Make or Buy Decision Factors:

World Bank Experience and Guidance Note for Treasury Management System

Custom-developed solutions
In custom-developed solutions, applications packages are conceived, designed and developed using internal or external expertise. In this approach, a full-fledged development team for each of the projects needs to be employed by the client Government department. These teams then develop new software packages tailored to the specific needs of the concerned Government department.

Features of Custom-made solutions:
- Can be customized according to specific requirements
- Can incorporate niche domain needs
- Can transfer specialized domain knowledge already built into any existing legacy systems
- Can exploit any established skill sets in custom software development
- More suited for division-wide or fragmented applications

Notwithstanding the above advantages world-wide trend shows that there is a rapid decline in custom development approach. The following are the reasons for this trend:
- Time taken for developing software in-house is typically very long
- Investment has to be made in developing IT resources with the government department / IT department, which depreciate at a very rapid rate
- Retention of the IT personnel is crucial and difficult, especially in government departments.
- Industry best practices may not be fully incorporated
- Co-ordination between departments / groups is very difficult and hence integration becomes a problem
- Version control becomes difficult
- Updates and improvements will take long time
- Dependency on platform and people is very high

Package-based solutions
In Package-based solutions, ready-made commercial off-the-shelf (COTS) packages are used for implementing the target applications. Worldwide trend in addressing application implementation is by using packages available in the market. The apparent advantage of using packaged solution is the time saved in development. Given below are a few other features of this approach.

Features of Packaged solutions:
- Robust, field tested software
- Reinventing the wheel for the same application avoided
- Easier migration to integrated solutions
- Best suited for enterprise-wide applications
- Incorporates best global processes
- Acts as a change agent for business process reengineering
Decision 1: Package based vs. Custom Development

In the following paragraphs, key parameters for deciding the software development approach are discussed. Ground realities in each client country is very different. The client government department must review their current status and decide on the key future requirements from the proposed systems.

Key Criteria
Three key criteria must be considered in a make vs. buy decision for software solutions: (1) the richness of the required functionality; (2) the volatility of the functional requirements, and (3) the client organization’s capacity for software engineering and maintenance.

These criteria can further be analyzed under the following parameters for purposes on decision-making:

a) Business and technical functionality: Packaged solution is generally comprehensive in business/technical functionality and ensure high level compliance with national/international standards laid down by professional and regulatory bodies. But 100% requirements can be met only by custom developed software. High level of functionality-fit is the most important consideration before deciding in favor of a packaged software solution.

b) Experience in developing application packages: Experience in development and operation of custom made solutions is important if a custom development approach is to be selected. In absence of necessary experience it would be preferable to select an appropriate package software and get it customized to department’s requirements. Organizations, particularly in public sector, must realistically assess their ability to develop and maintain the skill levels required to develop in-house systems and to keep pace with the technology shifts.

c) Legacy systems: Strong presence of legacy systems in the organization and the need to upgrade and integrate with these, would require a custom development based approach.

d) Need for Integrated application suite: Package software is a more proven and reliable approach in case the integration with other systems and modules is a major requirement.

e) Need for global best practices: Package ERPs incorporate best practice systems and proven lessons learned from implementation of systems in dozens of organizations.

f) Implementation approach - enterprise wide or single department: Custom development is suitable for limited single department/single function kind of implementations. However, if the systems are to be implemented organization wide and across several levels, a robust and proven packaged solution would mitigate the risks of large scale implementation failure.

