
AMISYS Advance Migration

**AMISYS Surround Code
OPENTURBO IMAGE Translator**

A Technical White Paper



IMAXSOFT Corporation
20410 Town Center Lane Suite 295
Cupertino, CA 95014
(408) 253-8808

Introduction

One of the major hurdles to the timely migration to AMISYS Advance has been the sheer number of COBOL surround code that must be converted to Pro*COBOL. For a typical AMISYS site, upwards of several hundred IMAGE-based COBOL programs must be individually ported, often creating a sizable burden on the migration schedule. This paper describes the automation tool built by IMAXSOFT that address the issue.

OPENTURBO IMAGE Translator

IMAXSOFT has successfully automated the IMAGE-to-Pro*COBOL conversion process. By leveraging it's core OPENTURBO technology, the IMAXSOFT Translator offers the following advantages over competing migration approaches:

- High Quality Pro*COBOL Code
- Consistent Code
- Maintainable Code
- Ease of Debugging and Testing
- Flexible to Change
- Predictable Schedule

High Quality Pro*COBOL Code

While many vendors market a generic migration tool sets, few can truly claim to have a large team of dedicated Pro*COBOL experts to work on your

particular migration. IMAXSOFT has taken years of accumulated Pro*COBOL porting experience and know-how, and codify them into the Translator. The Translator properly handles simulation of IMAGE dataset locking, batch retrieval, batch updates, and prepares SQL, to name a few.

Consistency

One challenge of working with several hundred programs is achieving consistency across the translation. The Translator will guarantee consistency in program structure, naming conventions (variable and paragraph names), as well as code formatting.

Maintainability

Consistency is required for maintainability, but is not sufficient. The converted programs should also conform to structured programming best practices and promote modularity. To meet that end, the Translator is designed to generate well-structured, succinct, and readable code. To keep file sizes small and maintainable, the Translator also separates main subroutine code from actual SQL code.

The generated code is very flexible for future update and/or enhancement in HP-UX and Oracle native environment.

Ease of Debugging and Testing

The converted program should be structured in a way that allows piece-meal debugging and testing. The Translator generates all Pro*COBOL SQL code into a separate file, making it easy to write standalone driver program to separately test and debug database access from core business logic.

Flexibility to Change

Migration process is often not a linear endeavor. At times, it is necessary to simultaneously support both the 3000 and the HP-UX and to continuously bring enhancements forward until the 3000 Legacy Application completely switched over to HP-UX. Such rework may add substantial risk to the migration schedule. By leveraging the repeatable conversion process of the Translator, changes on 3000 can be systematically brought forward, reviewed, and applied on a case-by-case basis.

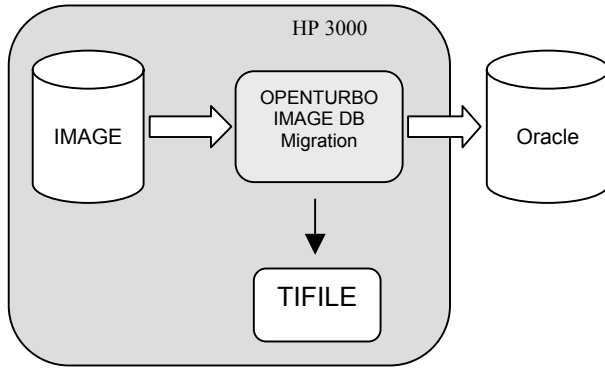
Predictability of Schedule and Budget

Manual brute-force approaches to migration poses great risks to the predictability of time and budget on both the client and the vendor. Since training effectiveness, resource turnover, and productivity are difficult to gauge, both parties are left with project plans that may be difficult to achieve. This poses two significant problems. First, with publicly announced deadline looming, fresh less-skilled resources may be

brought on with dire consequences. Second, as projects exceed fixed-bid caps with no end in sight, vendors may simply start to tune out. Since the Translator automates the bulk of the conversion process, human variables are reduced and overall schedule estimates become markedly more accurate.

