ClearDB Documenter

CONQUEST WHITEPAPER

DECEMBER, 2017

Overview

ClearDB Documenter documents an Oracle database and generates a comprehensive report with analysis results of code quality, logical structure of the system, database security and its overall efficiency. It's a Windows 32-bit application working with Oracle Databases, supporting versions 9i - 12c.

ClearDB Documenter supports 59 schema and non-schema object types and can collect such information as object properties, DDL, their source code, database instance properties, parameters, and the like. It works with data at four system levels: Database – Schema – Object Type – Object.

ClearDB Documenter has a built in Code Analyzer fulfilling several purposes:

- Code Illustration – draws diagrams and matrices based on relations between code elements (Flowcharts, Call Trees, CRUD) and DB objects (R&D, ER diagrams).
- Code Review – checks PL/SQL against predefined rules, detecting syntax and logical errors, poor programming practices, orphaned objects, legacy code, etc., and gives recommendations on how to improve the code.
- Code Metrics – calculates system's maintainability and complexity based on industry-standard measurements.

ClearDB Documenter runs a policy-enforced security audit of an Oracle database, scanning it against 630 potential risks. The audit results are outlined in a report, always password protected on a mandatory basis.

ClearDB Documenter compares two database reports or an existing instance snapshot with its current state to indicate differences and changes. The comparison is carried out at four levels (DB, Schema, Object Type, Object) and includes Security Audits as well.

ClearDB Documenter installs with docuVIEWER – a tool to open password protected reports and draw instant diagrams. docuVIEWER is also available as a FREE standalone app.

ClearDB Documenter generates reports in HTML and opens them either in docuVIEWER, or in any standard browser like Google Chrome, Mozilla Firefox, etc.

Security and Control

Database Security Audit scans a PL/SQL system evaluating its security level and looking for potentially weak places. The tool is recommended for ISOs/CISOs to detect vulnerabilities early in the development life cycle and return them to the development team so that breachable code does not get into production. SOC (security operation center) analysts can use it to work out effective plans to strengthen security of enterprises. For in-house and outsourcing Data Protection Officers (DPOs), Security Audit is an instrument to analyze systems in the scope of GDPR compliance and to develop a set of activities to reinforce personal data security.

Management and Consultancy

With the support of 59 schema and non-schema object types, ClearDB Documenter provides a vast database coverage and gives control over the processes undergoing in the system. For Project Managers

this is a strategy development tool to build up roadmaps and arrange for sprint planning. Its automation feature allows a smooth integration into agile environment and saves time documenting large databases in the background mode. Code metrics reflect the system's performance and scalability and allow external DB consultants to produce effective recommendations for improvement, without deep immersion into the product environment. For CIOs the generated reports are a decision-making tool, and they also help share project related information with other stakeholders.

Development and Administration

By gathering information from the four system levels – DB, Schema, Object Type, Object – and structuring it in a hierarchical way, ClearDB Documenter helps System Architects understand the system's logical structure and spot places that need optimization. The code review and quality control feature identifies fragments for refactoring and supports the traditional peer-to-peer code review. Project Leads and Senior Developers use ClearDB Documenter to track changes in the codebase made by other engineers and control development process. Moreover it facilitates the introductory stage for developers and QAs that are new to the project and helps learn corporate coding standards in short term. ClearDB Documenter is effective to detect legacy code, fight with inconsistent use of programming guidelines, and share technical processes beyond the development team.

