STP in the Back Office:
An implementer's view

Arnab Debnath

CEO, Anshinsoft Corp.

ANSHINSOFT

Presentation Outline

- Perspectives on STP

- STP Solution : Implementation considerations
  - Architectural & Technology Considerations
  - The debate: "BUILD" vs. "BUY"
  - Total Cost of Ownership

- Case Study

ANSHINSOFT

Perspectives on STP

- **Intra-Firm** STP : reasonably similar across different capital markets

- **Inter-Firm** STP : different from market to market; governed by the law of the land and the evolving market practices
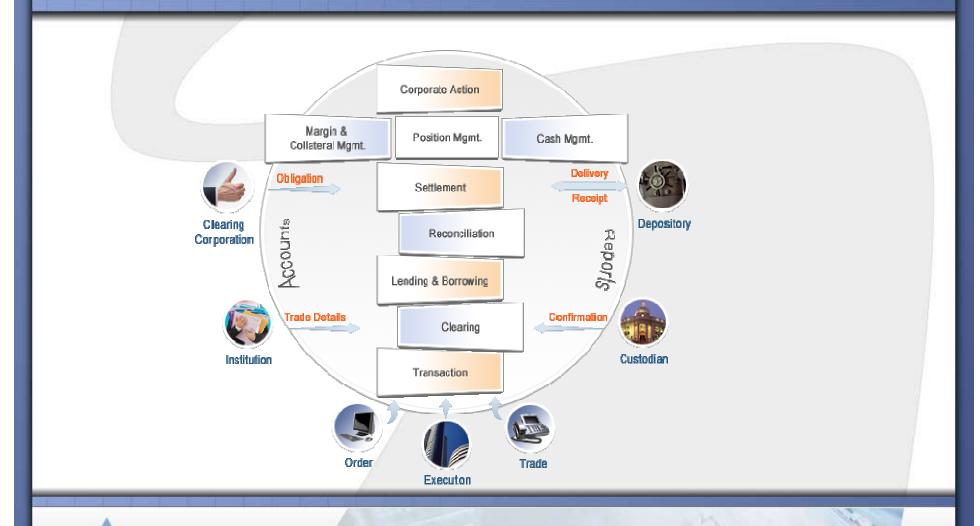
Many views on STP implementation

STP is . . .

- Reducing settlement cycles
- Elimination of manual intervention
- Reduction of operational cost & risk
- The backbone of efficient market mechanism

STP . . . implementer's view

- The real time system automating the entire post-trade life-cycle management, that provides the right balance between process efficiency, minimizing operational cost and limiting risk exposure

ANSHINSOFT

Implementer's view of STP

- **T+1 not top priority for securities industry**
  - New research challenges rationale behind reduction of Cost and Risk expected from T+1 implementation

    Tower Group report,
    "T+1: Cost, Risk, Benefit and Other Urban Legends"

- **Reference Data management viewed as the major obstacle in achieving STP**
  - Four out of five financial institutions view badly managed reference data as a major cause of failure in achieving intra-firm STP

    Report published by
    Tower Group, Reuters and Capco

ANSHINSOFT

- ## What role should regulators play in promoting STP?
  - – STP realistically can not be imposed on the marketplace from the top down, it should be allowed to create an organic STP structure.

  Daniel Doscas,
  HSBC Securities

- ## Has the STP direction changed over the past 5+ yrs?
  - – In 2000-2001, the securities industry faced about $1.8 trillion worth of trades remaining unsettled globally in single business day.

