The Industrialization of Mainframe-to-Cloud Migration

How Automation Dramatically Reduces Risk and Cuts Implementation Time

amdocs
make it amazing

Astadia
An Amdocs Company

Notice

Astadia, Inc. makes no warranty that the content of this document is timely or complete; or is free of omissions, inaccuracies, typographical errors, or other errors. All contents of this document, including but not limited to the text and images contained therein, are made available on an "as is" basis without any warranty, express or implied, of any kind, including the implied warranties of merchantability, title, non-infringement, quality, or fitness for any particular purpose. Certain sections of this document may contain forward-looking statements that are based on product management's expectations, estimates, projections and assumptions.

Words like "plans," "intends," "expects," "believes," "future," "estimates" and variations of these words and similar expressions are intended to identify forward-looking statements. These statements are not guarantees of future performance and involve certain risks and uncertainties, which are difficult to predict. Therefore, actual future results and trends may differ materially from what is forecast in forward-looking statements due to a variety of factors.

Trademarks

Astadia, the "A" logo, TestMatch, DataMatch, DataTurn and CodeTurn are trademarks or registered trademarks of Astadia, Inc. These trademarks may not be used without the permission of Astadia, Inc. The absence of a product, company, or service name or logo from this list does not constitute a waiver of the trademark or other intellectual property rights of Astadia, Inc. concerning that name or logo.

Other trademarks that appear in this document are used for identification purposes only and are the property of their respective owners. These marks may not be used without the permission from these owners.

Astadia joined Amdocs Cloud division in November 2023.

Table of Contents

1. A New Approach to Mainframe Migration 4

2. Industrialization: Three Levels of Automation 5

I. Automated transformation 5

II. Automated testing 5

III. Automated workflow 6

3. Introducing the Astadia FastTrack Factory 7

4. Game-Changing Technology 8

5. An Overview of the FastTrack Factory Concept 10

6. Automated Migration and Testing Tools 12

I. CodeTurn: Automated Code Transformation 12

II. DataTurn: Conversion and Transformation of Legacy Databases 14

III. TestMatch: Automated Testing based on Actual User Activity 17

IV. DataMatch: Purpose-Designed to Validate Migrated Legacy Data 18

7. A Holistic Approach to Automated Migration 19

1. A New Approach to Mainframe Migration

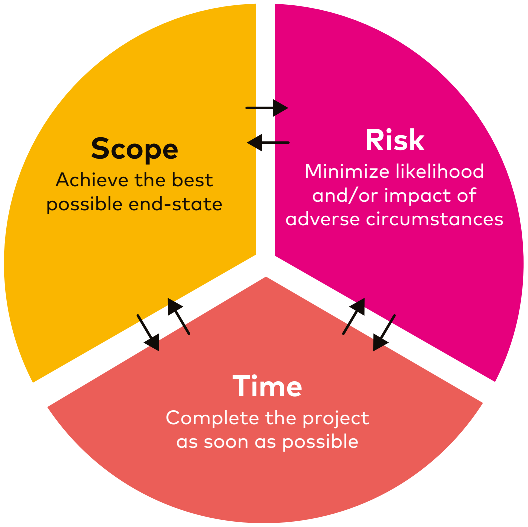
Mainframe modernization is a tremendously complex undertaking. Every project scenario is unique, requiring a carefully considered approach to assessing current system attributes, designing the desired end state, and managing the migration of applications and data. These are high-stakes endeavors involving mission-critical systems. Risk must be meticulously controlled and downtime kept to an absolute minimum.

In most migration projects, there are trade-offs between risk and speed of execution; or between scope and risk; or among some other combination of factors that may limit implementation options, delay project completion, or increase the potential for serious problems or even project failure.

Traditionally, project managers have chosen to mitigate risk by limiting scope (that is, by adopting a "least change" philosophy), or by taking an incremental approach in which major subsystems are migrated one at a time. In either case, tradeoffs appear inevitable. Something has to give. A third option has emerged that defies this paradigm altogether.

It is the industrialization of mainframe-to-cloud migration; a process in which planning, design, testing, and execution are integrated into a fully-automated "factory" approach that allows for unlimited iterations of testing using actual production data, at virtually zero marginal cost.

Organizations taking this approach can achieve highly predictable results through exhaustive testing, with the ability to initiate design changes at any point in the process. Project scope and risk no longer operate in opposition to one another. Moreover, the industrialized approach eliminates the need to break these projects down into increments, allowing for a "big bang" migration, but without the usual level of risk associated with that approach.


Scope
Achieve the best possible end-state
Risk
Minimize likelihood and/or impact of adverse circumstances
Time
Complete the project as soon as possible

As with any mainframe modernization project that Astadia performs, the industrialized approach is built to deliver on our promise of functional equivalence, performance equivalence, and maintenance equivalence. In other words, migrated systems will meet or surpass the functionality, performance, and maintenance effort associated with our client's legacy system.