Key issues in implementing packaged solutions:

Custom applications require thousands lines of code which could be seen as time consuming and costly and may slow the implementation cut-off time draining resources however it will fulfill the agencies demands and requirements. Package software may be perceived as an approach to reduce programming cost, time and effort, thus providing an more immediate return on investment. However, some key issues should be carefully considered while selecting the package software option:
a) One major concern is the adaptation of and Re-engineering processes necessary to have business processes and rules fitting in the existent and sometimes rigid functionality of package applications. Contract and negotiation are also important aspects to be considered for evaluating in-house purchasing expertise, time needed for negotiations, legal aspects, mutual responsibilities, maintenance, bug fixes, pricing and payment strategies.

b) Customizing the packages: Most of the packages today come with the global best practices, future upgrades and support. However, these packages need to be customized and modified to meet the client departments functionality (Refer Annex –1 for an illustrative list). It is advisable not to modify the package in those areas where any such change causes loss of some of the best practices.

c) Importance of integration: A critical issue in the current IT environment at the client government organization is the lack of integration amongst different processes. Worldwide trend shows that this has been the prime concern for many organizations across the industry sections. It is quite apparent that when the applications are integrated, the benefits from the total solution far exceed the aggregation of individual solutions. Packaged ERP solutions provide seamless integration across functions and modules.

d) Importance of Selected Vendor:
Vendors assume great significance in case of packaged software (Refer Annex –2 for a partial list of Treasury system vendors). Credibility of the vendor, and the support for the package is very important. Worldwide trend is a partnership mode of functioning. This helps vendors in implementing their best practices gained through their earlier experiences and to retain the best practices in the client government organization. Investments in R&D, both in absolute terms and in terms of percentage of revenue plough back, their road map for the product and interest in continuing business in India, are some indicators of vendors’ reliability.

Comparative Analysis of Packaged (COTS) vs. Custom-built software solution approach:
<table>
<thead>
<tr>
<th>Parameter</th>
<th>COTS</th>
<th>Custom-built</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance and upgrades</td>
<td>Maintenance and upgrade is more or less assured, but subject to payment of annual maintenance fees.</td>
<td>Maintenance subject to agreement with developer or may be done by client’s IT unit, when present. Upgrades not usually available</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Limited to the extent that the vendor would allow.</td>
<td>Highly flexible, as required by client.</td>
</tr>
<tr>
<td>Integration with other CSC systems</td>
<td>Limited</td>
<td>Integration parameters can be included in the functional specifications and design of the system.</td>
</tr>
<tr>
<td>Performance and Quality</td>
<td>Stable as it has been installed previously and subjected to testing both in simulated and actual production environments</td>
<td>Still subject to rigorous testing, even after full deployment. Needs continued trouble shooting and debugging.</td>
</tr>
<tr>
<td>Documentation and Training</td>
<td>Documents are available for evaluation. Training packages are also readily available</td>
<td>Documentation and training are only available at the end of the development cycle.</td>
</tr>
<tr>
<td>Software Evaluation</td>
<td>The software can be evaluated immediately. In most cases, the client can try the product before buying it.</td>
<td>No evaluation on the software itself can be done immediately.</td>
</tr>
<tr>
<td>Legal Redress</td>
<td>Agreements usually contain provisions that customized functionality is the responsibility of the client.</td>
<td>Agreements contain provisions that acceptance of the software functionality is decided by the client and disagreements are usually resolved in favor of the client.</td>
</tr>
</tbody>
</table>

Guidelines
In order to be able to properly decide on whether COTS software is better than a custom built application under given circumstances, certain rules of thumb can be used. These are as follows:

- If the COTS software addresses more than 70% of the business and technical requirements, it is advisable to consider the COTS software approach. Otherwise, a custom built solution would be cost effective and more appropriate considering all other factors.
• If the organization does not have an IT department or need to enhance it to build the custom solution, then a COTS application may be more appropriate.

• If a generic application with customization is acceptable for the enterprise level integration, then COTS software may be acceptable. If it is not acceptable, then definitely specific requirements need to be defined.

• If the scope of the system bridges large and complex business units, then COTS software may be applicable.

• If a comparison of the costs of the Customized COTS software vs. Customized software is high, then the customized software solution may be considered.

• If the customization of the COTS takes longer than 6-8 months, then the Customized solution should be considered.

• If the ownership of the IPR is important, leverages most of the future deployments without the expensive licenses and shifts most of the advantages to the client, then the Customized Solution should be considered.

