



business solutions

OpenARMS

As part of the Inglenet migration methodology, Inglenet has created a comprehensive process and suite of tools called "**OpenARMS™**". The **OpenARMS** process consists of a number of services that are provided to ensure that the modernization project will deliver the desired results. These services include the following:

- Planning and Assessment
- Project Management
- Database Design and Implementation
- Data Conversion Services
- Re-Hosting Delivery Services
- Testing Services
- Production Launch Management

This unique process is continuously revised to reflect the latest techniques employed in legacy system modernization.

Heritage Support Package for 2200 – HSP/22

OpenARMS' conversion tools and utilities for migrating Series 1100/2200 applications to UNIX and Windows are referred to as Heritage Support Package for 2200, or HSP/22, and are unique to Inglenet. HSP/22 consists of the following:

- COBOL 68/74/85 to Micro Focus COBOL 85
- TIP/1100/2200 and HVTIP to TIP Studio
- IMS/1100/2200 to TIP Studio
- ECL runstreams to runstreams powered by UNIX Bourne shell scripts
- SDP/DPS Screens to TIP Studio screens
- RPG to Micro Focus COBOL 85
- DMS/1100/2200 to ORACLE with TIP Studio database interface
- MSAM to ORACLE with TIP Studio
- 9 bit to 8 bit ASCII data conversion

The OpenARMS process in a Nutshell – Data, Interface then Code

The ultimate goal of the modernization project is to create a new IT environment that employs current technology in the areas of database, user interface and development techniques. The Inglenet approach to accomplishing this goal is to segregate the entire process into discrete phases where each phase has a specific deliverable with measurable results and benefits.

In general, the phases of a legacy modernization project are as follows:

- Phase I – The Data Modernization Phase
- Phase II – The User Interface Modernization Phase
- Phase III – The Code Modernization Phase

Why focus on the data first?

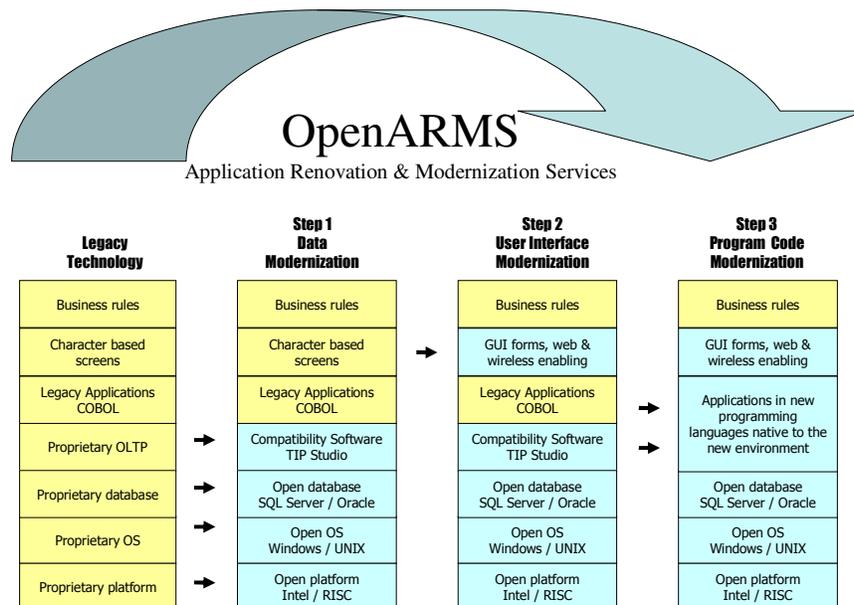
The data is clearly an extremely important and valuable asset of the organization; however, is it truly more important than the code that defines the organization's business rules or the skills that exist in the people who use and maintain the IT systems? It is not the relative importance of the data but its impact on other aspects of the modernization project that determines that it be the focus of the first phase of the project.

If a fundamentally sound data model, that will support the business needs of the organization, is not developed as part of the initial phase then future modernization efforts will be more difficult and less likely to succeed.

To modernize the data, the data must be moved to a newly designed data model hosted by a modern relational database management system on a new platform. This requirement will drive several other activities that all become part of the data modernization phase of the project.

During the first phase, many components of the legacy system are replaced with new components; however the application code remains unaltered by design. By keeping the application code intact, the initial phase of the modernization project is much more manageable and cutover to a new hardware/software platform can be accomplished in a shorter time frame than would otherwise be possible. Ingenet's **TIP Studio** compatibility software components make this possible.

The following diagram illustrates the phase 1 activities of the modernization project.