In this white paper, we'll take a deeper dive into the concept of industrialized migration to offer a more complete picture of how it works and the tremendous value it provides.

2. Industrialization: Three Levels of Automation

There is a consensus that automation is key to mainframe modernization regardless of the transformation approach selected. However, there are three main areas where automation can be implemented during a transformation project.

I. Automated transformation

The first level of automation is transformation. Transformation of data and code needs to be automated, no doubt about it. There is less agreement on the degree of automation, especially of the code. For instance, one school of thought promotes manual and semi-automated interventions when coming off the mainframe and moving to the cloud, as a result of this leaving the principle of iso-functional transformations. The idea of iso-functional transformations, aka like-for-like, on the other hand promotes 100% automated code transformation, postponing other changes until the application has landed in the cloud, and the latest skills and tools are available to undertake this endeavor. Additional benefits are that one does not mix two different objectives and increases risk, and that testing can be automated.

At Astadia, we believe that considerably sized projects deserve and benefit from 100% automation during transformations, leading to shorter project times and much higher success rates.

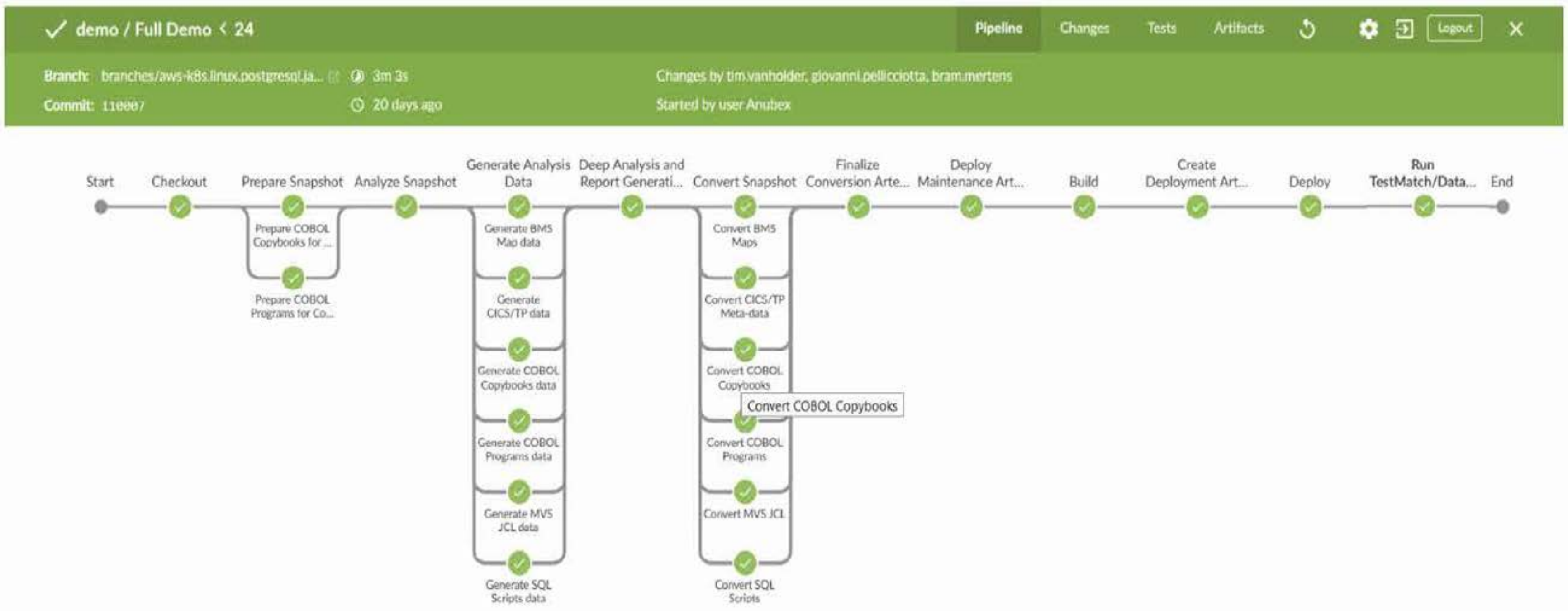
II. Automated testing

The second level of automation is testing. If the transformation is 100% like-for-like or iso-functional, there is this fantastic benefit of being able also to automate the testing. One can compare the functionality of the mainframe environment with the functionality of the cloud environment by software in order to establish the degree of correctness of the transformations. It is hard to overestimate the benefits of this capability and its positive effect on code quality as well as project duration.

III. Automated workflow

The third level of automation is the workflow. We "factorize" the project by automating not only the individual steps in the process but the process itself. The entire refactoring or replatforming project is now automated and can be operated by virtually anyone in a repeatable, consistent, predictable, and risk-free way.

The below screenshot shows a Jenkins Pipeline that performs all needed steps to analyze and convert a COBOL application to Java, build and deploy the Java, and execute the automated tests on the converted application.



