

Data and Test Management

X-Datatest provides effective data and test management tools for IBM i

What's Inside

A description of X-Datatest, the Fresche Legacy tool that prepares test databases, develops test cases and provides output comparisons.

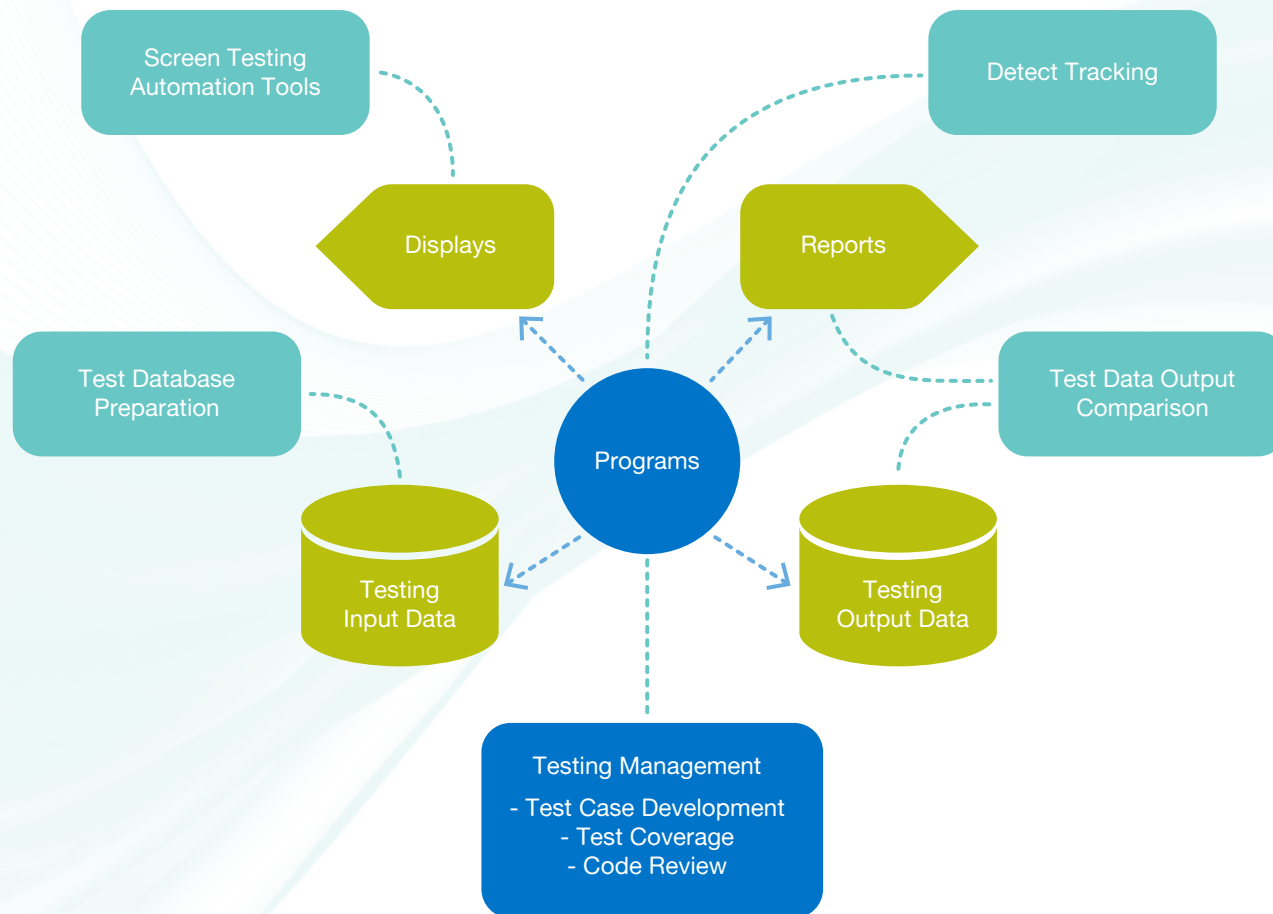


X-Datatest

Data and test management on IBM i

Helping you effectively test large, undocumented applications on IBM i

IBM i applications are often heavily modified over the years, and they typically do not come with a complete library of test cases or specifications. Building on X-Analysis, the world's leading IBM i code analysis toolset, Fresche has developed the industry-leading approach to testing IBM i applications. X-Datatest provides a fully-equipped framework and appropriate tools that configure and automate test cases; it gives you complete independence to carry out advanced testing operations.

