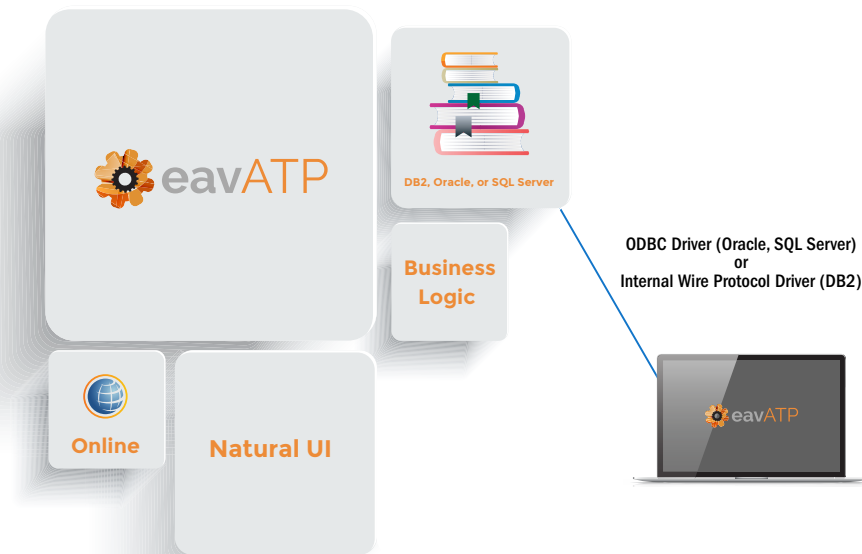


REPLATFORMING NATURAL & ADABAS IN WINDOWS

As an alternative to the re-engineering and automated conversion solutions available, application replatforming dramatically reduces the time frame for the replacement cycle while retaining the legacy application codebase. Replatforming offers full preservation of legacy assets and business rules, liberation of legacy data, & migration to modern infrastructure.

HOW IT WORKS



The Application Transparency Platform™ (ATP™) is a revolutionary Natural Adabas Replatforming solution that provides an end-to-end strategic solution for lower license costs, higher productivity and full preservation of your legacy assets and business rules. As an alternative to the re-engineering and conversion solutions available for Natural and Adabas, ATP dramatically reduces the time frame for the replacement cycle and accelerates return on investment.

ATP gives you the freedom to choose a new platform without the high cost and long project timeframes associated with other solutions.

- Reduce or eliminate mainframe costs with a short-term project that provides fast ROI
- Eliminate expensive maintenance/license fees
- Retain your Natural applications, expertise and skilled teams
- Upgrade your technology and computing platform without impacting your business
- Replace Adabas with industry-standard easy-to-access relational databases
- Add off-the-shelf RDBMS reporting, query and business intelligence tools



← CASE STUDIES



VIDEO OVERVIEW

WHITEPAPERS →



REPLATFORMING NATURAL & ADABAS IN WINDOWS

GETTING TO THE GOAL LINE

Portfolio Analysis

The Portfolio Analysis is a complete research and analysis project that outlines all mainframe application and database refactoring candidates. Components are classified and listed in detail. Notes are attached to components requiring special attention during the refactoring process. All application components are inventoried, classified by language, and cross-referenced. Missing components are collected and added to the inventory. Duplicate components residing in multiple customer repositories are eliminated from the inventory. The Portfolio Analysis results in a complete understanding of the current processing environment.