✓ demo / Full Demo < 24

Branch: branches/aws-k8s.linux.postgresql.ja... 3m 3s

Commit: 110007 20 days ago

Changes by tim.vanholder, giovanni.pellicciotta, bram.mertens

Started by user Anubex

3. Introducing the Astadia FastTrack Factory

To offer the highest level of automation on the market, Astadia introduced the FastTrack Factory: a software platform combining tools, processes, and technologies to accelerate and standardize a successful mainframe migration project. The Factory can automate the entire migration process:

- Transformation and conversion
- Testing of online and batch applications
- Workflow driven based on a proven methodology

Since the "migration know-how" is embedded in this automation, the specific source and target know-how required is limited to market available IT skills. No extensive mainframe experience is required to be a great contributor in the Factory. People can be trained to an operational level in a matter of weeks.

As a result, the Astadia FastTrack Factory is highly scalable. It can be established anywhere, at the location of choice for the partner or client. Its objective is to achieve a successful and timely migration project while prioritizing a focus on risk and cost minimization. To achieve this, the Factory builds upon clear processes, automated migration, and testing tools, as well as a unique methodology.

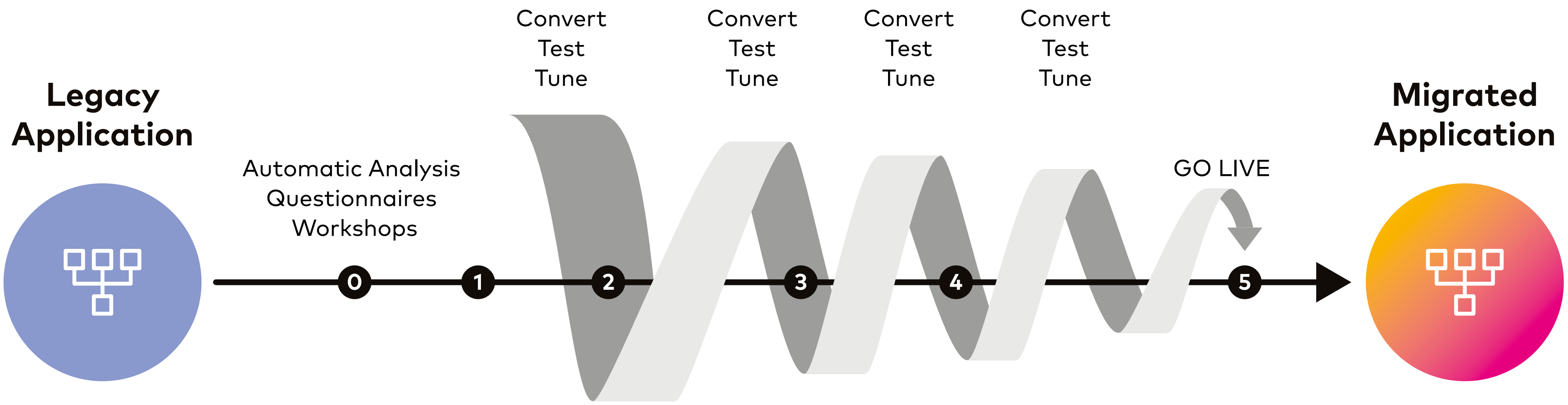
99.99% Automation

The FastTrack Factory provides 100% automated code transformation, compared with the market average of 70-75% automation / 25-30% manual.

4. Game-Changing Technology

In terms of risk mitigation and speed of execution, the FastTrack Factory is a game-changer. With this approach, we spend most of our effort working with clients to design and map the migration process. Once that is complete, we can run test after test, as many times as we want, using the latest data and new test scenarios based on actual user activity. By the time we're ready to go live, we know exactly what to expect. We know how much data needs to be moved and how long the process will take. We will have run as many test scenarios as necessary to establish complete confidence in the migrated system.

Whereas many traditional migration projects are executed in stages in order to mitigate risk, the FastTrack Factory enables a clean cutover of both code and data to a new target system because the migrated system has been so thoroughly tested. While some clients will still opt to perform their migrations in stages, – the FastTrack Factory provides a considerably greater range of options, in large part because the risk involved with performing a clean cutover is so dramatically reduced.



