

About Client

Vestcom is one of the largest print and mail industry in North America. Founded in 1969, Vestcom provides outsourced digital printing services for some of the largest companies in the insurance, financial services, travel, leisure and gaming industries.

Vestcom's work is seen by millions of people daily in the form of account statements, insurance policy enrollment kits, direct mail and personalized marketing materials. Vestcom's Marketing and Business Communications (MBC) division prints over one billion pages per year and has multiple production facilities strategically located across the US and Canada.

The Requirement

Vestcom's IT infrastructure is a critical component of Vestcom's performance. The IT processes guide the business operations by organizing, formatting and printing large batches of paper on a daily basis. Legacy is core to these critical operations. In 2003, Vestcom took a hard look at their expenses and came to the conclusion that dramatic action was necessary to reduce mainframe expenses. The company outsourced the hosting of their mainframe operations and paid a sizeable monthly fee that represented 25 percent of the total IT budget for CPU and networking services. The company looked at a handful of vendors before deciding on Macrosoft.

After a successful pilot in late 2003 the decision to migrate major operations from the mainframe came in early 2004. Vestcom launched a \$1 million initiative to move over 220 independent and complex applications from the mainframe to four 2.5GHz machines running Windows®Server 2003. Overall statistics of the project is given below:

- **Total applications = 220+**
(for more than 36 Vestcom Clients).
- **COBOL's = 500+**
(averaging 2000 lines per program)
- **JCLs & Procs = 1600+**
- **Quikjob components = 175+**
- **Syncsort components = 520+**
- **Fileaid components = 560+**
- **Print utility components = 1-2 per application**
(Anchor, Transformer, Dialogue, EZ letter etc.)

Solution

The project started in April 2004. Over 20 developers were involved in the project. The project was successfully completed in March 2005, and all the applications were live in May 2005. Thus in a period of 1 year we could successfully shift 220+ applications from mainframe to PC (migration rate of approximately 1 application per day). Our migration toolkit helped in achieving this

Application Architecture

The programs are designed for batch execution (background processing) in order to handle large volume data. Typically every application's data files are more than 3 GB in size. We have also envisaged future web enabling of the applications.

This is a true one-to-one conversion retaining all features in the mainframe application. Each application is a collection of many programs (called steps), scheduled, monitored and processed by the main application processor. This scheme facilitates restart facility at an intermediate stage in case of failure (retaining the RESTART facility of mainframe). Steps can also be bypassed. Each execution state is archived for audit (ie; not only data, but also the environment used to create that data is automatically archived at the end of execution). Each application can be run in "Test" or "Production" mode at any time. The various executable in an application are developed in different languages (Cobol, Perl etc.) as well as may call third party tools. All these modules communicate to each other via shared memory/disk in a standard messaging format. This provides unlimited expansion possibilities.

Operator interaction is required only with a configuration file. Operator gets detailed reports including "Reconciliation Reports" (summary of processing), and "Floor Reports" (Action points for the operator). Most of the programs creates their own timestamped log reports. Auto emails are generated on error (attaching the error reports), and also on successful execution (status feedback). These are send to all the designated addresses during the execution.

The application server and data server are separate. A central data server acts as the central repository of data and the applications residing in application server collects raw data, and deposits processed data, as and when required. This facilitates movement of applications among Test, Preprod and Production servers smooth. The applications are self-tuning, and installation is not dependent on machine, disk or directory.

Methodology & Processes

This project was executed using our "global delivery model" which involves collaborative work among three geographically distributed teams ie; "US Team", "Indian Team" and the "Client" (Near shore, Off shore, On site). 3 dedicated PC servers are used for this purpose, and all the three teams work on these servers remotely through high speed VPN connectivity. The servers are kept at the client site. There are standard processes for downloading the blueprints (mainframe programs with analysis docs & input data) to the designated directories, uploading the converted modules for unit tests, integration and system test etc. The final integrated software gets moved to preprod (pre-production) server for trial (parallel) run. On user acceptance, it finally goes to the production server for the actual run.

The Macrosoft Migration Tool Kit (MMK)

The main components of our migration toolkit which made this project successful, are briefed below:

JCL ANALYZER :

It takes the JCLs and associated procs kept in the source directory and generates analysis documents. It separates steps, comments etc., resolves symbolic variables and associates related procedures.

These outputs are user-friendly HTML pages with expand tabs (+). You can click on these expand tabs to zoom into details. It also creates the necessary data, which can be directly input to the "JCL CONVERTER" utility.

JCL Converter:

This is the JCL code converter. It reads in the data file generated by JCL ANALYZER and automatically generates the code for the target platform. This saves much of the development time, and also helps in standardizing the code (uniform look and feel). The developers can thus concentrate on specific conversion/development works in each step.

COBOL Converter:

This module makes standard conversions for COBOL source code.

QUICKCODE :

This is a source code generator to replace IBM 3rd party utilities such as Quikjob, Fileaid etc.

SHELLGEN:

This is a generic utility for creating the handlers for IBM 3rd party utilities such as Syncsort, Transformer, Anchor etc. The program reads in an I/O specification file and generates the code

GDG HANDLERS:

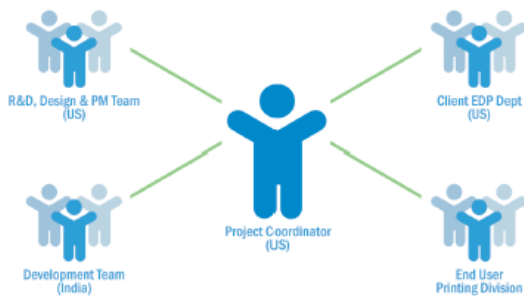
The GDG (Generation Data Group) handlers simulate GDG type datasets on PC.

DATALEANER:

This is a utility for analyzing and cleaning up data files in different formats (and EBCDIC to ASCII conversion etc.).

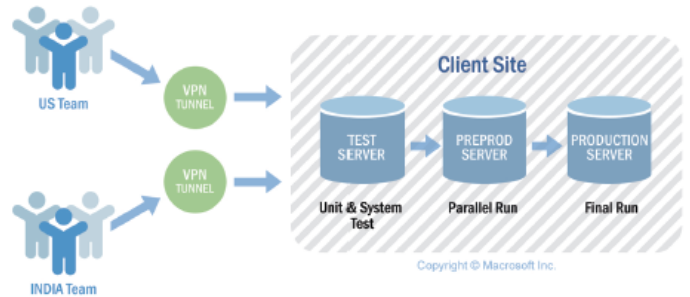
JOB SCHEDULER:

This gives a visual interface to the operator by which scheduling, configuration changes, execution history viewing etc. are very easy and user friendly.



Key features of the project implementation includes the effective "MVS Change Management" process, Gap Analysis, Tool based approach, and established data/database migration strategies.

Development & Deployment Process



The "Project Coordinator" at US, who controlled all the delivery and communications, did the overall coordination. Strict document control and flow, 24x7 helpdesk, separate customer support team, independent QA team etc. helped in managing this efficiently.

Partners with:

MicroFocus

Members of:

Migration & Transformation Consortium

Mainframe Migration Alliance



www.macrosoftinc.com