Core Functionality

DB report generation

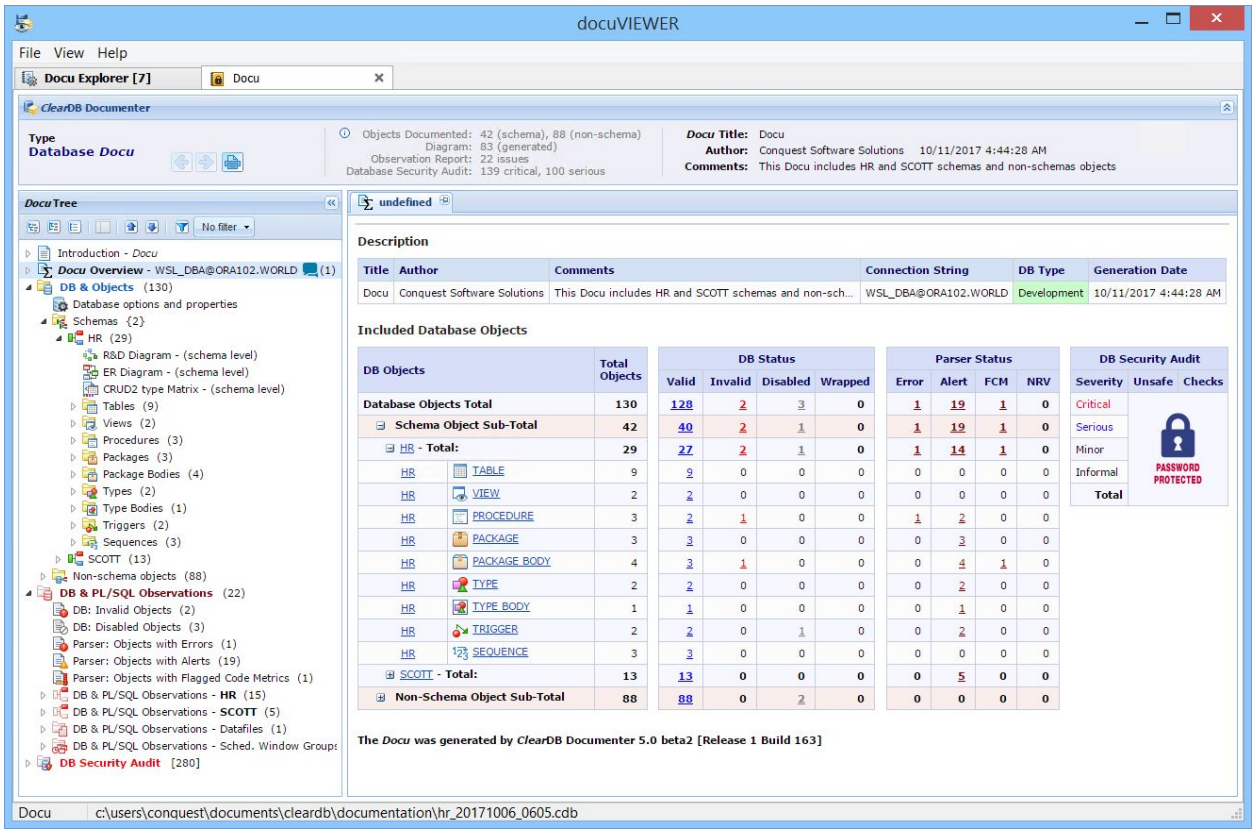

Fig.1 - Database report

Database report generation is the core mechanism of ClearDB Documenter. One report stores data for one Oracle database at a time, collecting information from 59 schema (tables, views, procedures, functions, packages, types, triggers, etc.) and non-schema (contexts, directories, profiles, users, roles, etc.) object types with their properties, DDL, privileges, synonyms, references, dependencies, source code as well as database instance properties, options, and initialization parameters for Oracle versions up to 12c.

The output report is generated in HTML and can be fully or partially password-protected for enhanced security. It has a hierarchical structure gathered at four levels (DB, Schema, Object Type, Object) and is presented as a Report Tree and content page. The report is interactive at all levels and can be customized depending on the use case, meaning you can remove or add any of its sections and adjust the layout.

The report can be viewed in standard browsers and in docuVIEWER – an embedded tool designed to open reports generated by ClearDB Documenter. docuVIEWER is also available as a FREE standalone app and serves to open password protected reports or their parts (Security Audit Report) and to generate instant diagrams.

To use docuVIEWER, it is not mandatory to have ClearDB Documenter installed or to be a license owner. This allows sharing password protected reports with third parties securely.