INNOVATION NEXT

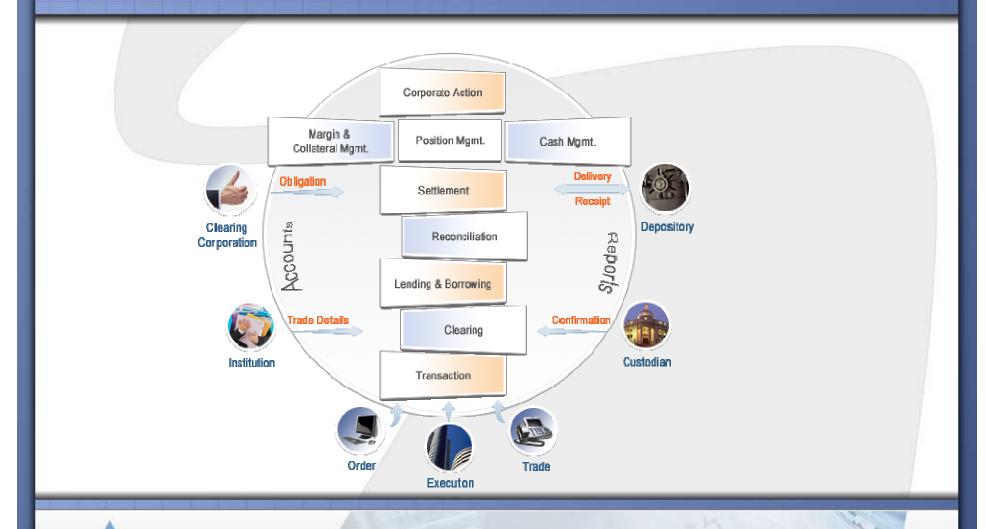## Current mindset ..

STP initiatives – How to be more Cost & Time-effective

Complete back office system overhaul is difficult/risky

Third party outsourcing not viable option for everyone

Off-shoring only makes sense for the 'big-boys'

What is the way to go ?

ANSHINSOFT

v 1.0

Modular STP Solution Framework

v 1.0

Presentation Outline

- Perspectives on STP

- STP Solution : Implementation considerations
    - Architectural & Technology Considerations
    - The debate: "BUILD" vs. "BUY"
    - Total Cost of Ownership

- Case Study

ANSHINSOFT

Architectural Considerations

- Architecture backbone

- Scalability

- Distribution

- Virtualization

- Customizability

- Preparing for tomorrow

Architecture : Backbone

- Tier based?

- Grid based?
  - Messaging grid (distributed messaging)
  - Processing grid (parallel processing)
  - Data grid (distributed caching)

- Service Oriented?

Everything towards the common goal …

ANSHINSOFT

v 1.0

Offer a powerful computing framework, capable of handling capacity transaction throughput, enabling clustering, high availability, location transparency and consistency of secured services across all the layers of the architectural stack.

ANSHINSOFT

v 1.0

- Scalability is a measure of how cost-effectively you can grow your operating capacity

- Plan ahead for a scalable system based on off-peak and peak trade volume (and future growth plan)

- Linear scalability – scaling OUT through additional resources (hardware, application instances)

Data distribution

- – Data grids (distributed cache, network attached memory)

Processing distribution

- – Cooperative parallel processing (map/reduce paradigm)

Messaging distribution

- – Message bus

Goal : Loosely coupled distributed services for implementing high performance and low latency STP system
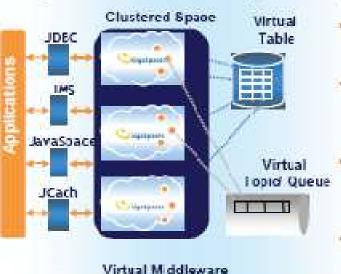
'Write-Once' business logic and scale anywhere

Virtualizing the Middleware using Space based technology
(figure copyright : Gigaspaces)

Architecture : Customizability

Rule based processing

- Market characteristics vary

- Localization for different regions

- Using rule engines for dynamic inference and externalizing the customization activities

ANSHINSOFT

Architecture for tomorrow – Multi-core

- Multi-core is here – STP applications need to take advantage multi-core platforms

- Functional languages will lead the way
  - More and more being used in the Wall Street
  - Erlang OTP platforms have proved to be multi-core ready

- Stateless scalability

- **Buy or build**: Requires active support / involvement of the management and the business/user community