Decision 2: ERP vs. Non-ERP

Once it is decided to take the package-based approach, different possibilities of package implementation are to be taken up for further analysis. Keeping in view the industry practices worldwide and the requirements of the client department, two solutions may emerge.

  a) Enterprise-wide Resource Planning packages are increasingly becoming popular for complete solutions. In ERP solutions, an ERP package forms the central package and drives the application architecture.
  b) The alternate approach is to implement independent specialized packages for domain specific applications. These packages are referred to as ‘best-of-breed’ as they have a complete fit for those specific domain needs. In ‘best-of-breed’ solutions, a collection of packages forms the application architecture.

ERP solutions
ERP packages view the organization as one entity providing seamless integration of the modules for the independent functions. ERP packages can be altered to suit the needs of the organization in which it is implemented with the help of the exhaustive libraries provided for this purpose. As the number of installations grow the learning from these organizations are incorporated in the product by the vendor making it a rich repository of best practices in the industry.

Features of ERP based solution :

  • Interfaces are minimized due to the high degree of integration among the various business modules
  • Industry best practices already incorporated for global standardized processes
  • Robust and proven functionality for the value chain processes
  • Uniform user interface and ease-of-use
• Availability of pre-developed templates for industry specific processes, which further reduces
time to configure and implement
• Uniform scalability and expandability
• Rapid system development and deployment possible proving rapid return on investment (ROI)
• Enterprise wide data model
• Decision making tools in-built for Enterprise - Top management can have consolidated current reports.

Best-of-breed solutions (non ERP)

Many packages have been developed to address the issues of individual specialized functions. These packages, developed by the domain experts and software experts, address the entire issues specific to those functions. For those processes, which are specific to an industry, these best-of-breed solutions become almost essential.

Features of Best-of-breed approach :
• Maximum functionality fit achieved for the specified applications
• Vendor focused on specified area and hence enhancements very effective
• Advantage of the developers long experience in that process can be leveraged
• Decision can be taken at a lower level
• Suitable for a phased roll-out

Treasury Management Software Solution – World Bank’s experience and recommendations

ERP approach has proven to be effective for those applications that form the core of organization’s functions. Finance & Treasury functions are core to government’s working and can be standardized across departments and various levels of the central and state governments. ERP with its integrated approach and extensive functionality towards application packages, fulfils most of the requirements of the target functional area. When it comes to the processes specific to government functions like land records, citizen welfare schemes, health services delivery, water sector modeling and simulation, agriculture sector farmer services etc., a generic ERP will not be able to comply with all the requirements of the government functions. In exceptional circumstances, specific requirements of the client government may also warrant the use of a customized solution approach.

The basic functions required for public sector treasury management are rather straightforward and can be developed as bespoke software with a level of effort typically within reach of finance ministries in most middle income countries. Furthermore, a bespoke, basic treasury management solution can be supplemented with packaged software utilities such as for report writing and policy-based security.

However, when the requirements extend to all or most of the complex functions involved in advanced public financial management, such as budget formulation, budget execution, general ledger, accounts receivable, accounts payable, cash management, human resource management, fixed asset management, requisitioning, procurement, contract management, debt management, etc., developing bespoke software that integrates the solution to all these functions requires a software engineering effort of very large scope.

Unfortunately, experience confirms time and time again that large software development projects executed by government institutions have a staggering frequency of failure. Therefore, government officials who sponsor such projects take an enormous responsibility and risk. Projects of this kind are known to produce no useful software after burning 500% of the original budget and time. The risk is
not only of delay and extra cost. The distinct risk is that the software will actually never work. This can happen for several reasons:

1. The complexity of software engineering increases dramatically with the functional scope of the system. Managing this complexity is beyond the capabilities of even the best of project managers unless they adhere to highly sophisticated work methods involving consistent use of technical standards, design tools, documentation tools, quality measurement tools, communication tools, etc. The level of this so called “software engineering capability” that a technical team would need to successfully develop a complex, integrated treasury management system can only be acquired over several years of disciplined application over a large number of projects. This is how the software industry was created. Trying to replicate this level of capability on the first try for a single project is usually a heroic proposition and a foolish risk. It is like a ministry of transportation deciding to manufacture the truck fleet needed for a large road network simply because it has highly sophisticated mechanic engineers in its staff.