Database Security Audit

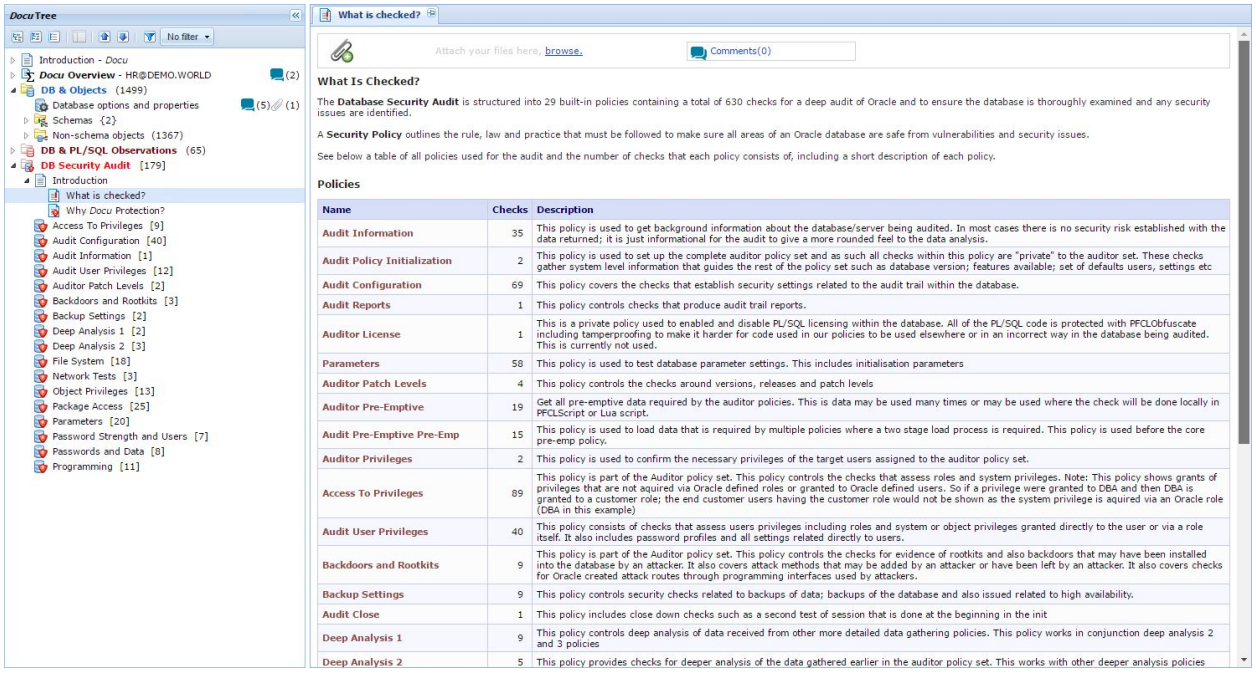

Fig.2 - Database Security Audit

Database Security Audit scans an Oracle database for vulnerabilities using 29 policies and 630 checks. As an output, it produces a comprehensive security report, outlining detected issues and sorting them by

severity. There are four types of severity from “Critical” to “Informal”, where “Critical” calls for immediate action and “Informal” notifies you this might be an issue.

Violated security checks are grouped by corresponding policies with a brief explanation of threat nature and mitigation techniques.

A Security Audit Report is always password protected and can be opened only in docuVIEWER, not in a standard browser. Encryption, as an underlying protection principle, unique password, and ability to view security-related data in a specially designed tool allows for safe storage of business critical data. Such a three-step protection mechanism is applied not only to a security report but to any other report, generated by ClearDB Documenter.

Code Review

Fig.3 - Code Review

During report generation, Code Analyzer checks database objects and their source code against a set of predefined rules, detecting errors, typos, poor programming practices, and other flaws. In the output report, the results are grouped into a separate section “DB & PL/SQL Observations” and are structured by levels. The detected issues are split between the *database issues* (invalid, disabled, wrapped objects, objects with invalid dependencies and warnings) and *parser issues* (objects with errors, alerts, warnings, flagged code metrics, and naming rule violations).

Code review rules are based on the widely used programming practices and can be of three types: Program Structure, Readability, and Maintainability. In the report, the violated rules go along with their rationales and are linked to the line in the source code containing the detected issue.

