Summary Report: Field Tagging and Insights

a. Approach to Field Tagging

To extract structured insights from free-text data, I employed a keyword-based tagging approach across five key fields: Root Cause, Symptom Condition, Symptom Component, Fix Condition, and Fix Component. Each of these columns contained unstructured, free-text descriptions often riddled with inconsistencies in formatting, casing, and punctuation.

The first step was data normalization. I removed special characters and symbols from the relevant columns and converted the text to lowercase to ensure consistent keyword matching. This preprocessing was critical for accurate string matching.

Then, for each tagging field, I defined a curated list of keywords based on domain knowledge or frequently occurring terms. For example:

1) Root Cause was tagged using failure-related terms like 'Leaking', 'Blown', or 'Faulty'.
2) Symptom Condition included observable issues like 'Oil Leak', 'Not Working', and 'Error Codes'.
3) Symptom Component and Fix Component used hardware or part names such as 'Sensor', 'Harness', 'Bulkhead Connector', etc.
4) Fix Condition reflected actions taken, including 'Repaired', 'Installed', or 'Cleaned Out'.

Each record in the dataset was processed row by row, where the combined text of relevant columns was scanned for the defined keywords. If a keyword was found, it was added to the corresponding tagging column. Rows with no matches were labeled as 'Unknown' initially, and later cleaned to leave blank values instead of 'none'.

For multi-label fields like Symptom Condition, where multiple issues could occur, we split the data on commas and expanded into separate columns for better clarity.

b. Potential Insights

This structured tagging opens up powerful analytical opportunities. For instance:

1) Failure Pattern Analysis: By analyzing frequently occurring root causes (like 'Leak' or 'Blown'), organizations can prioritize engineering improvements or supplier evaluations.
2) Component Reliability: Correlating symptoms with components helps identify which parts (e.g., 'Hydraulic lines' or 'Sensors') are more prone to failures.
3) Fix Effectiveness: Observing which fixes are most commonly applied (e.g., 'Repaired' or 'Replaced') can shed light on maintenance efficiency and recurring issues.

4) Predictive Maintenance: If historical symptom and root cause patterns are consistent, machine learning models can be trained to predict failures before they occur.
5) Process Optimization: Frequent tagging of issues like 'Not Mentioned' may indicate documentation or procedural gaps needing attention.

In conclusion, transforming unstructured complaint and repair logs into structured data not only enhances visibility but also unlocks the potential for automation, process improvement, and predictive analytics.