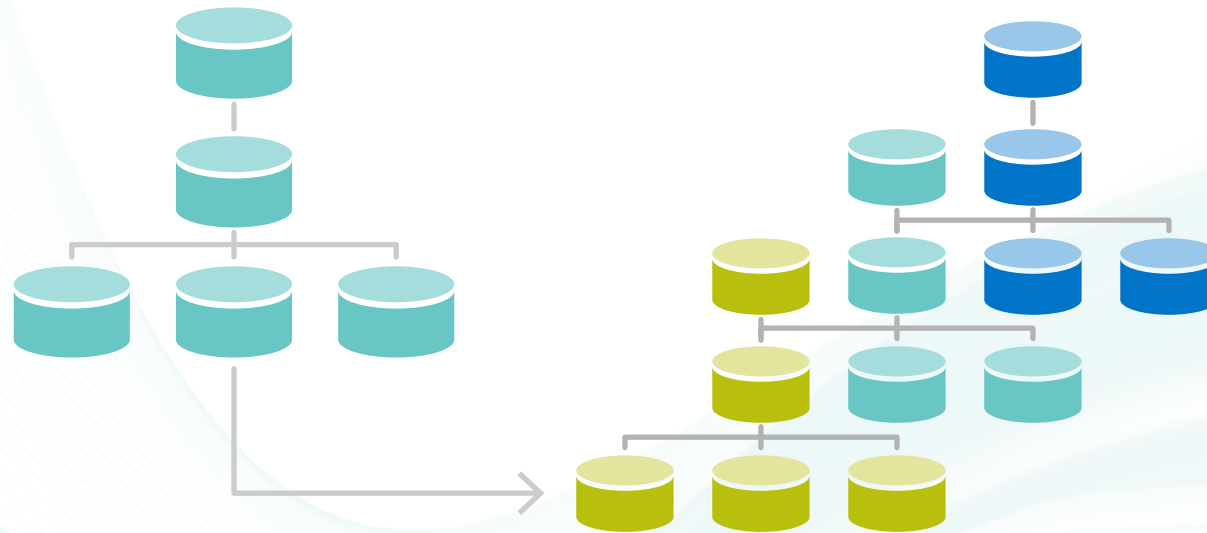


X-Datatest Features

- ▶ Creates complete test data subsets from live data based upon the existing application data model.
- ▶ Verifies and reports on referential integrity of entire live data based upon existing application data model.
- ▶ Ages dates forwards and backwards — days, months or years
- ▶ Scrambles sensitive data while retaining integrity of special formats such as telephone numbers, names, Zip/Post codes etc.
- ▶ Sets up test environments
- ▶ Defines and populates checkpoints
- ▶ Repeatedly executes the tests
- ▶ Compares the resulting data in files/tables/reports
- ▶ Reviews the differences in a user-friendly manner
- ▶ Provides screen testing script automation

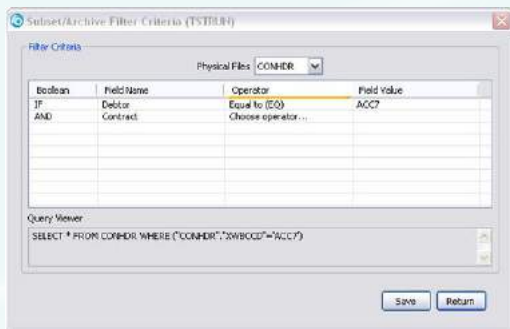
X-Datatest Automated test database preparation

Create controlled database subsets for accurate testing

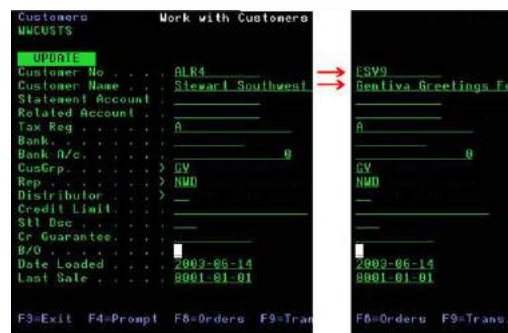


In complex situations with multiple data environments for development, testing and production, it is difficult to maintain data integrity for testing purposes. Copying production data is the most common practice, but keeping data current for each discrete project and environment combination is a very labor-intensive task, not to mention the impact it has on performance and disk capacity. A test database containing only the files and related data used to test a development or maintenance project improves testing performance and accuracy and long term coding quality.

