Technical User Guide: QoreStor Dictionary Health Check Script

This document provides a mid-level technical overview of the Bash script designed to extract, display, and analyze system metadata, storage usage, deduplication dictionary utilization, and cleaner process statistics. It is intended for system administrators, support engineers, and DevOps professionals who need to understand the operational logic and structure of the script for maintenance or troubleshooting purposes.

1. Overview

The script collects system and storage-related metrics by invoking built-in management commands (`system --show` and
`ctrlrpc`) and processing their output. It supports optional runtime modes for displaying cleaner statistics specific
to either local or cloud operations.

2. Script Parameters

The script accepts one optional argument:
- `--local`: Displays only Local statistics.
- `--cloud`: Displays only Cloud statistics.
If no argument is provided, the script displays both local and cloud cleaner information by default.

3. Major Functional Sections

System Information

Extracts and displays general system data such as System Name, ID, Version, and Build details.

Main Data Repository Discovery

Identifies the path of the main data storage repository using `system --show`.

Main Data Repository Usage

Retrieves partition usage for the data repository, including device name, size, used, and available space.

Metadata Repository Discovery

Determines the location of metadata storage, including its filesystem and partition usage.

Dictionary Size and Lookup

Calculates the size of the deduplication dictionary (`dict2`) and maps it to an appropriate maximum key count using a lookup table.

Consumed Dictionary Keys

Fetches deduplication statistics via `ctrlrpc show.dedupe_stats` to determine the percentage of used dictionary keys.

Cleaner Stats and Refcount Logs

Traverses metadata directories to summarize cleaner progress by counting phase logs (`pl`, `nl`, `zl`, `cl`).

Cleaner Status Summary

Displays current cleaner statuses (Local and Cloud) and aggregate statistics using `ctrlrpc show.cleaner_adminstats`.

4. Error Handling

The script performs validation checks for key directories, files, and command outputs. If critical information is missing
(e.g., metadata or dictionary files), it prints an error message to `stderr` and exits with a non-zero code.

5. Output Summary

The output includes formatted sections for system metadata, storage usage, dictionary utilization, and cleaner statistics.
Numeric values such as key counts and sizes are formatted with thousands separators where supported (`numfmt`).

6. Dependencies

The following commands and utilities must be available on the system for the script to function correctly:

- system
- ctrlrpc
- df
- awk, grep, cut, sed, find
- numfmt (optional for number formatting)

7. Example Usage

Example 1: Default Mode (show all information)

```
$ ./system_cleaner_report.sh
```

Example 2: Local cleaner mode only

```
$ ./system_cleaner_report.sh --local
```

Example 3: Cloud cleaner mode only

```
$ ./system_cleaner_report.sh --cloud
```

8. Maintenance Notes

Future improvements could include modularizing repeated command parsing logic,
supporting JSON output for automation tools,
and adding verbose/debug modes for diagnostic detail.