QUANTUM INSPECTOR

“Integrated QA Inspection System for Android Automotive HMI”

Author: David Erik García Arenas

Organization: Paradox Cat GmbH

Context: Android Automotive · HMI · QA Tooling

Document type: Internal technical whitepaper

Status: Functional Proof of Concept (PoC)

Confidentiality: Internal use only

Executive summary.

Quality assurance in Human-Machine Interfaces (HMI) built on Android Automotive requires engineers to work simultaneously with multiple heterogeneous artifacts: what is visually rendered on screen, the actual UI hierarchy, structural attributes, selectors used in automation, and runtime execution signals.

In current practice, these artifacts are handled through independent tools. This results in fragmented workflows, high cognitive load, and a strong reliance on manual reasoning by the engineer to reconstruct the correct context during each inspection iteration.

This document presents QUANTUM Inspector, a functional proof of concept developed to validate a concrete hypothesis: that in HMI QA, efficiency and quality are not primarily limited by the power of individual tools, but by the lack of contextual integration between them.

QUANTUM proposes unifying visual inspection, structural UI analysis, and QA reasoning into a single synchronized interface. Results observed during the PoC indicate clear improvements in operational flow, reduced context switching, and increased robustness in selector construction.

These results should be interpreted as trends observed in a controlled validation environment, not as definitive metrics of a production-ready system.

0. Scope of the PoC.

QUANTUM Inspector is explicitly presented as a validation-oriented proof of concept, not as a closed or production-ready tool.

The PoC functionally demonstrates:

- visual and structural correlation between pixel and UI node,
- synchronized inspection of visual captures and XML hierarchies,
- snapshot analysis and review in offline mode,
- heuristic support for selector generation and evaluation.

Out of scope for the current PoC are:

- direct integration with production CI pipelines,
- multi-user or multi-role scalability,
- exhaustive validation across all OEM configurations,
- full automation of QA decision-making.

This framing is intentional and ensures that the presented results are interpreted within the appropriate context.

PoC availability.

The source code corresponding to this PoC is published in a GitHub repository for technical review and internal discussion purposes:

https://github.com/Hawaiiiiii/qa-snapshot-inspector-poc

The repository reflects the current state of the PoC and is intended to facilitate evaluation of the proposed approach and technical design. It is not conceived as a production tool nor as a final architectural reference. The repository does not include OEM-specific dependencies or internal configurations and is limited to illustrating the conceptual and functional design of the system.

1. Introduction.

Modern HMIs based on Android Automotive are highly dynamic systems characterized by:

- deep and frequently changing view hierarchies,
- focus and event management mediated by complex overlays,
- real-time rendering pipelines,
- frequent state transitions,

- strict quality requirements imposed by OEMs.

In day-to-day work, QA engineers are repeatedly required to answer a set of very concrete questions:

- which element is actually being observed on screen,
- which node in the hierarchy it corresponds to,
- how it can be referenced in a stable way,
- which runtime information explains its behavior.

Although system complexity has increased significantly, the tools used to answer these questions continue to operate in isolation, forcing engineers to manually reconstruct context during each inspection iteration.

QUANTUM Inspector emerges as a response to this structural inefficiency, not to replace the engineer's judgment, but to help preserve it throughout the inspection process.

2. Problem statement: fragmented QA inspection.

2.1 Traditional workflow.

A typical QA inspection cycle in Android Automotive includes:

1. visual capture from the device or test rack,
2. export of the UIAutomator XML dump,
3. opening the XML in an independent viewer,
4. manual identification of the corresponding node,
5. copying relevant attributes,
6. manual construction of selectors,
7. parallel consultation of logs,
8. iteration through trial and error.

This workflow introduces three clear structural issues:

- tool fragmentation,
- high context switching,
- error-prone manual reasoning.

2.2 Cognitive and operational cost.

From the perspective of human–computer interaction and cognitive psychology applied to technical work:

- frequent task switching introduces measurable time and error penalties,
- an increased number of alternatives raises decision time (Hick-Hyman law),
- fragmented workflows generate fatigue and reduce executive efficiency.