X-Datatest automates the building and data refreshing of test databases. X-Analysis cross reference meta-data is used to map all data files related to a maintenance task, while the derived foreign keys of the X-Analysis Data Model are used to ensure that records selected for testing have all related records from related files. The testing databases have predictable data with referential integrity. Refresh of data functions can be integrated into managed testing procedures, or with SCM tool task management processes.



Filter Criteria function lets you easily extract a coherent subset of data for testing



Masking feature lets you scramble sensitive data while retaining integrity of special formats

| View Coverage Report For TSTRUN | |
|---------------------------------|---|
| File/Field/Value | Comment |
| CONDET | |
| XWRICD | 2 of 3 TRNTYP values exist in test data |
| INV | Found in test data |
| CRN | Found in test data |
| SUN | Missing from test data |
| CONHDR | |
| PERSON | 3 of 6 SLMEN values exist in test data |
| STU | Found in test data |
| JKL | Found in test data |
| NWD | Found in test data |
| MTT | Missing from test data |
| KAN | Missing from test data |
| SDS | Missing from test data |

Automated customizable reports help you keep track of test coverage

X-Datatest Automated regression testing

Benefit from automated high volume parallel database and report comparisons

Comparison of XTSR3 Results with XTSR0 - App Area: TSTRUN

Result Comparison

- Database File
 - Differences in Contract Detail(CONDET)
 - Differences in Stock Balances(STKBAL)
 - Difference in record 126: "Onhand Quantity"(XWBHQT) XTSR0: 0.0 - XTSR3: 1.0; "Pur Ord Balance"
 - Difference in record 128: "Onhand Quantity"(XWBHQT) XTSR0: 0.0 - XTSR3: 1.0; "Pur Ord Balance"
 - Difference in record 130: "Onhand Quantity"(XWBHQT) XTSR0: 0.0 - XTSR3: 1.0; "Pur Ord Balance"
 - Difference in record 132: "Onhand Quantity"(XWBHQT) XTSR0: 0.0 - XTSR3: 1.0; "Pur Ord Balance"
 - Difference in record 127: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 129: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 134: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 135: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 136: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 137: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 138: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Difference in record 139: "Store"(XWAACS) XTSR0: UK - XTSR3: SWI; "Onhand Quantity"(XWBHQT)
 - Differences in Store Master(STOMAS)
 - Records appearing in XTSR3 (that are not in XTSR0)
 - "Store"(XWAACS): SWI
- Spool File
 - Differences in (BALPRDSPL)
 - Records appearing in XTSR0 (that are not in XTSR3)
 - Page No: 0
 - (Line 0) 000001 RSA ROBIN1 20.0000 .000
 - (Line 0) 000027 RSA
 - (Line 0) 000086 UK
 - Records appearing in XTSR3 (that are not in XTSR0)

Detail Comparison of Test Result with XTSR0 for File STKBAL - Rec No 126

| Field | Test value | Base Value |
|-------------------------------|------------|------------|
| Store (XWAACS) | RSA | RSA |
| Product (XWABCD) | 000100 | 000100 |
| Grp 1 (XWAGCD) | 10 | 10 |
| Grp 2 (XWAHCD) | 10 | 10 |
| Grp 3 (XWAICD) | 30 | 30 |
| U/M (XWA2CD) | EA | EA |
| Onhand Quantity (XWBHQT) | 1.0 | 0.0 |
| Pur Ord Balance (XWBKQT) | 1.0 | 0.0 |
| Stk Bal Sales Order (XWBM...) | 0.0 | 0.0 |
| Stk Bal Production (XWVFQT) | 0.0 | 0.0 |

Enhancing or changing legacy code can introduce errors in existing code. These software bugs or "regressions" can be very difficult to find and labor intensive to test for in large complex systems. An effective way to test for regressions is to analyze data before and after code changes. This analysis of potentially thousands of fields and records after every code change is time consuming and requires significant manpower. Many companies wait for regressions to appear in production environments, but they sometimes only appear years later.