Code Illustration

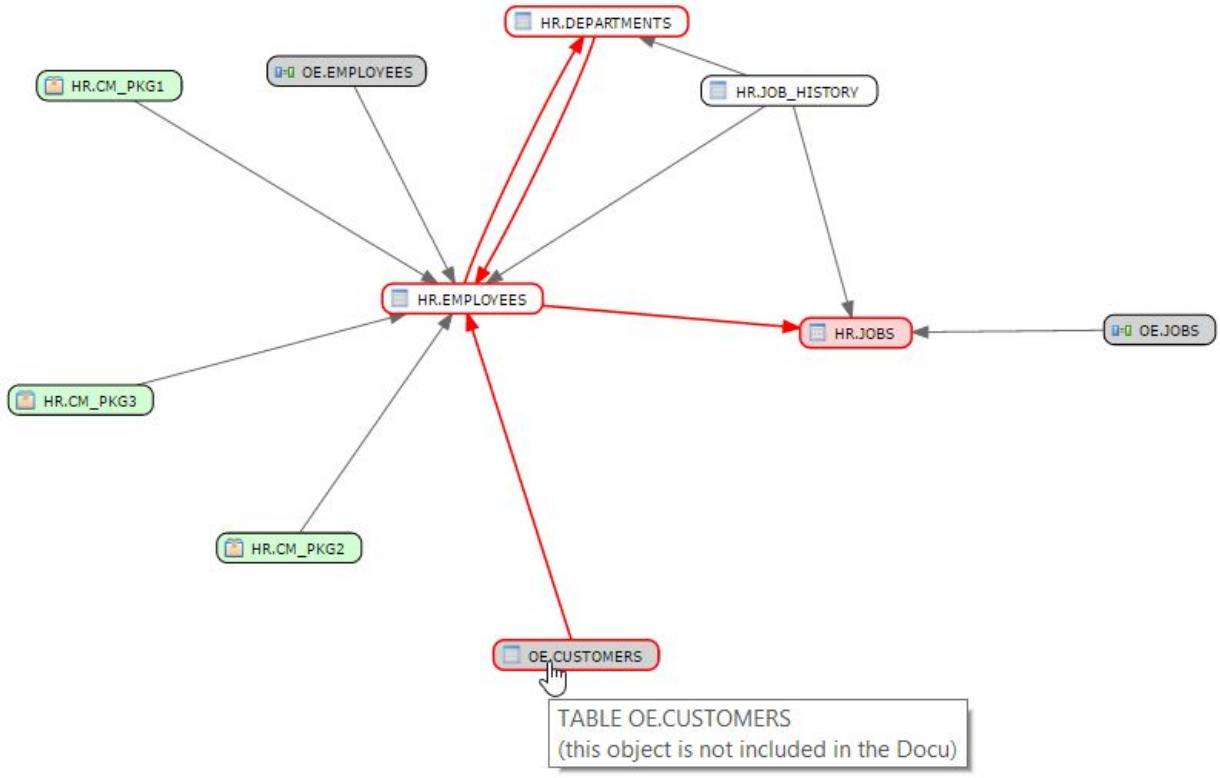

Fig.4 - Code Visualization (R&D diagram)

Along with the running code review, Code Analyzer also draws diagrams and matrices based on object dependencies and relations between code elements. R&D (reference and dependencies) diagrams show how objects in a database correlated with each other. This helps detect orphaned objects and understand the DB structure. ER (entity-relationships) diagrams illustrate relationships of datasets stored in the database and reveal its logical structure. CRUD (Create, Read, Update, Delete) matrices show dataset objects and DML operation where these objects take part. This helps instantly identify problematic areas and, as a result, deliver error-free code. These diagrams and matrices are available at the database and schema levels.

Flowchart diagrams visualize the code execution path of a package or a stand-alone subroutine and display the conditional branches, loops, and jumps, thereby helping understand the opaque logic. The visual patterns find places of possible code refactoring or module restructuring and make the reasons for high values of Cyclomatic Complexity metrics obvious. Call Trees are used to visualize the flow of data between subroutines and database tables. This is performed via executing a DML statement. Such data flows are clickable and they show how subroutines: get data from data objects (table, view) with SELECT INTO statements; how they put data back with INSERT or UPDATE statements; how they delete data with DELETE statements. Flowcharts and Call Trees are generated at the object level only.

To save up space and time, you can generate diagrams on demand right from the report. However, this feature is available only in docuVIEWER. Along with standard formats, diagrams can be generated in SVG, which supports highlighting of the connected diagram elements.