0

Pre-Migration Assessment
- Establish scope, strategy, roles & responsibilities
- Establish project plan
- Legacy documentation

1

Factory Certification
(syntactic correctness)
- Target data model defined & data migrated
- Migration Tools parameterization
- Establish coherent codebase
- 100% compatible output code

2

Factory Exit
(semantic correctness)
- Migration Tools customization
- Automated testing (TestMatch, DataMatch)
- Automated refactoring
- Preparation of Integration Topics (Job Scheduler, Printing, Interfaces)

3

System Integration Test
(functional correctness)
- Application like-for-like (iso-functional)
- Performance test and tuning
- Semi-automated refactoring
- 1:1 test current and future platform

4

User Acceptance Test
(overall correctness)
- Operational use cases correct
- Periodical use cases (day end, year end, etc.) correct

5

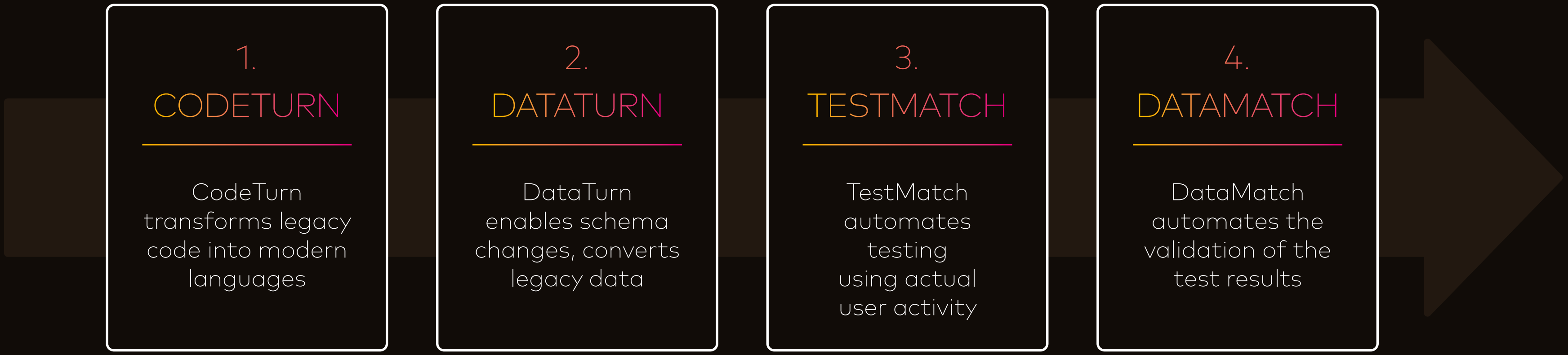
- **Code Freeze**
- **Final Data Migration**

5. An Overview of the FastTrack Factory Concept

In order to achieve the desired outcome, the FastTrack Factory must perform four distinct functions, all of them operating in harmony with one another. The platform achieves this using four primary building blocks, which we call CodeTurn, DataTurn, TestMatch and DataMatch.

- CodeTurn: Automated Code Transformation
- DataTurn: Automated Data Conversion
- TestMatch: Automated Code Testing
- DataMatch: Automated Batch Testing

The development of these products started over 20 years ago, and today includes coverage for legacy technologies such as COBOL, Assembler, IDMS and Natural ADABAS that can be migrated to, among others, Java or C# in combination with all the leading SQL-databases. The resulting applications are ready for deployment in the target platform of choice.



1. CODETURN

CodeTurn transforms legacy code into modern languages

2. DATATURN

DataTurn enables schema changes, converts legacy data

3. TESTMATCH

TestMatch automates testing using actual user activity

4. DATAMATCH

DataMatch automates the validation of the test results

These four tools work in close harmony with one another. A change to the database schema, for example, automatically rolls through to the code transformation process and test automation, without requiring any additional human intervention. CodeTurn accommodates changes made using DataTurn, eliminating the need to manually rewrite database queries within the target system's code. DataMatch, likewise, is aware of database schema changes and automatically accounts for those changes as it performs a database comparison between the source and target systems.

Because the tools in the FastTrack Factory are fully integrated into a single workstream, and because they are so highly automated, the processes of making changes, re-running data transfers, recording new test scenarios, replaying those tests in the target system, and validating the results can be performed quickly and efficiently.

This is what makes the FastTrack Factory such a game-changing innovation. These four components, working together, ensure that each change implemented as part of the migration process is automatically and accurately implemented across every other stage of the project. Highly automated regression testing can then be performed as many times as is necessary to build complete confidence that the new system will behave in a functionally equivalent manner relative to the legacy system.

The “big bang” approach, previously considered to be associated with elevated project risk, now becomes the low-risk alternative. Project leaders can perform exhaustive testing and move to the migrated system knowing that it has achieved the functional and performance equivalence to match the legacy system. Scope, risk, and project timelines no longer act in opposition to one another. Instead, project leaders can get the best of all worlds, – design flexibility, low risk, and rapid completion of legacy modernization initiatives.

6. Automated Migration and Testing Tools

I. CodeTurn: Automated Code Transformation

CodeTurn transforms legacy source code written in COBOL, Natural, ADS, JCL, or Assembler and renders functionally identical code using modern programming languages such as Java, C#, Perl, PowerShell, or bash. It also transforms any embedded statements in the COBOL programs, such as those accessing an IDMS, Db2, or ADABAS database, or CICS. If other languages, such as PL/I or EGL, are in scope of the application migration, Astadia works with established migration partners to offer a complete solution.

Newly transformed code will run as a native first-citizen application on the target environment. CodeTurn does not rely on emulators or sandboxed applications.