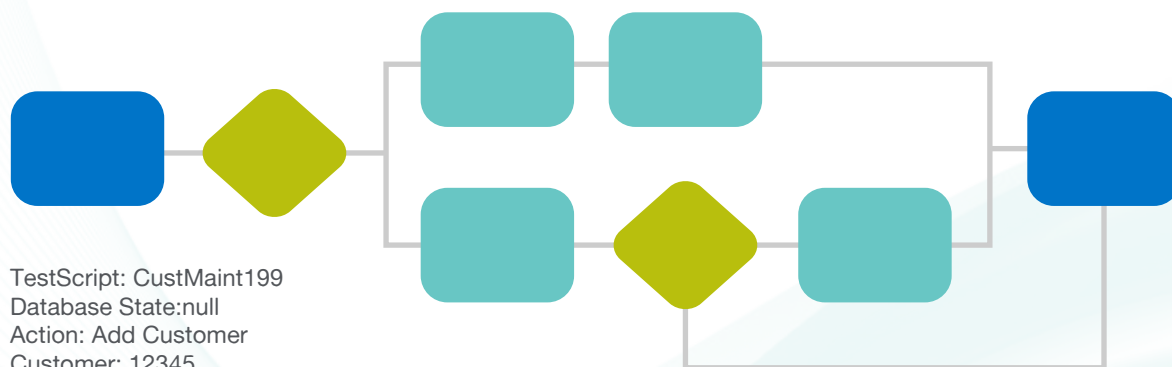
X-Datatest provides you with a high-level view of test results and comparisons, enabling you to quickly home in on potential defects.

Important to this sort of analysis is the ability to filter out expected and unavoidable differences for such things as time-stamps. X-Datatest gives you complete flexibility for filtering out expected discrepancies so you can focus your energy and time on meaningful indicators of defects.

Drill-downs provide a high level of comparison detail

X-Datatest Test Case Development

Test case functionality helps construct high quality suite of tests



```
TestScript: CustMaint199
Database State:null
Action: Add Customer
Customer: 12345
Name: Tom Brown
Cust Type: blank
Expected Result:
Error message: Cust type...
```

Developing test cases that maximize test coverage of large amounts of code is challenging and time consuming. Legacy systems often contain many thousands of business rules and legacy databases often contain many surprises.

Through the use of recovered business rules, Pseudo Code, and data content analysis, the general purpose features of X-Analysis can help you efficiently construct a high quality suite of test cases.

| CUSNMN1: (Changed on 2010-06-05) | CUSNMN1: of 2010-07-02 |
|----------------------------------|-------------------------------|
| C *ENTRY | C *Entry parameters |
| C PLIST | C PLIST |
| C PARM | C PARM |
| C KWBCDD | C KWBCDD |
| C KWBTXT | C KWBTXT |
| C *Initialize message subfile | C *Initialize message subfile |
| C ROVE '1' | C ROVE '1' |
| C *CUSTOMNT1 | C *CUSTOMNT1 |
| C ZDPOM | C ZDPOM |
| C KWBCDD | C KWBCDD |
| C SWBCDD | C SWBCDD |
| C EXSR | C EXSR |
| C GETREC | C GETREC |
| C *-?set date | C *-?set date |
| C ROVE *DATE | C ROVE *DATE |
| C ZIDATE | C ZIDATE |
| C MOVE *YEAR | C MOVE *YEAR |
| C ZYR | C ZYR |
| C ROVE '1' | C ROVE '1' |
| C FILLR1 | C FILLR1 |
| C ROVE *MONTH | C ROVE *MONTH |
| C ZMTH | C ZMTH |
| C ROVE '1' | C ROVE '1' |
| C FILLR2 | C FILLR2 |
| C ROVE *DAY | C ROVE *DAY |
| C ZDAY | C ZDAY |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C DOWEQ '0' |
| C *IN12 | C *IN12 |
| C ANREQ '0' | C ANREQ '0' |
| C *C? | C *C? |
| C *IN03 | C *IN03 |
| C IFEQ '1' | C IFEQ '1' |
| C *IN12 | C *IN12 |
| C OREQ '1' | C OREQ '1' |
| C LEAVE | C LEAVE |
| C ENDIF | C ENDIF |
| C *Action command keys | C *Action command keys |
| C *IN28 | C *IN28 |
| C IFEQ '1' | C IFEQ '1' |
| C ROVE '0' | C ROVE '0' |
| C *IN12 | C *IN12 |
| C ELSE | C ELSE |
| C *Change main data | C *Change main data |
| C EXSR | C EXSR |
| C CHGMA1 | C CHGMA1 |
| C ENDIF | C ENDIF |
| C ENDDO | C ENDDO |
| C * | C * |
| C **Main loop | C **Main loop |
| C *C | C * |
| C *IN03 | C *IN03 |
| C DOWEQ '0' | C |