2. Software engineering at this level of complexity is unusually dependent on a stable cadre of high caliber engineers and project managers. This stability is achieved in the software industry mainly through very high levels of compensation and job enrichment, which are way out of reach for the pay scales and technical exposure opportunities in government. High rotation of technical staff and particularly of project managers can quickly doom the best conceived software engineering project.

3. A substantial level of change in functional and technical requirements is inevitable in integrated treasury management systems. COTS software accommodates this change through generalized functions configurable through parameters. Developing these generalized functions is of course far more complex that developing the bespoke functions required in a specific case. Without them, however, the software cannot easily accommodate functional change and software maintenance becomes a very complex and ongoing activity. Public sector O&M budgets can seldom provide sufficient resources for this activity with predictable consequences of early obsolescence for the software system.

For the above reasons, among others, the Bank seldom endorses bespoke software development as a solution approach to treasury management systems. This notwithstanding, several countries in south and central America have chosen this approach with mixed results. The bespoke treasury management systems developed by these countries in the early 90s went through two or three generations and several of them are considering replacing these with packaged software platforms.

The Bank recognizes that the licensing costs of top of the line ERP and financial software packages can be viewed as excessive, particularly when such software is applied to early stages of treasury management functionality. However, the risks and potential costs of a bespoke approach are so much larger that the Bank focuses its advice on helping country clients design procurement documents that balance qualification requirements, mandatory technical specifications and merit evaluation methods so as to maximize competition and minimize price of packaged software solutions.
Annex –1

**Indicative Functional & Technical requirements Checklist for Treasury Systems**

**INVESTMENT/DEBT INSTRUMENTS HANDLED:**
- Commercial paper
- Certificates of deposit
- Acceptance credits – bank/trade bills
- Eurobonds
- Floating rate notes
- Repos and reverse repos
- Letters of credit

**CASH/INVESTMENT:**
- Call accounts
- Money market deposits
- Secured deposits
- Government bonds
- Treasury bills
- Equities

**DEBT:**
- Money market borrowing
- Loans – complex profiles
- Syndicated debt
- Multi-currency facilities
- Commercial paper programmes
- Trade bills
- Finance leases

**INTEREST RATE RISK MANAGEMENT:**
- Interest rate swaps
- Swap options
- Caps, collars, floors
- FRAs
- Interest rate futures
- Interest rate/currency swaps

**COMMODITIES:**
- Commodities – full capability or commodity defined as currency

**COUNTERPARTY RISK MANAGEMENT:**
- Risk by counterparty
- Risk by counterparty group
- Country/region risk
- Credit ratings - application

**BANK RELATIONSHIP MANAGEMENT:**
- Comparative bank quote history
- History of facility utilisation
- Current headroom availability

**LIMITS MANAGEMENT:**
- Counterparty limits
- Dealer limits
- Limit breach reporting

**CURRENCY MANAGEMENT:**
- System base currencies
- Restrictions on currencies
- Cross currencies
- Spot and forward transactions – conventional/inverse switch
- Currency swaps
- Option period value dates
- Currency options - OTC/traded
- Currency options – American/European
- Currency options – Other
- Currency futures
- Euro

**DATA INPUT/USER INTERFACE:**
- Deal number format (manual/automatic)
- Dealer scratch pad
- Transaction input screens
- Deal linking
- Building instruments
- Holidays database
- “Help” facility
- Language

**CASH MANAGEMENT:**
- Bank balance and statement detail – data management
- Daily cash position worksheet
- Bank reconciliations
- Payments – electronic, fax etc.
- Settlement netting
- Multi-lateral netting
- Cash flow projections

**RISK MANAGEMENT:**
- Mark to market
- Position revaluation
- Sensitivity analysis
- Value at risk
- Stress testing
- Performance analysis