Throughout this process the operation is protected through extensive testing, parallel operations, and user involvement.

Our initial phase of the modernization project provides the following:



business solutions

- Develop a new data architecture. A new Architecture simply states the rules for how the data will be managed.
- Analyze and understand the current data storage and retrieval systems, be they DMS, RDMS, flat files, or use some other data storage technique. This is really a consulting assignment that we undertake to understand what the data requirements for the organization are.
- Work with the customer to design a new data storage and retrieval system based on current database technology (RDBMS).
- Provide tools and services (OpenARMS) to transform the data from the old model to the new one.
- Provide interface routines that allow the current legacy applications to run unaltered using the new data model on the new servers.
- Move the existing legacy code to the new platform using the interface routines we have created.

Completing these initial steps already provides tremendous benefits. Data is now stored in a way that it can easily be used by modern reporting and query tools. Development can now build new systems using new tools, languages and frameworks to further process the data while having their core legacy applications continue to operate the way they always have.

Next phase is to modernize the User Interface.

Modernizing the user interface can provide significant benefits to those who use the information systems. Apart from the enriched experience and all the well-documented ease of use and accuracy features a modern graphical interface provides, the most significant benefit may be one of user perception. When the user interface is modernized to employ current form-based graphical techniques, users have the feeling they are working with a modern information system. The new user interface is also a great way to visibly demonstrate the progress being made in the overall legacy modernization project.

The user-interface modernization phase can either be performed as a distinctly separate phase or it can be combined with the data modernization phase and accomplished during the switch to the new hardware and software platforms. The determination of when this phase is done is usually made based on factors such as; the overall scale of the project, the logistics involved in introducing a new user interface and how much customization may be desired in a new user interface.

Modernizing the user interface is dependant on the kind of interface currently employed by the application programs. If the applications are sending low-level, terminal specific codes such as UTS control codes or DICE sequences, then significant modernization is not really possible. However if the programs are using externally defined formats such as DPS screens, then **TIP Studio™** provides techniques to allow these character-based screen formats to be replaced with graphical forms

Consideration is also given to situations where the end user is not a person, but rather another computer. Having one computer communicate with another one through a terminal interface has been a popular technique for sharing information in the legacy environment. This technique, often called "screen scraping" is somewhat fragile as even the most cosmetic of changes in the information displayed by one computer may break this interface. Modern systems have long abandoned these types of techniques in favor of proper programmatic interfaces such as web services and XML data structures. However, these older interfaces have to be preserved as the systems are moved to new platforms. Components of IngleNet's **TIP Studio™** software guarantee that other computer systems



business solutions

connecting into the legacy system running on a new platform will interface exactly as they did when the system was run on the proprietary mainframe. This will ensure that operations can continue uninterrupted. When these interfaces are eventually replaced with a modern one, the networking compatibility software will no longer be required.

The final phase is to modernize the Code

Once the legacy applications have been migrated to a new, open platform the process of modernizing the applications can begin. IngleNet utilizes one or more of the following strategies to modernize application code:

- **Develop new applications while the legacy applications do the old work.** Applications built to utilize technologies such as the world-wide-web and wireless networks may serve the company better than focusing limited resources on renovating legacy applications that are performing well.
- **Rewrite applications as requirements dictate significant changes**At this point, due to the work done in data modernization phase, we have already laid down the foundation for rewriting the applications by creating a modern, well structured database that can be used by both legacy and newly created applications. Using this technique, applications are gradually replaced with new ones over time according to the business requirements of the organization.
- **Recode applications manually.** Since this activity only replaces current applications with a new one that essentially has the same functionality, organizations may choose to make this a secondary priority that is done when no critical or “must do” tasks exist.
- **Transform applications using an automated process.** Tools exist that will convert existing legacy applications originally in early generation languages such as COBOL to modern languages such as Visual Basic or C++. The tools reach levels of completeness as high as 80 to 85%.

An IngleNet solution considers your legacy system, and provides a strategy, methodology and the services for moving to open system server based architecture. With a proven strategy for dealing with modernization issues, IngleNet Business Solutions has successfully managed the migration for many Unisys mainframe customers. In all cases, the projects were carefully planned and executed within the established time frame and budget.

The methodology employed by IngleNet has been refined with the experiences gained from each modernization project — whether the project involved a small System 80 system or a large, enterprise-wide 2200 environment, IngleNet can do the job.

4310 Sherwoodtowne Blvd., Suite 400, Mississauga, ON, Canada, L4Z 4C4

Phone 905 848 2030 : Fax 905 848 3108

For more information, email info@inglenet.com or visit our website at www.inglenet.com

