Planning for Modernization
System Assessments

By

Roger Hammer
Product & Services Director
June 6, 2013

System Assessment

Introduction

blackboxIT's approach to legacy system modernization is unique in our ability to leverage the investment made in the legacy system through years of development and capturing of the organizations business processes in the source code of the application.

Our automated analysis tool evolveIT can sort through the decades of extensions, architecture and coding philosophies, technology workarounds, etc. to "mine out" the code artifacts of value to build a road map for the modernization effort. We break apart the system to clearly define a highly manageable project through a " Process Chunking" approach and document the Requirements for the modernization project in 1/10th the time of traditional approaches. By leveraging the existing legacy system our approach maximizes accuracy enabling our customers to ensure they can replicate key functionality. Our approach provides:

System Assessments and planning modernization

The most common approach to system assessments for modernization planning is to document a high-level definition for the existing business processes of the application as a framework for decision making to determine the approach, cost, risk and value of the modernization project. Analysts typically rely on interviews with Systems Experts to document the existing business processes. They may look at screens, existing system manuals or other existing documentation. They may even gather some system metrics, number of lines of code, screens, jobs, if's, GoTo's, etc. However, due to time and cost inhibitors, they leave the analysis of the more-detailed information about applications, their components and their interrelationships and governing business logic to the execution phase of the project.

This approach can often be a recipe for disaster as it fails to account for difficulty of replicating decades of investment and process refinement captured in legacy system. A serious risk and cost impediment to application modernization projects.

blackboxIT augments the traditional top-down approach of interviewing System Experts with automated code analysis to understanding the details of Cobol systems. We provide our customer with many more "facts" about the system to make an informed modernization decision. Our top-down bottom-up approach provides a more complete solution at lower cost and lower risk.

blackboxIT Approach

blackboxIT will use evolveIT, a commercially available product of blackboxIT, to conduct analysis and produce the documentation with greater efficiency than with traditional mainframe tools and analysis processes. The blackboxIT tool set can be a tremendous asset in assisting discovery and documentation of existing processes, business logic, batch processes, data and control flow of mainframe systems. While ultimately the code is the only 100% accurate source for how the system processes information, interpreting the code within large legacy COBOL systems is difficult particularly when the analysis is not produced within the context of specific business outcomes.

Key Steps to Streamlining Assessment Process

1. Break system up into manageable pieces for analysis - By IT process from a Batch and On-line perspective
2. Tie each IT Process to Business Process narratives to provide context for analysis
   a. Each IT process will have a one to one or one to many Business Process relationships
3. Map External Interfaces to each IT process
4. Tracing Data Transformations in each IT process will provide the details of the current state implementation
   a. evolveIT repository ready and able to provide data transformation analysis
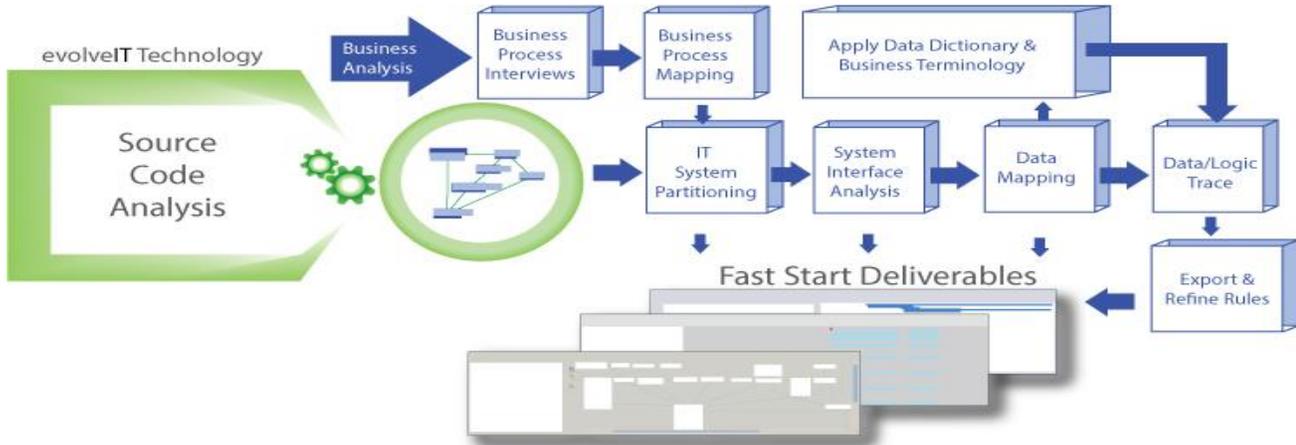
blackboxIT will work with Customer IT group to define/name the IT and business processes at the most basic level from a batch schedule and screen perspective. From this starting point, blackboxIT will use evolveIT to identify the key driver screens and what job schedules start batch sequences of significance then through evolveIT's control and data flow understanding. We identify all the down line jobs, programs, copybooks etc. that are possible for each IT process segmentation. Once completed for each IT process starting point, we will cross reference each to find overlaps, but more importantly we identify any system components, jobs, programs, screens…that aren't tied to any of the researched IT processes. We now have a list to investigate those that aren't accounted for that will need further investigation.