These costs rarely appear in raw metrics, but they accumulate significantly in real-world QA environments.

3. QUANTUM Inspector: system overview.

3.1 Design principles.

QUANTUM Inspector is based on three core principles:

1. **Single context**: a unified, synchronized mental and visual workspace.
2. **Immediate correlation**: direct linkage between pixel, node, and structure.
3. **Reduction of peripheral decisions**: heuristic support for the engineer without replacing judgment.

3.2 Capabilities validated in the PoC.

The PoC functionally integrates:

- live device mirroring,
- hierarchical UI visualization,
- pixel-to-node mapping,
- assisted selector generation,
- selector robustness heuristics,
- ascension to the relevant clickable node,
- contextual log inspection,
- offline review via snapshots.

3.3 Selector evaluation model.

Selectors are heuristically classified into three categories:

- **unique**: expected high stability,
- **ambiguous**: potential collisions,
- **fragile**: dependence on indices or layout.

This model does not automate decisions, but makes selector quality explicit, reducing downstream errors in automation.

4. Evaluation methodology.

4.1 Design.

The evaluation was conducted as an A/B comparison in a controlled Android Automotive HMI QA environment during the PoC validation period.

The unit of analysis was the complete inspection iteration, defined as obtaining:

- visual confirmation,
- correct node identification,
- a usable selector,
- sufficient debugging context.

4.2 Metrics considered.

Three families of metrics were observed:

- operational metrics (cycle time, flow efficiency),
- quality metrics (selector robustness, detected errors),
- cognitive metrics (context switching, perceived decision load).

These metrics are interpreted as trend indicators, not as absolute values extrapolable to production.

5. Observed results.

5.1 Operational efficiency.

During the PoC, the following were observed:

- consistent reduction in time required to complete an inspection iteration,
- increase in the number of inspections achievable per unit of time,
- reduced need to repeat iterations due to correlation errors.

Exact values vary by scenario, but the trend remains stable.

5.2 Result quality.

Observed improvements include:

- reduced generation of fragile selectors,
- increased visibility of risks prior to automation integration,
- reduction of errors caused by manual correlation.

5.3 Cognitive efficiency.

Workflow unification resulted in:

- fewer tool changes per iteration,
- reduced peripheral decision-making,
- greater continuity of technical reasoning.

6. Discussion.

6.1 Value of integration.

The primary value of QUANTUM does not lie solely in speed, but in preserving mental context during inspection.

By collapsing dispersed artifacts into a single coherent space, the engineer can focus on the actual problem rather than on the mechanics of inspection.

6.2 Implications for QA teams.

As a positive side effect, the following were observed:

- reduced onboarding curve,
- externalization of expert knowledge into inspectable artifacts,
- reduced dependency on individual memory or implicit experience.

7. Limitations and future work.

Current limitations:

- scope restricted to the PoC context,
- heuristic approaches rather than predictive models,
- manual baseline management.

Potential future directions:

- progressive integration with continuous integration systems,
- assisted visual and structural diff analysis,
- collaborative review modes,
- instrumentation of usage metrics.

These directions represent potential evolution paths, not implementation commitments.

8. Conclusion.

QUANTUM Inspector validates a clear hypothesis: in high-complexity HMI environments, QA efficiency and quality depend more on contextual integration between tools than on the isolated power of each tool.

As a proof of concept, QUANTUM demonstrates that an integrated approach can:

- reduce inspection time,
- improve selector quality,
- decrease cognitive load,
- scale QA capacity without increasing team size.

This document does not present a closed solution, but a solid and realistic foundation on which internal QA tooling can evolve.

Acknowledgements.

This work was developed in the context of real Android Automotive QA operations at Paradox Cat GmbH.

I would like to thank **Leandro Lourenço Miranda** for the technical guidance, critical feedback, and professional context provided during the development of the PoC.