

COBOL - STG, Inc. - Department of Defense Activity Address Directory (DODAAD)

STG, Inc

Department of Defense Activity Address Directory (DODAAD)
Defense Logistics Agency

HP COBOL to C++

History: STG, Inc. awarded a contract to TSRI for the transformation, re-factoring and documentation of the DODAAD system, along with re-hosting the system's database.

The DODAAD system ran on an HP-UX OS based system utilizing a combination of VT100 escape sequences for direct screen and keyboard I/O supported by a large number of platform specific extensions to ANSI standard COBOL Display and Accept statements. The COBOL source code contained embedded SQL Screen Displays and Screen Accept statements that were tightly interleaved with the business logic.



Challenge: TSRI resolved challenges for the DODAAD project in four distinct areas; code transformation, re-factoring, data manipulation, and the graphical user interface (GUI). The transformation of DODAAD code from COBOL into C++ was accomplished with 100% automation. The DODAAD business logic, data base logic and GUI logic were segregated during this transformation into distinct C++ classes. Re-factoring was used to further partition the application into multiple tiers, to simplify control structure, and to eliminate redundancy. Automated GOTO elimination was accomplished using local and global control flow analysis during a post-transformation re-factoring phase. GOTOs that violated perform ranges were similarly factored out of the control structure in the transformed C++ application logic. All HP UX SQL calls were transformed into ODBC compatible SQL and into an ODBC manipulation layer, providing the application with data base independence and separating the data base access from business logic. The HP UX COBOL Display and Accept statement were transformed and re-factored into a distinct layer of C++ classes and methods. This separated the GUI functionality from the business logic. The segregation of GUI logic prepared the modernized system for a future web based interface when needed.

Results: The partitioning of DODAAD into separate Business Logic, Data Base, and GUI tiers, was achieved through automated re-factoring, enabling the customer to satisfy stringent DTIC architectural requirements. The DODAAD modernization project illustrates the effectiveness of automated transformation and re-factoring for migrating monolithic mainframe applications into object-oriented architectures capable of platform independence. TSRI awaits deployment of the DODAAD pilot by STG and the government and expects to begin work on phase-2 in early 2005.