blackboxIT will also meet with the business to provide overlaying business context to each of the IT system process segmentations.   We could also drill down and do the specific traces of data for selected process outcomes. This should provide Customer's the necessary understanding to determine the effort to complete further traces. Since some data is static and is never transformed we can focus traces on data outcomes that are transformed from input to output. In later stages this can help to provide boundaries for which IT processes need to be broken apart and aligned to the business process mapping.

Sample High Level schedule

Phase	Steps
• **Refine Scope and Assessment Charter**	• **Complete System Component accounting and build evolveIT database** • **Onsite Customer Interviews to identify system processes and inputs** • **Eliminated Dead and Obsolete Code/processes** • **Complete system partition heat map**
• **Code and System Analysis**	• Map the Business Process to the Named Interfaces and IT Processes. • **Map Input Interface Data to Output Interface Data** • Identify important Data Transformations • **Trace relevant data to capture Transformations and related Decisions**
• **Refine and Deliver**	• **Apply business terminology to Code**

Sample Results

System Analysis to limit scope of future projects
- Analyze Job Schedules and dead code to eliminate from scope of work.
    a. List of obsolete components related to Jobs that are obsolete.
    b. List of components related to Obsolete Transactions.
   c. Identify Missing components as defined by evolveIT.

System Analysis
- Onsite Interviews with IT and Business analyst / Users to:
    a) Collect all A/R System source code.
    b) Collect Batch Job "Last Run Date" for all Batch Jobs.
    c) CICS "Last Executed" report(s).
    d) Understand all Naming conventions for Batch Streams and CICS Transactions.
    e) Collect CICS report defining the Online system structure.
    f) Identify the major Business processes related to the A/R System.
- **Job stream analysis**
    a) Identify dependencies between all schedule streams.
    b) Identify all dependencies within all Batch streams.
    c) Build Schedule /Job stream top down call trees.
    d) Identify all files / tables associated with Batch stream.
- **Screen / Transaction analysis**
    a) Identify all Screens and how they relate to Business process.
    b) Build screen to driver program with call trees.
    c) Identify all files / Tables that are associated with the CICS online applications
- **Component Use analysis**
    a) Identify all components that are used in multiple system processes that have been identified so far (Schedule / Job stream and Screen/Transaction analysis).
    b) Identify all components that are not used in any of the system processes identified so far.
    c) Review all currently unused components to determine if they are:
        i. Dead components.
        ii. New processes that need documented.
        iii. A part of a currently defined process with some kind of indirect relationship.
- **Identify all of the inputs and outputs of the system and develop an Interface map**.
    a) EDI, file inputs, file outputs, screens, MQ Series calls, etc.
    b) Capture any related names that can help identify the interface, (Screen name, etc.)
    c) Build a data map for each of the interfaces.
        i. Identify all data that is a part of the interface.
        ii. Include copybook name with all data formatting.
- **Use each of the interfaces to the system to understand how the business process connects to the system.**
    a) Work with the Business Analyst and Users to develop a narrative of how the business uses the interface.
- **Determine which input data is not transformed.**
    a) Identify the input data that is not changed by the program other than input edits.
    b) Data remaining in the interfaces should be the data that needs traced to determine how it is created.
- **Map all input interface data to output data.**
- **Backwards Trace sample data that is not well understood by the business.**
    a) Show customer how trace will provide understanding when further analysis is needed.