CodeTurn keeps the transformed code concise, readable, and maintainable, without code bloat, by centralizing sections of code within libraries and software services. These become an integral part of the generated code, delivered in source code format.

The tool is also highly configurable. It consists of several language parsers, analyzers, rule-based convertors and generators that work together to perform complex transformations on existing source code. These convertors and generators can be easily configured to produce optimal code. For example, if your organization has implemented specific standards for coding, CodeTurn can be configured to incorporate those standards into its transformed application code.

CodeTurn will also ensure the automated transformation and functional equivalence of the code that runs under CICS or IDMS (z/OS or z/VSE)

CodeTurn

Transforms legacy code into modern languages

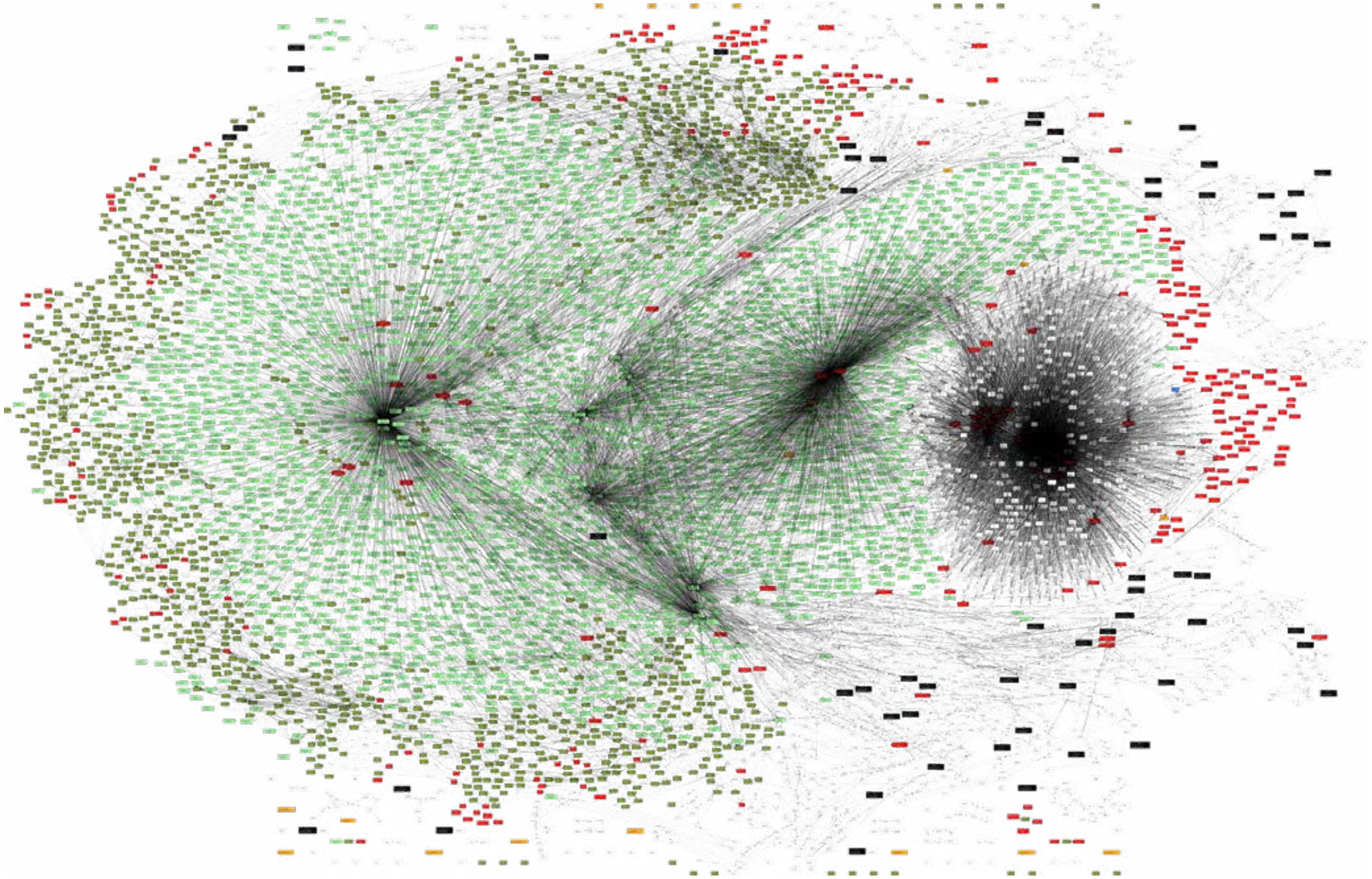
CodeTurn automates the transformation of legacy code into modern programming allowing an automation level of approximately 99.99%. The other 0.01% of the time, when human intervention is required, it is triggered as part of the broader automated workflow process.

as well as OpenUTM (BS2000/OSD) environments. Moreover, it converts the screen definitions to XML and offers a standalone WYSIWYG tool to edit them.