Recovered business rules can help you construct a high quality suite of test cases.

[illegible]

Before and after data content analysis

At a Glance...



X-Datatest

X-Datatest

The Fresche Legacy product that provides effective data and test management tools for IBM i.



Authorized Dealer
555 Winderley Place Suite 300
Maitland, FL
(786) 206 6512 - (786) 513 2806
info@greenlt.com
www.Greenlt.com

Fresche Legacy — www.freschelegacy.com

US:
9245 Laguna Springs Drive,
Suite 200
Elk Grove, CA, 95758

Australia:
9/622 Ferntree Gully Road
Wheelers Hill
VIC 3150
Australia

Canada/Corporate Office:
995 Wellington, Suite 200
Montreal, CAN,
H3C 1V3

India:
Atrauli, Gaurabagh,
P.O. Gudumba, Kursi Road,
Lucknow 226026
Uttar Pradesh, INDIA

X-Datatest Functionality Summary

- ▶ Sets up test environments
- ▶ Defines and populates checkpoints
- ▶ Repeatedly executes the tests
- ▶ Compares the resulting data in files/tables/reports
- ▶ Reviews the differences in a user-friendly manner
- ▶ Provides screen testing script automation
- ▶ Creates complete test data subsets from live data based upon existing application data model.
- ▶ Verifies and reports on referential integrity of entire live data based upon existing application data model.
- ▶ Ages dates forwards and backwards: days, months or years
- ▶ Scrambles sensitive data while retaining integrity of special formats such as telephone numbers, names, Zip/Post codes etc.



About Fresche Legacy

As a leading expert in legacy management and modernization, Fresche Legacy helps enterprise organizations transform their business to improve financial performance, increase market competitiveness, remove risk and add business value. Our team of experts has successfully completed hundreds of transformation projects within the most complex enterprise environments, helping organizations future-proof their business by modernizing their business processes, technologies, infrastructure, and methodologies. Committed to 100 percent customer satisfaction, Fresche Legacy's services and solutions span the complete legacy modernization spectrum from concept to maintenance, and include Discovery Services, Modernization Solutions, and Application Management Services & Transformation. For more information about our company, visit us on the Web at www.freschelegacy.com

X-Analysis Professional is the foundation of the full X-Analysis toolset, a suite of productivity tools for your IBM i applications. Following is a brief description of all modules in the suite:



X-Analysis Professional: The foundation tool, with all of the basic functionality.



CA 2E Analysis: Everything required to analyze and document CA 2E applications.



Application Process Mapping: Business rule and relational data model extraction and documentation of application processes and flows.



Application Modernization: RPG, COBOL and CA 2E (Synon) automatically converted to Java.



Audit, Quality and Change Management: Auditing of core application functionality - including design, quality and complexity - to identify and change problematic areas in the application.



Database Modernization: Automated conversion of DDS to DDL, including creation of constraints, long field names and views.



Data and Test Management: Analysis of data quality; data archiving, data subsetting and data masking. Test data automation and management.



Open Systems Analysis: Cross-referencing and documentation of Java, C#, PHP, VB/VB.NET and PowerBuilder.

How to get on board: X-Analysis products are available as a complete package or as individual modules. At Fresche Legacy, we work closely with you to assess your needs and recommend the best solutions. To get started, contact us using the information below:

www.freschelegacy.com | info@freschelegacy.com |
1.800.361.6782 (US, Can) | 00 800 361 67 82 0 (Belgium, France Germany, UK) | 0011 800 361 6782 0 (Australia)