System Data Index

	A	B	C	D	E	F	G	H	I
1	Job	Job Step	Program	Datastore Type	Datastore Name	Data Type	FD Record Name	Data Name	Format
2	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Data Rec	AR31-HEADER-RECORD	01 AR31-HEADER-RECORD	
3	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-BTCH-DT	PICX(10)
4	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-BTCH-ID	PIC9(10)
5	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-CV-NBR	PIC9(07)
6	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-MOD-PGM	PICX(08)
7	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-MOD-TS	PICX(26)
8	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-OPER-CO-ABBR	PICX(03)
9	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-ORGL-PGM	PICX(08)
10	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-ORGL-TS	PICX(26)
11	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-PMT-CTL-AMT	PICS9(11)V99
12	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-PMT-CTL-ID	PICX(30)
13	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR31HDR	FD Field	AR31-HEADER-RECORD	05 AR31-STAT-IND	PICX(01)
14	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Data Rec	AR32-DETAIL-RECORD	01 AR32-DETAIL-RECORD	
15	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-BL-INV-ID	PICX(12)
16	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-BTCH-DT	PICX(10)
17	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-BTCH-ID	PIC9(10)
18	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-CUST-PMT-AMT	PICS9(11)V99
19	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-ITEM-ID	PICX(30)
20	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-MOD-EMPE-NBR	PIC9(09)
21	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-MOD-PGM	PICX(08)
22	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-MOD-TS	PICX(26)
23	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-OPER-CO-ABBR	PICX(03)
24	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-ORGL-PGM	PICX(08)
25	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-ORGL-TS	PICX(26)
26	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-PMT-CTL-ID	PICX(30)
27	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	&&AR32DTL	FD Field	AR32-DETAIL-RECORD	05 AR32-STAT-IND	PICX(01)
28	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	ACC.P.CMC.EDI.DRA.AR820(+0)	FD Data Rec	EDI-VTIMS-RECORD	01 EDI-VTIMS-RECORD	
29	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	ACC.P.CMC.EDI.DRA.AR820(+0)	FD Field	EDI-VTIMS-RECORD	05 EVR-CHECK-NBR	PICX(30)
30	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	ACC.P.CMC.EDI.DRA.AR820(+0)	FD Field	EDI-VTIMS-RECORD	05 FILLER	PICX(10)
31	CMCAC11P	JOB.PROC0010.STEP0040	ACB010	File	ACC.P.CMC.EDI.DRA.AR820(+0)	FD Field	EDI-VTIMS-RECORD	05 FILLER	PICX(139)

System Partitioning Map (Heat Map- identifies system data coupling and enables system partitioning)

