tools rapidly (~3.7-hour development time), lowering the barrier to clinical software innovation.