CodeTurn is designed in such a way that it can flexibly be extended to cover additional source and target programming languages. At the same time, third-party conversion tools have already been integrated to cover the conversion of legacy languages such as PL/I and EGL.

During the Factory Certification, which is the first phase in each migration project, CodeTurn performs a deep-dive analysis of the legacy code, identifying and reporting potentially missing programs.

During this process, CodeTurn provides a map of the application scope. A key output of this analysis is what is called the application control flow – a visual report showing how different application components interact with each other.



II. DataTurn: Conversion and Transformation of Legacy Databases

DataTurn serves a critical function in migration projects, not merely because it maps data from the old database to a newer one; it also allows for transformation and normalization of legacy data into a relational structure in the target system. In other words, DataTurn enables clients to modify data structures to better suit their needs following the data conversion.

Most legacy databases, for example, store data in ways that defy the fundamental principles of relational databases. Redefines, for example, are common in many legacy systems, – in effect storing more than one value in a single field and parsing those values at runtime. Occurrences, repeating groups, and other non-relational concepts have been implemented in virtually all legacy databases.

DataTurn enables users to re-map these kinds of anomalies to the target database, where they can be red signed and stored according to relational principles. For example, in a legacy application that allows each customer record to be associated with 10 different phone numbers, the original schema might have been designed as a flat record, with each customer record containing 10 separate phone number fields (Phone_01, Phone_02, etc.). In a relational database, it would generally be preferable to store customers' phone number information in a separate table that links on the customer ID. In this way, database designers can transition from a flat (and less flexible) data structure to a relational one-to-many model.

This is where the FastTrack Factory is such a powerful tool: when data structures differ between source and target databases, application code would normally need to be modified to accommodate the differences.

DataTurn works in conjunction with CodeTurn to ensure that refactored code is generated in such a way that it can continue to access the right data, even when the underlying data structures have changed.

Using the phone number example cited above, a direct mapping of the legacy data structure might result in a query that looks something like this:

```
SELECT CustName, Phone_02 from Customers
WHERE Customer_ID = "ABC123"
```

In a relational context, this would look a bit different:

```
SELECT Customers.CustName, CustPhone.
PhoneNumber from  Customers,  CustPhone
WHERE Customers.Customer_ID = "ABC123" and
CustPhone.Phone_ID=2
INNER JOIN Customers ON Customers.Cus-tomer_ID
= CustPhone.Customer_ID
```

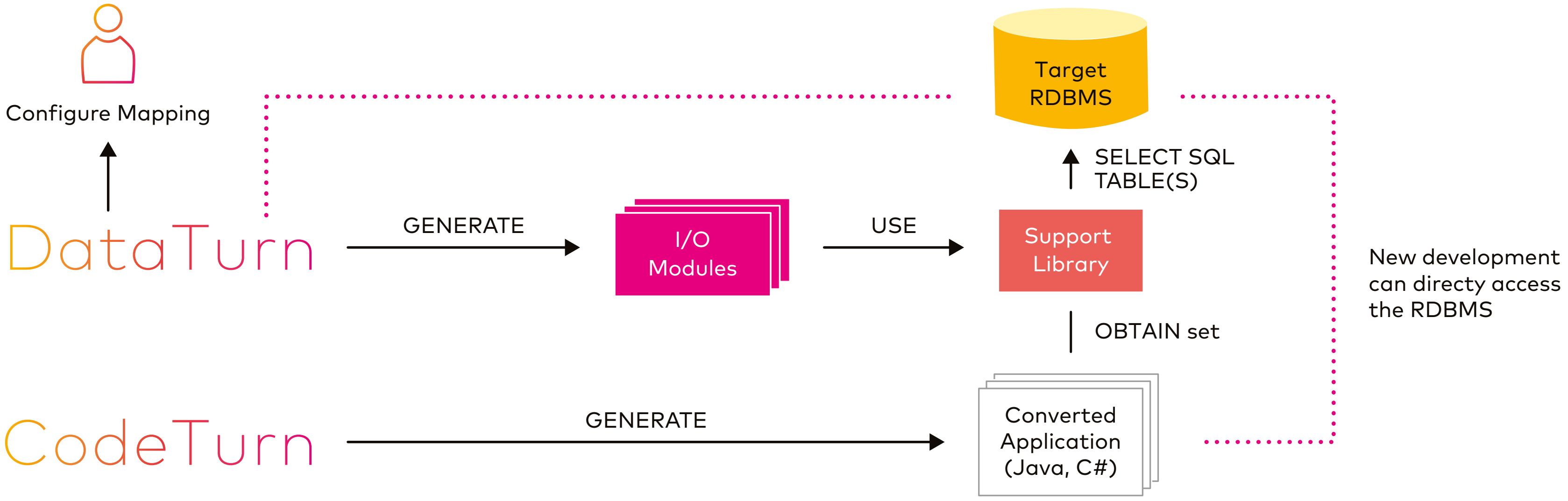
DataTurn

Enables schema changes, converts legacy data

DataTurn automates the conversion of data from legacy databases such as IDMS and ADABAS to virtually any modern relational database, – including SQL Server, Oracle, IBM DB2, PostgreSQL, and others. It does more than simply map existing data schemas to a relational data structure, though. The toolset allows clients to modify data structures to better suit their needs following the data migration.