System Partition Documentation
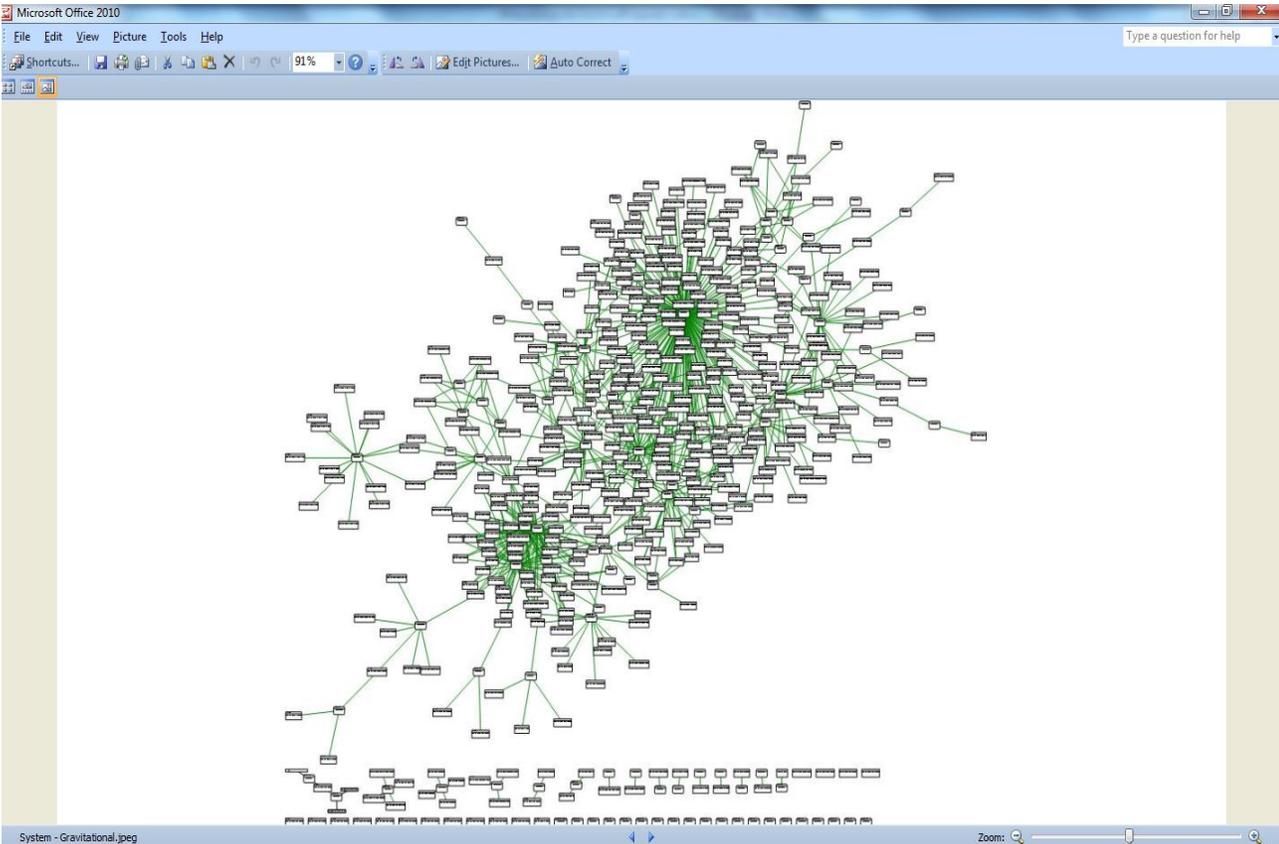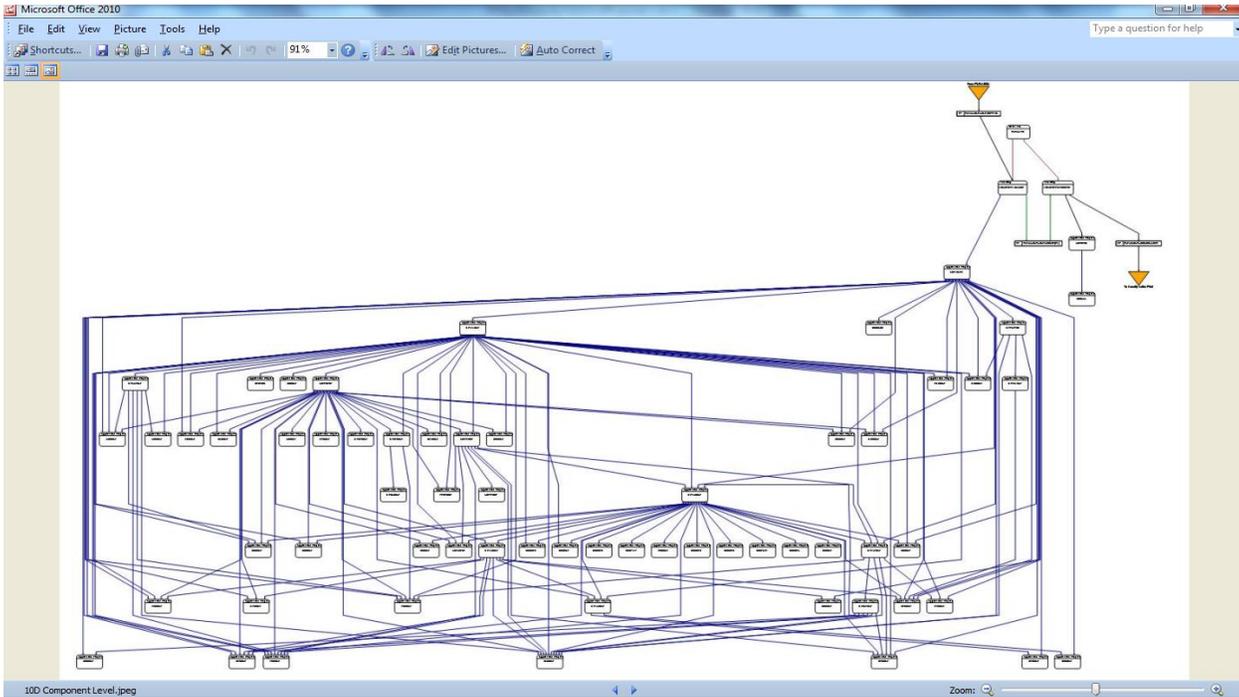
Partition specific Batch data and control flow