



Table of Contents

Concept.....	1
Problem Addressed.....	1
Traditional Development Methodology:	2
System Development without Programming	2
Leading-edge Technology	2
LeoBiz Double Advantage	2
MenSitpi Technology.....	2
MenSitpi Architecture	3
Schema Server.....	3
Win-form-Client	3
Report/Update	3
N-Tiered Architecture	3

Concept

Problem Addressed

Business Application Development process whether product or project has not changed from its inception. Three issues still haunts them.

1. Programming is still unreliable and new system tends to have lot of bugs
2. Whenever technology Changes applications need to be redeveloped from Scratch
3. Lack of Business Domain Knowledge of the IT professional

End result is Most of the Mission critical applications systems and products in Commercial and FSI Industry are still using archaic Multiuser systems and Client Server Technology. One or two in three tier technology, even they are scratching the surface of technology not using its full capability and not reaping maximum benefits. They still use RDBMS technology for data base or Indexed File structure's. Built on programming language such as procedural COBOL and RPG II. Many of them Use Green Screens or Screen Scratching by emulating green Screen in GUI. They cannot really use new technologies such as server

less cloud computing and NoSQL data type without complete rewrite. Business applications uses technology that are 10-20 years old. Not the current one. Therefore they are costlier and time consuming to maintain and enhance. Multiple systems using various technologies of different type are used in an organization requiring costlier interfacing of systems and numerous technical problems.

Users understandably do not want to try new systems and products using new technology because Software development process is inherently unreliable. They wait until it is used by several adventurist and less affluent users. But by the time they adopt, its technology has become obsolete.

We believe that we have found the **Silver Bullet** for the Business application development process. It delinks Transaction Processing Application development from Technology intricacies and Application System becomes specifying Application requirement which will be interpreted by re-use of only four software components at execution time. This delinking brings three benefits

- Application and Technology can be enhanced independently resulting in elimination of re development from scratch when technology changes so that systems and application stay in current technology and to get its benefits.
- Application development by business domain experts while technology is handled by Technologist means Users get the benefits of a meeting the business needs exactly using always current technology.
- Faster, reliable and less costly Business application development/maintenance

We spent 12 years in the R&D to bring out a commercially viable Development Tool MenSitpi. In the process as a proof of concept, we developed using MenSitpi an ERP product LeoBiz. LeoBiz Development was done with unmatched ease, using few



business graduates with some IT skills as opposed to traditional IT knowledge workers.

Using new set of Skills, we produced unmatched functionalities especially the ability of single LeoBiz product to cater the requirements of both commercial and financial services Industry.

Unlike any other software company who bring out incomplete products to be experimented in the Market place, we tested in a limited customer base for 12 years.

Traditional Development Methodology:

Traditional methods of software development do not guarantee error free software as they are developed. The reliability can only be determined by usage and is the main reason that prospective customers insist on reference sites before buying software.

The underlying causes of reliability are the **programs** themselves as each line of a program is a potential cause of failure. When a software package has about **ten million** lines of code, as in a typical ERP system developed using traditional methodology, the reliability has to be ensured by usage at several reference sites.

System Development without Programming

LeoBiz has been developed without using traditional methods of programming, but by a revolutionary method which has not been used by any other company in the world.

Use of **MenSitpi** development framework designed and constructed in Sri Lanka resulted in only around 2,000 lines of application-specific code in the development of ERM package, **LeoBiz**.

Therefore reliability is inherent in this **software**. It does not require usage to ensure reliability. With a programming ratio of 1:1000 lines of code, the reliability ratio will minimally be 1,000,000:1, assuming that reliability is the square of the lines of code at the least, though in practice it is much more.

Leading-edge Technology

The international ERP packages currently in the market place have been developed at least two decades ago, which is why they have a large number of reference sites. These packages do not use the current technology enhancements that have increased flexibility and cost effectiveness of IT.

The design of these older software products depended on Technology that existed two decades ago and are therefore costly and obsolete. **LeoBiz** costs much less since it uses the latest and most cost-effective technology.