DataTurn automatically accommodates this change by creating an abstraction layer for data access, which we call the I/O Module. In other words, as the code is refactored, database queries are moved into the abstraction layer, where mapping between the old database structure and the new one is handled automatically. The refactored code does not make direct database calls; rather it speaks to the I/O Module at runtime, which serves as a kind of on-the-fly translator. This allows for full automation and complete control of the process.

Data Access at Runtime



SOURCE	MAIN FUNCTIONALITIES
File (ISAM, VSAM, SAM, ...)	Record/Fields, Group/REDEFINES Fields, OCCURS Fields, Primary Keys, Alternate Keys
Network DB (IDMS, UDS, ...)	Schemas, Subschemas, Areas, Records/Fields, Group/REDEFINES Fields, OCCURS Fields, DBKEYs, Calc Keys, Sort Keys, All Set Types (including System and Multimember sets and preserving set order)
Adabas	PE fields, MU fields, (super/hyper) descriptors...
IMS	Databases, Segments, Fields, (Concatenated) Keys, Parent/Child Relations

Database Normalization Example using MU/PE from Adabas

Customer Number	First Name	Last Name	PE BOC	MU BOC	Street	City	State	ZIP
19811	Laura	Cagnetti	3	1	118 Glade	Erie	PA	16509
				2	271 Larue	Cincinnati	OH	45211
					P.O. Box 58			
				2	733 Hall	Easton	PA	19014
					P.O. Box 7			

Elementary Field

Multiple Value Field

Group Field (Name)

Periodic Group (Address)

DataTurn supports a wide range of legacy databases and file-based data stores, including ADABAS, IMS, UDS, IDMS, VSAM, Db2 and more. It accommodates legacy methods of handling data storage and retrieval, including REDEFINES and OCCURS, schemas and subschemas, PE and MU fields, and set types; translating these into structures and methods that work well within a modern relational database environment.

Table Customers

ISN	Customer Number	First Name	Last Name
123	19811	Laura	Cagnetti

Table Address

ISN	PE BOC	City	State	ZIP
123	1	Erie	PA	16509
123	2	Cincinnati	OH	45211
123	3	Easton	PA	19014

Table Street

ISN	PE BOC	MU BOC	Street
123	1	1	118 Glade
123	2	1	271 Larue
123	2	2	P.O. Box 58
123	1	1	733 Hall
123	1	2	P.O. Box 7

III. TestMatch: Automated Testing based on Actual User Activity

TestMatch provides automatic testing of mainframe OLTP (terminal based Online Transaction Processing applications). The process begins with recording test scenarios from actual user activity using standard mainframe tools. (No client-side or server-side installation of Astadia tools are required.)

Since the recording is done at the network level, it works regardless of the underlying technology used (such as IDMS, BMS, DMS, or Natural). TestMatch supports a broad range of protocols, including 3270 (IBM), 9750 (Siemens), Entire/X, MQ, and TCP/IP.

TestMatch can then replay these recorded scenarios, either against the migrated application or against the original application. Standard replay mode can allow for user think time, simulating normal system behavior with respect to performance. A different replay mode designed for stress-testing can launch many scenarios in parallel to test the limits of system performance under very heavy loads.

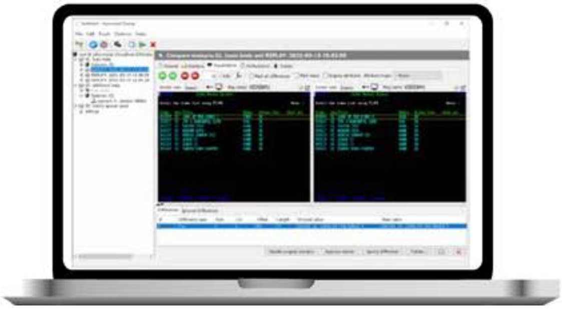
TestMatch provides visualizations of test sessions and scenarios. This includes the terminal contents as well as all attributes that are typical for terminal-based applications: protected/editable, MDT bit, reverse-video, colors. The product includes an integrated diff tool that can display a side-by-side comparison of each recorded scenario and its replayed counterpart.

TestMatch offers detailed analysis of response times and throughput, and also includes helpful utilities to detect performance problems. It calculates elaborate test scenario coverage statistics and the pass rate of replayed tests.

TestMatch

Automated Testing using actual user activity

TestMatch uses tracing capabilities within source IBM or Unisys platforms to record user activity in the live production system. In this way, the Factory's test automation processes have access to real-world test cases at any time, based on actual usage of the legacy system. TestMatch "replays" that user activity in the target system as an automated test scenario. Next, it will compare the activity traced on mainframe with that of what's being replayed on the target platform and flag any errors.