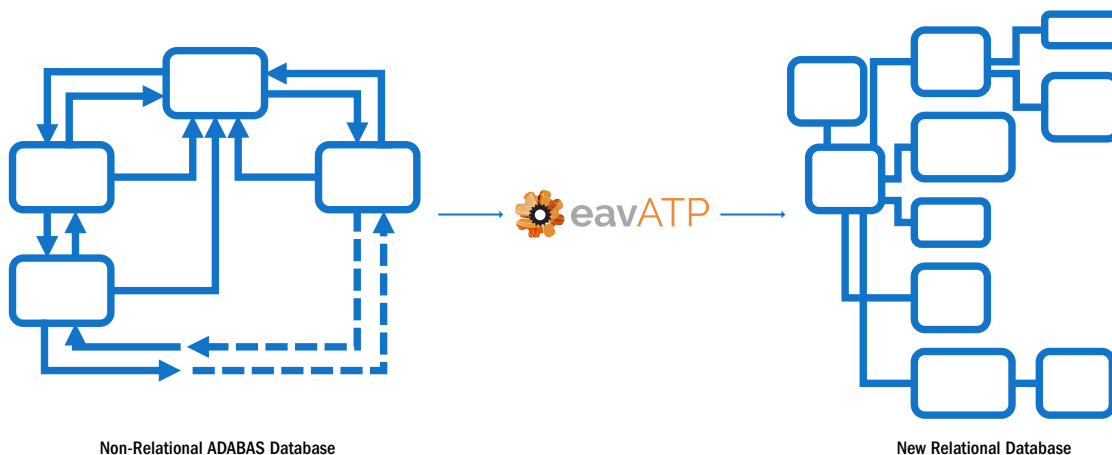
This phase also includes discussing and reviewing the overall system test strategy and the division of the converted code into work packets.

Modern Systems will define a set of topics during the Portfolio Analysis phase that must be addressed prior to refactoring. In order to speed the modernization process, these topics should be addressed by those team members who are best suited to understand the topic, the solution options, and any changes or activities that are required to address the Area of Concentration.

Once the Areas of Concentration are identified, customer teams and Modern Systems will address the areas about which they are most knowledgeable and for which they are best suited to implement a solution. Any additional Areas of Concentration that are identified during the course of the Project will be addressed and assigned in the same manner, to the most appropriate team.

Database Conversion

The fully automated database conversion process refactors your Adabas databases to the new database technology of your choice: SQL Server, Oracle, or DB2. The ATPconvert process includes the generation of the new database definitions, population of the tables with the current Adabas data and allows for data cleansing activities.



REPLATFORMING NATURAL & ADABAS IN WINDOWS

GETTING TO THE GOAL LINE

Lift & Shift

ATP allows you to replatform your applications without affecting your business team's user experience. The functionality remains the same so no training is required. All screens retain the same look and feel. Even the keystrokes, tabbing and PF-keys remain unchanged. It is easy to update the default colors and backgrounds, but user interface changes are optional.

Natural code is not converted, but instead it is imported into ATP for execution and no changes are required to access the new relational databases. In fact, Natural development teams can continue to maintain and enhance the applications in a language they are accustomed to. In addition, modern web applications can access the relational databases directly, allowing you to benefit from industry-standard SQL-based reporting, query, and business intelligence tools.

Test, Refresh, Deploy

Modern Systems tests a subset of the refactored code using a test plan with documented test scenarios (test scripts) provided by the customer. The customer will write the tests and run them on the existing system to capture and record the expected results. Modern Systems will then run the tests against the same data on the refactored system, then identify, investigate, and fix discrepancies in the expected behavior of the modernized application. Pre-delivery testing will consist of test cases picked from all available functional test cases in order to be representative of different parts of the applications.

Once pre-delivery testing is complete and any discrepancies in application behavior are resolved, Modern Systems performs a code refresh to ensure that any changes that took place in the legacy application environment during the refactoring process are accounted for, refactored into the target language and environment, and tested.

The Modern Systems teams work closely with our customer teams to ensure a smooth and error-free transition night or weekend. A primary team may be onsite during the cut-over, and all other Modern Systems team members are on-call should assistance be required. Our teams also provide any required assistance during the 90-day warranty period following the deployment.

We also offer post-application support of the modernized application, available at a fixed rate or in hourly buckets purchased as needed by the customer.

Maintain

The advanced development platform provides features beyond those available to Natural developers today. With ATP, Natural developers can document their applications, debug processing at run-time, execute in dual run-mode, benefit from version control, and continue Natural development and maintenance using enhanced features and a GUI interface.