- **Build**: Whether built in-house or by external vendor, active collaboration required from internal operations

- **Buy**: Even if functions/features match, operational and technical integration may not be obvious or trivial

- Off-the-shelf package solutions may require significant customization, raising the total cost of ownership

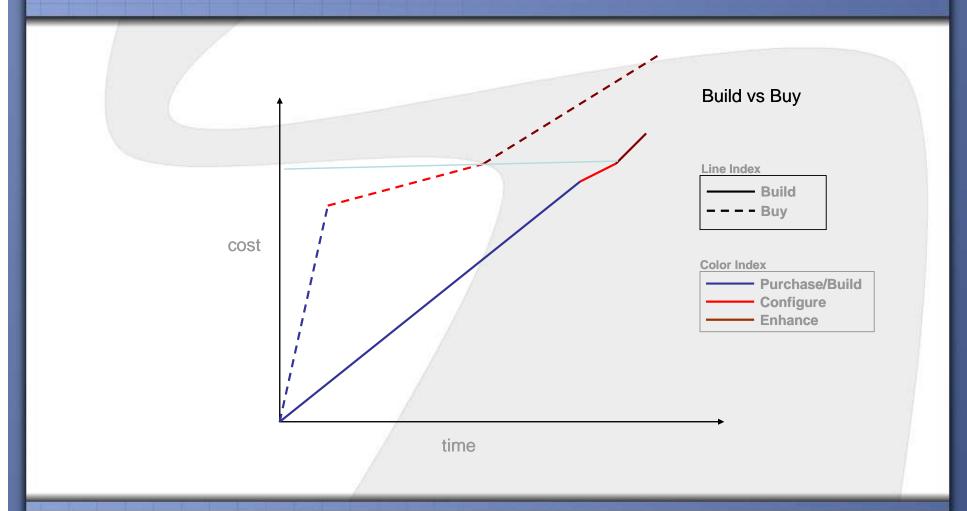# Factors Influencing Build or Buy

- Time to deploy (including integration/migration)

- Budget allocation (incremental vs. upfront)

- Firms internal resource profile and IT capabilities

- Handling future changes and enhancements

- Feature differences between a custom built solution and packaged solution and their criticality to the firm

- Total Cost of Ownership (not just the licensing cost)

Build vs Buy – cost & time estimation

Build vs Buy

cost

time

Line Index

———— **Build**

- - - - **Buy**

Color Index

———— **Purchase/Build**

———— **Configure**

———— **Enhance**

ANSHINSOFT

Build vs. Buy : Our Views

- Having the right balance between what you BUY and what needs to be BUILT on it

- No single solution truly fits all – so critically evaluate the application capabilities against your requirements

- BUY Customizable Solution Framework – as it may be difficult to address all custom requirements in a fully packaged solution

Pay careful attention to the real cost

- TCO is combination of procurement cost plus ongoing cost of support, operations and maintenance

- Ongoing cost factors depend on the flexibility/effort for:
  - Customization, Integration, Migration
  - System Testing & UAT, User Training
  - Supporting future business requirements & enhancements
  - Ongoing system maintenance – hardware and software
  - Managing operational scalability

Perspectives on STP

STP Solution : Implementation considerations
- Architectural & Technology Considerations
- The debate: "BUILD" vs. "BUY"
- Total Cost of Ownership

Case Study

Broker-cum-Custodian Back Office STP system

for

Global Financial Services Company

Customer

Securities broker-cum-custodian with operations in multiple capital markets (countries)

Goal

Implement and migrate to a new system that would eliminate their operational inefficiencies and enable them to meet future business demands -

Handle higher volume, Ready for 1-day settlement and have common solution backbone across global operations

Their Operational Issues

- Too many scattered legacy systems

- Scattered reference data for each legacy system

- User operation and training – multiple system knowledge required

- Lack of standards in information exchange protocol

- Old systems unable to handle increasing volumes

- Manual process for data verification/reconciliation

ANSHINSOFT

High-level Objectives

- Phase out scattered legacy systems with a single new system built on latest technology platform