Code Metrics

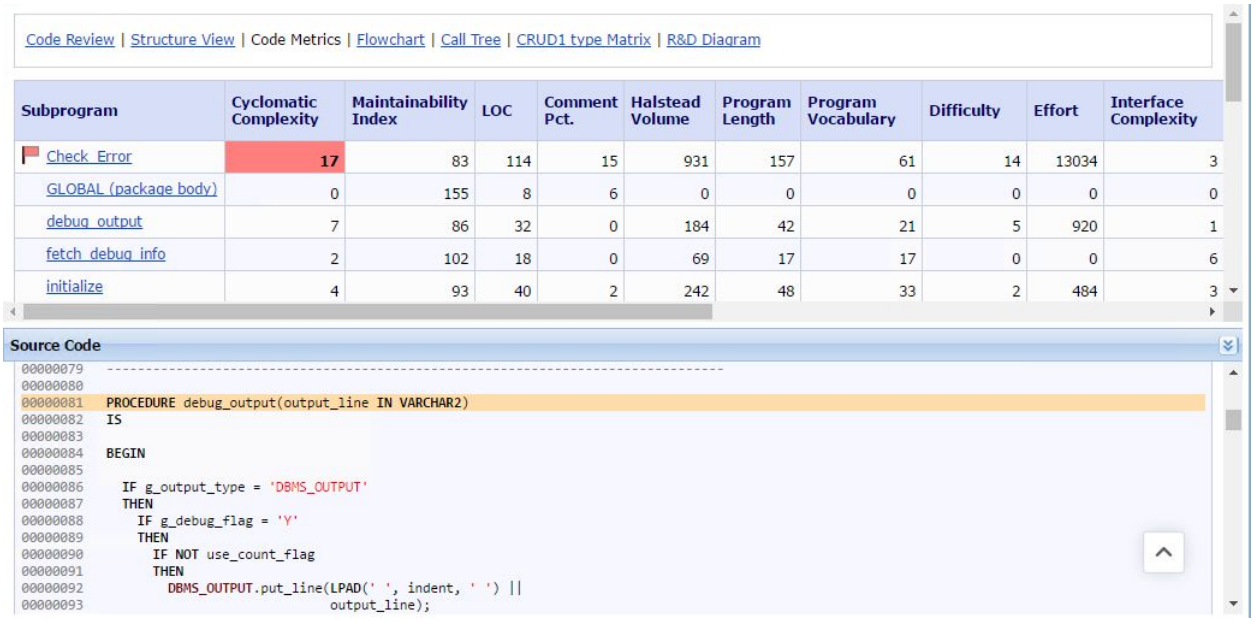

Fig.5 - Code Metrics

Another instrument to check quality of PL/SQL code with ClearDB Documenter is Code Metrics. The measurements are based on four metrics: Maintainability Index, Cyclomatic Complexity, Halstead Volume, and Interface Complexity, which in their turn are computed using more minor metrics such as Lines of Code, Comment Percent; Halstead Program Length, Vocabulary, Difficulty, and Effort, and the like.

These metrics are targeted at the evaluation of software quality by measuring its technical characteristics and identifying areas that can potentially cause problems or errors. They also help detect legacy code and segments for refactoring. The metrics are adjustable, meaning you can set the necessary limits, like maximum Maintainability Index value, and the areas where the limits are exceeded will be marked to draw extra attention.

Code metrics are available in a separate section, where the violations of recommended values are highlighted.