IV. DataMatch: Purpose-Designed to Validate Migrated Legacy Data

The final component of Astadia's FastTrack Factory is DataMatch, a powerful, scriptable database comparison tool that enables quasi-automatic validation of batch job results and reduces testing effort dramatically. DataMatch is designed to address the unique challenges of comparing the contents of exotic mainframe databases such as IDMS and Adabas with a converted relational database residing on the target platform.

DataMatch has two main components, a graphical user interface where database connections are set up, comparisons are configured and results are analyzed; and a scriptable engine that can execute previously configured comparison runs, and which can be easily integrated in other testing frameworks, logging and monitoring tools.

This way, even long-running batch jobs can be tested without requiring manual intervention.

Although DataMatch is designed to be an integral part of the Astadia FastTrack Factory, there are other use cases that call for using it standa-lone. For example, DataMatch can help with regression testing in day-to-day maintenance, ensuring that programs raise the same output both before and after the release. Similarly, it can be helpful when performing operating or database upgrades, validating that pro-grams still work correctly.

DataMatch can also be used to visually inspect the results of bulk data operations such as the mass cleanup of historical data from database tables.

DataMatch

Automates the validation of test results

DataMatch performs a comparison of the source and target databases, – adjusting for any modifications that might have been made to the target data schema as part of the migration process, – and reports any discrepancies, providing fully automated support for testing batch code this way.

7. A Holistic Approach to Automated Migration

Now that we have addressed the four key components of an industrialized approach to mainframe migration, let's look at how these four parts work in harmony to achieve fast, flexible, low-risk results. Although the following steps are presented in list format, they are in reality parts of an iterative process that can be repeated as many times as necessary to establish complete confidence in the resulting migration. Automation makes it possible to perform additional iterations quickly, and with very little marginal effort. That, in turn, means virtually unlimited testing cycles and a high confidence level when it's time to put the migrated system into production mode.

DataTurn Database Configuration & Design	Database configuration is read from the legacy source and used to create a database configuration in the target RDBMS. Design changes can then be made to the target database using DataTurn's configuration tools. The other three components of the FastTrack Factory are "aware" of design changes and accommodate them automatically. DataTurn generates an I/O module that acts as an intermediary between the transformed code and data access. This eliminates the need to rewrite queries within the source code to accommodate new data structures in the converted database.
CodeTurn Code Transformation	CodeTurn consumes legacy code and transforms it into a modern language such as Java or C#. At runtime, the migrated code will access the target database via the aforementioned I/O module, ensuring functional equivalency regardless of any changes made to data structures.
TestMatch Generate & Run Test Scenarios	Test scenarios are recorded on the source system using actual user activity. TestMatch runs those scenarios on the target system, populating the target RDBMS with new and updated data. TestMatch also records performance metrics and allows for execution of normal or stress-testing scenarios.
DataMatch Database Comparison	DataMatch copies the legacy data into a standalone database that conforms to the target system's specifications. This serves as a comparison copy which should match the test data produced by the target system itself after test scenarios have been executed.
Additional Changes & Iterative Testing	Additional changes will be made as needed, and testing repeated as many times as necessary to establish complete confidence that the migrated system will meet or exceed expectations with regard to functional equivalence and performance. When stakeholders are satisfied that the target system is performing as expected, a final migration of data to the target system can occur and a clean cutover can be made.

By industrializing this process, Astadia's FastTrack Factory helps our clients accomplish their goals much faster than they otherwise might, reducing the duration of a typical migration project by as much as 90%. Automation also reduces risk because it enables repeated testing based upon real-world scenarios derived from actual user input to your production system, recorded and replayed in a refactored test environment to ensure functional equivalence to existing legacy systems.

For more information about the Astadia® FastTrack Factory™, contact cloud@amdocs.com

Amdocs helps those who build the future to make it amazing. With our market-leading portfolio of software products and services, we unlock our customers' innovative potential, empowering them to provide next-generation communication and media experiences for both the individual end user and large enterprise customers. Our approximately 30,000 employees around the globe are here to accelerate service providers' migration to the cloud, enable them to differentiate in the 5G era, and digitalize and automate their operations.

Listed on the NASDAQ Global Select Market, Amdocs had revenue of $4.89 billion in fiscal 2023.

For more information, visit Amdocs at www.amdocs.com

For nearly three decades, the experts at **Astadia** have performed mainframe modernization projects for government agencies and companies throughout the world.

- 300+ total mainframe projects
- Billions of lines of COBOL converted
- Unparalleled access to mainframe modernization subject matter experts, architects, developers, engineers and project managers
- Industry-leading migration success rates that few can come close to matching

For more information, visit Astadia at www.astadia.com

amdocs | make it amazing

www.amdocs.com