- Unified system for handling multiple asset classes

- Same solution platform across multiple offices

- Seamless integration with the required legacy systems

- Centralize Reference Data repository

- Enable intra-firm STP and be ready for inter-firm (T+1)

- Be scalable for handling increasing trade volume

- Designed flexibly, for easier incorporation of changes in market regulations and firm's business practices in future

ANSHINSOFT

The Solution

Implementation of a clearing & settlement engine with real time interface between market intermediaries and straight-through transaction processing capabilities

- Comprehensive post-execution trade life-cycle management
- Works in real time and batch mode
- Supports multiple asset classes
- Supports cross border trades
- Integrated reporting solution
- Real time posting and general ledger journalization
- Integrated with FO and MO in real time
- Integrated with market intermediaries (where supported)

ANSHINSOFT

High Level Approach

- Custom solution was first implemented in their Asian operations in the following order: Japan, Hong Kong and Singapore.

- Followed by implementation for their US operation (NY) : the system has successfully completed UAT.
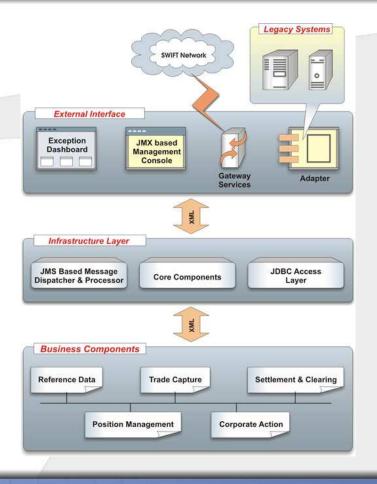
ANSHINSOFT

The Solution – Functional Components

- Trade capture and management

- Settlement processing

- Corporate action

- Position management (customer, inventory, nostro and stock record)

- General ledger

- Reporting (regulatory, transaction, risk assessment, audit and client reports)

- Exception monitor

ANSHINSOFT

Solution Architecture

ANSHINSOFT

Architectural Decisions

- ## Architected based on async. messaging and MOM
  - Reliable delivery
  - Standards compliant (JMS)

- ## Web based
  - Centralized deployment and maintenance
  - Standards compliant (Java EE)
  - Container based (Java EE) management

- ## Centralized exception handling and reporting

ANSHINSOFT

Scalable

- – Stateless service layer
- – Clustered

Fault Tolerant

- – Oracle RAC
- – Fail-safety of MoM

The Benefits

- Less number of disparate systems to use/manage/monitor

- Faster processing to handle increasing trade volume

- Real time information availability and decision making

- Less manual intervention

- Daily generation of all regulatory and client reports

- Faster exception handling, lowering operational cost & risk

- Quick detection of operational system errors due to an integrated application

- Automatic reconciliation and discrepancy identification

ANSHINSOFT

Challenges

- Varying levels of complexity for similar processes in different markets [complexity increases with market age]

- Unifying different market practices in a single platform

- Eliminating the local legacy systems in each operations

- User acceptability and inherent resistance to migrate to new system

- Balancing between cost, time and resources (with hard to scale knowledge constraints)

ANSHINSOFT

Lessons Learnt

- Have a proper reference data strategy. If not defined, insist on getting it defined before starting implementation.

- Everything cannot be made configurable – performance and configurability needs to be balanced.

  - Drastic differences in market practices may require some rebuild.

- Plan for performance early – evaluate/estimate current and expected /future performance metrics

- Set scalability expectations early – how much can be scaled and at what cost

- Involve users from initial stages to get to know usability requirements (and get buy-in for new system)

- Have a well-defined migration & implementation path

  - Estimate efforts for parallel run

ANSHINSOFT

Thank You

arnab@anshinsoft.com

ANSHINSOFT

v 1.0