**DECISION SUPPORT:**
- Strategic modelling
- Instrument pricing tools
- Graphics capability
- Instrument calculator
- Historic rates database
- Yield curves stored

**ACCOUNTING:**
- Nominal ledger
- General ledger
- Treasury cash book
- Inter-company accounts
- Journal entries
- Generated book exchange gains/losses
- Other accounting support
- Taxation support

**REPORTING:**
- Range of standard reports
- Report writer
- Output capability
- Screen views
- Diary report
- Maturity schedules
- Position reports
- Cash flow forecasts
- Confirmation letters
- Scheduled report production & distribution

**SYSTEM ACCESS & SECURITY:**
System administrator • Access rights defined • Use of passwords • Levels of verification and authorisation • Dual
authorization available • Audit reporting • Other security

SYSTEM INTERFACES:
Rate feeds from market information systems – Reuters, Bloomberg,
Telerate/Bridge, other • MIS links real-time? • Import bank balances and statement information • Poll parsing
capability • Export to electronic bank payment systems • Import/export cash flows • Electronic confirmation
matching systems • Links with netting systems • Links to third party general ledger systems • Internet/intranet •
Spreadsheets

SYSTEM ENHANCEMENTS:
Enhancements due for release • Other enhancements planned

FUTURE PLANS:
Improvements due for release • Improvements planned

ACCOUNTING STANDARDS:
US – FAS 133 • International – IAS 39

UPGRADES:
Date of system introduction • Date of first installation • Current version number • Date of last upgrade • Date of
next upgrade • Frequency of regular upgrades

EXISTING USERS:
Numbers of existing user licences • Geographical spread • Typical user profile • Named examples of existing
users • Site reference availability

SUPPORT:
Worldwide office locations • Country locations of support • Availability of support • Method of support • Policy
regarding visits to users • Implementation support • Initial system training provided • Number of dedicated support
staff • User group meetings frequency & location.

TECHNICAL ISSUES:
Underlying database language • File server system • Workstation hardware • Server hardware • Single user
operating system • LAN system • Wide Area Network • Source code availability • Relational databases

WEB BASED TECHNOLOGY:
Core system structure • Communications application • Bureau service provider

SUPPLIER DETAIL:
% of revenue from treasury management systems
PRICING: Is system modular? • Licence purchase • Lease or other options • Minimum system cost • Maximum
system cost • Typical system cost • Training costs • Annual maintenance and support
## List of Some Packaged Treasury Systems