LeoBiz Double Advantage

LeoBiz has the advantage of revolutionary **methodology** and **leading edge technology** complementing each other, creating unparalleled reliability and an extremely cost effective product

MenSitpi Technology

It uses the advanced N-tiered Architecture while almost all other products use the Client-Server architecture.

The N-tiered Architecture protects the investment in Hardware.

- ❖ Benefit from the latest technology from Microsoft; “.Net Framework” and C#. This advanced technology runs on Windows component servers rather than on database servers. Microsoft plans to deliver this on other platforms as well in the near future.
- ❖ It is built using the Open Model of **OLE DB**, so that it could be used with several Hardware and Database Servers. Choices of Server/Operating Systems are **Microsoft Windows, IBM eServer, iSeries, pSeries and IBM zSeries**.



- ❖ The following databases are currently supported: IBM DB2, Oracle, Microsoft SQL Server, or, MSDE.
- ❖ It is built using Sri Lanka expertise both for design and construction.
 - The price / performance ratio is extremely high.

MenSitpi Architecture

It consists of four components that are reused to implement all functionalities of any application almost eliminating any code, For **LeoBiz** apart from these four components only around 2,000 lines of Microsoft Dot Net C#. is used

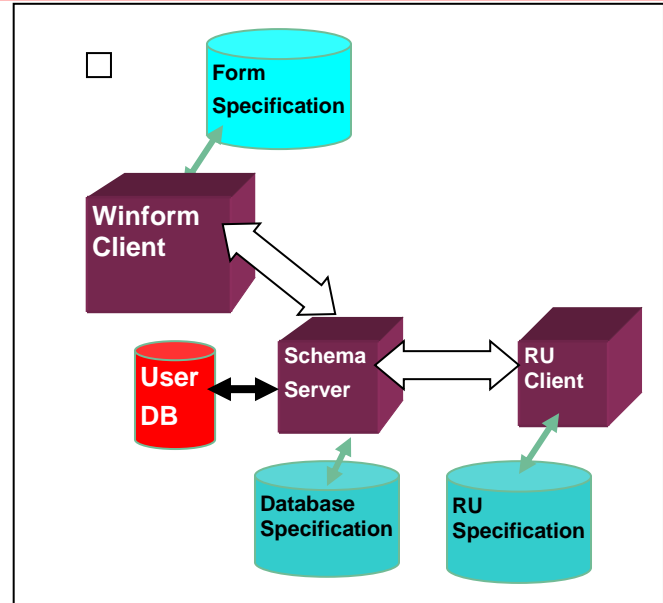
The **MenSitpi** framework has four functional components that are re-used to implement any application:

Schema Server

A middle layer which mainly enforces specification driven, with options for code complementing, business rules. This layer manages the User Data Base, updates and ensures the integrity of the Data by enforcing the business rules.

Schema server highlights are:

- Business rules defined and enforced better than in most RDBMS,
- Enforcement of business rules specified without repeating in multiple instances and as and when the user interacts with the application; which none of the other existing tools provide.
- Provides object interfaces to program interfaces as user plug-ins - these are written in any industry standard language such as C#, Java, COBOL, Visual C++, Visual Basic instead of propriety scripts
- Upgrade of versions, without user intervention



Win-form-Client

Window Client driven by specifications, with option for code complementing, interacts with the users for data capture and inquiry. It navigates through data capture and inquiry screens to accept data from users to update databases and to retrieve data.

Report/Update

Specification driven, with options for code complementing. Report and batch process component that produces reports and performs periodic batch updates.

N-Tiered Architecture

MenSitpi Framework and Applications developed using the Framework will be “**component based**” enabling **N-tiered Software Architecture**. The advantage of N-tiered technology is the ability to provide **scalability** of the application without discarding any hardware components on which investments have been made.

A user could start from a single machine in which all the components are executed and as the volumes grow, components could be placed in different vertical layers of component servers. Obsolete Client-Server



MENSITPI

Architecture in which almost all tools and application of today are based is a sub set of N-tiered architecture in which N is equal to 2 and cannot grow beyond.

Several **Component Servers** may be placed in **parallel** to **balance** the **load of the operations** with the existing equipment being supplemented by additional equipment instead of being discarded.