The OPENTURBO Technology

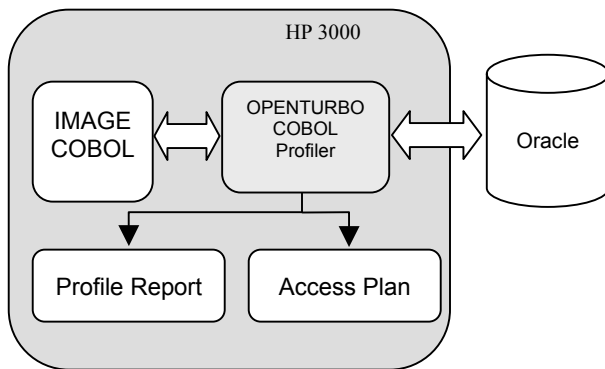
Fundamental to the success of the Translator is the use of OPENTURBO profiler technology to collect run-time database access plan. By cross-referencing true database access patterns, the Translator is able to pin-point the actual DBGET modes used by the programs and generate program-specific SQL queries. Without the profiler, it would be quite difficult to deduce from source code all the modes and keys being used. And without such information, one would be forced to implement SQL queries for all potential mode-and-key combinations. In practice, this would lead to very large programs, with thousands of lines of dead-weight SQL code that may not be utilized during execution, and make it quite difficult to debug and maintain.



Using the OPENTURBO Translator

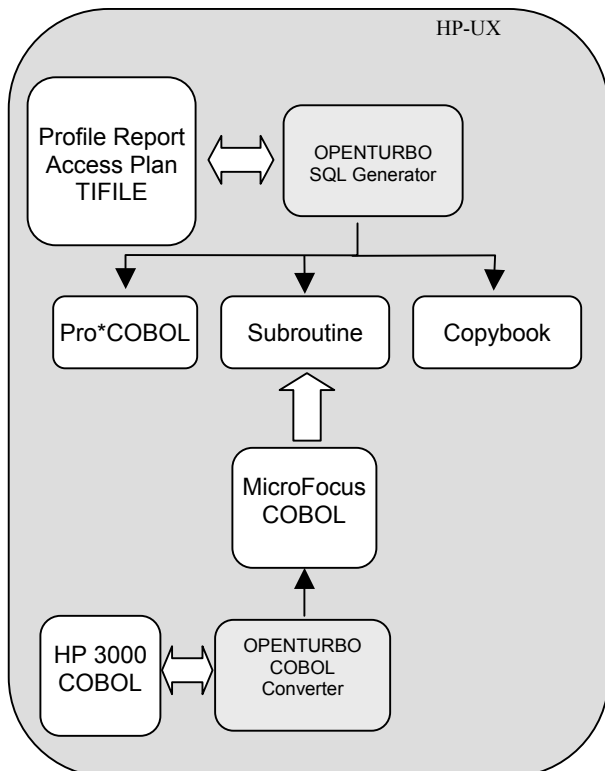
Step 1: Generate the IMAGE-to-ORACLE Mapping File, TIFILE.

The TIFILE is used for the profiler step below, which records the actual run-time data access patterns.



Step 2: Using OPENTURBO Profiler, generate Profile Report and Access Plan files.

The Profile Report and Access Plan provide OPENTURBO SQL Generator the necessary run-time access plan for the most accurate, succinct, and optimized SQL translations.



Step 3: Using OPENTURBO SQL Generator, generate ProCOBOL SQL file, subroutine file, and copybook. Use OPENTURBO COBOL Converter and Intrinsic library to migrate original COBOL program to MicroFocus on HP-UX.

The generated SQL code is organized into three parts: a copybook file, a Pro*COBOL file holding all the SQL statements, and a COBOL subroutine. The COBOL Converter automatically replaces IMAGE calls to the appropriate generated Pro*COBOL subroutines.

Conclusion

Due to the complexity and volume involved, the conversion of IMAGE COBOL surround code to Pro*COBOL can easily become a risk area of your Advance migration. IMAXSOFT's OPENTURBO Translator can help mitigate your risk. Unlike alternative solutions on the market, the Translator is designed to intelligently generate only the SQL for the modes used by your programs.

IMAXSOFT
Suite 295
20410 Town Center Lane
Cupertino, CA 95014
Tel: 408-253-8808
Fax: 408-253-4008
www.imaxsoft.com