<table>
<thead>
<tr>
<th>Vendor</th>
<th>System Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaps consultants</td>
<td>ADAPS Treasury Management System</td>
</tr>
<tr>
<td>AFA Systems International Ltd</td>
<td>AFA Musketeer, DART ++, Musketeer Light, Musketeer Plus</td>
</tr>
<tr>
<td>BRADY Ltd</td>
<td>Trinity</td>
</tr>
<tr>
<td>CCK Treasury System Pty Ltd</td>
<td>Guava Dealer/Risk/Operations</td>
</tr>
<tr>
<td>Capix Software (Europe) Ltd</td>
<td>Capix</td>
</tr>
<tr>
<td>City Financials Ltd</td>
<td>eTC</td>
</tr>
<tr>
<td>CMFC Finance Software GmbH</td>
<td>CMFC ID</td>
</tr>
<tr>
<td>Concept</td>
<td>Treasury Manager</td>
</tr>
<tr>
<td>CS Lucas Associates PTE ltd</td>
<td>Alpha Station</td>
</tr>
<tr>
<td>Double Helix Software and Services ltd</td>
<td>Dealerbase</td>
</tr>
<tr>
<td>Eco Finance</td>
<td>Integrated Treasury System ITS</td>
</tr>
<tr>
<td>Econintel Treasury Systems Ltd</td>
<td>ETS for Windows</td>
</tr>
<tr>
<td>Eurobase International Group</td>
<td>Siena</td>
</tr>
<tr>
<td>Financial Sciences Corporation</td>
<td>ATOM (Advanced Treasury Operations Management)</td>
</tr>
<tr>
<td>Financial Software Systems</td>
<td>Spectrum</td>
</tr>
<tr>
<td>FMS Group</td>
<td>SGAT + CCEMS (Risk Management)</td>
</tr>
<tr>
<td>FXpress Corporation</td>
<td>FXxpress Foreign Exchange</td>
</tr>
<tr>
<td>Gateway Systems, Inc.</td>
<td>Exposure Management System</td>
</tr>
<tr>
<td>INSSINC</td>
<td>Gateway Treasury Workstation</td>
</tr>
<tr>
<td>Integrity Treasury Solutions plc</td>
<td>Integra - T.com</td>
</tr>
<tr>
<td>Investment Intelligence Systems Corporation</td>
<td>Spearhead</td>
</tr>
<tr>
<td>Logotech Systems</td>
<td>Treasury and Cash Flow Management (TACSYS)</td>
</tr>
<tr>
<td>Manex Treasury Systems Ltd</td>
<td>Parity</td>
</tr>
<tr>
<td>MISYS International Banking Systems</td>
<td>OPICS</td>
</tr>
<tr>
<td>Openink</td>
<td>Findur</td>
</tr>
<tr>
<td>Oracle Corporation</td>
<td>Oracle Treasury</td>
</tr>
<tr>
<td>PeopleSoft</td>
<td>Peoplesoft Treasury Cash/Deal/Risk Management</td>
</tr>
<tr>
<td>Reuters Limited</td>
<td>Kondor+</td>
</tr>
<tr>
<td>Reval</td>
<td>Reval</td>
</tr>
<tr>
<td>Richmond Software Ltd</td>
<td>Millennium / Odassey / E-Cash</td>
</tr>
<tr>
<td>Salmon Software Ltd</td>
<td>Salmon Treasurer - 2001</td>
</tr>
<tr>
<td>SAP AG</td>
<td>SAP R / 3 System</td>
</tr>
<tr>
<td>Selkirk Financial Technologies Inc.</td>
<td>Treasurer Manager</td>
</tr>
<tr>
<td>Silver-Net Computer Systems Ltd</td>
<td>FRM (Financial Resource Management)</td>
</tr>
<tr>
<td>SimCorp Ltd</td>
<td>IT / 2</td>
</tr>
<tr>
<td>Software Options International Ltd</td>
<td>BOSSPRO Treasury &amp; Trading</td>
</tr>
<tr>
<td>SS&amp;C Technologies Inc.</td>
<td>SS&amp;C Debt &amp; Derivatives 2000</td>
</tr>
<tr>
<td>Summit</td>
<td>Summit Systems</td>
</tr>
<tr>
<td>SunGard Treasury Systems</td>
<td>Resource IQ</td>
</tr>
<tr>
<td>SunGard Treasury Systems</td>
<td>iCMS Treasury Systems</td>
</tr>
<tr>
<td>SunGard Treasury Systems</td>
<td>Global Treasury + Risk Management System</td>
</tr>
<tr>
<td>SunGard Treasury Systems</td>
<td>Quantum</td>
</tr>
<tr>
<td>SymPro Inc.</td>
<td>SymPro Treasury Management Software</td>
</tr>
<tr>
<td>Thomson Financial</td>
<td>CAPITAL</td>
</tr>
<tr>
<td>Tietonator Financial Solutions</td>
<td>System 10</td>
</tr>
<tr>
<td>Treasurynet Limited</td>
<td>TreasurlNet</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Trema Treasury Management AB</td>
<td>FinanceKIT</td>
</tr>
<tr>
<td>Wall Street Systems, Inc</td>
<td>The Wall Street System</td>
</tr>
<tr>
<td>XRT - Cerg</td>
<td>CF / Cash and Treasury Master</td>
</tr>
<tr>
<td>XRT - Cerg</td>
<td>Globes$</td>
</tr>
</tbody>
</table>