Comparison Report

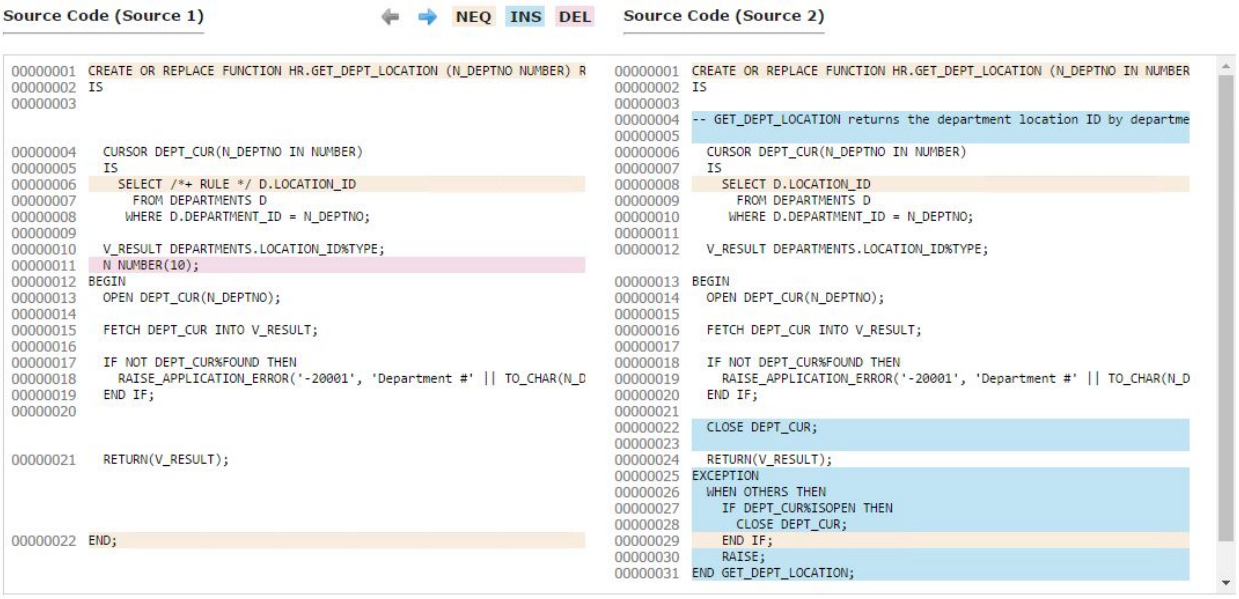

Fig.6 - Database Comparison Report

Another type of reports ClearDB Documenter can generate is a comparison report. It compares two databases reports or an existing instance snapshot with its current state. Technically, the comparison is run between two previously generated database reports. With the report you can trace data movements and alterations, check code review session results, make sure new code has not brought about any security issues, see how Database A differs from Database B.

Comparison is done at four levels: DB – Schema – Object Type – Object, including comparison of the Security Audits. The output information is grouped into several major sections: the Comparison Report, structured hierarchically, and the source reports (if optionally included). The output file is delivered in HTML, similar to other ClearDB Documenter reports, which makes it interactive. For instance, a click on a link under “Invalid” opens the “Invalid Objects” page in the source DB report so that you can view this object’s details.

The comparison report also differentiates between the types of discrepancies found in the compared objects. These can be non-equal properties, parameters existing only in one of the databases or deleted from either of them.

Other Features

Report Generation Modes

Along with generating a database report “from scratch”, ClearDB Documenter provides three more generation modes:

- New report based on existing report. Generates a new database report using DB objects, content, generation, and code analyzer options from another database report. It helps produce a similar report and automatically apply configured settings.

- Actualize content of existing report. Synchronizes a database report you generated some time ago to the current state of a database. It also generates a Comparison yReport as a part of the main report, outlining the differences between the Source and Target reports and represents them in parallel for ease of reading.
- Extract subset from existing report. The “offline” generation mode with no database connection required. Extracts a part of an existing report into an HTML file. For example, if the report contains information about all objects in the database, you can extract information for non-schema objects only.

Automation

Report generation can be automated and scheduled via the Job Manager. Select database objects, configure content, generation, and code analyzer options, and schedule job execution for the preferred time. ClearDB Documenter will run report generation in the background mode. It is also possible to automate configuration of settings by saving them to a template.

Extract DDL

ClearDB Documenter extracts DDL of database objects into one file or for each object separately. The feature does not require database connection and is applicable only to those reports that were initially generated with the “Extract DDL” option enabled.

Summary

ClearDB Documenter is a multifunctional application designed for IT-auditors, architects, DBAs, CISOs and migration consultants, helping them create detailed logical snapshots and run security audits on Oracle databases. With the Code Analyzer at its core, it also checks code quality and identifies potentially faulty areas at all system levels, providing visual representation of the database and code structure. The Security Audit engine extensively validates database security and helps avoid data leaks. Automation features allow for successful integration into an agile workflow. It’s a key tool to track changes to DB structure, provide comprehensive security self-audits, enable distributed and disconnected architecture reviews, and mitigate development and migration risks for complex database instances.