Integrating public debt management within Integrated Financial Management Systems (IFMIS)

by

Mr. Marcelo Abalos

DMFAS Programme, UNCTAD

The views expressed are those of the author and do not necessarily reflect the views of UNCTAD.
Integrating Debt Management Software with Integrated Financial Management Systems (IFMIS)

Mr. Marcelo Abalos
Systems Analyst, Team Leader

Geneva, 14 November 2017
Integrating Debt Management Software with Integrated Financial Management Systems (IFMIS)

Outline:

- Challenges faced by DMOs
- Solutions provided
- Benefits of integration
- Recommended activities
- Key success factors
- Lessons learnt and future developments
Challenges faced by DMOs

The evolution of Information Requirements and Information Technologies, including changes in best practices of Public Administration confronts DMOs with:

- Enforce data quality and security
- National capacity for development and maintenance of interfaces
- Needs to automatically and securely integrate Front, Middle and Back office procedures and subsystems
- Other public administration areas (treasury, budget & accounting)
- High demand for Integration of DMOs
Solutions provided

As a result of the evolution of requirements, DMOs increasingly require advice and support to software solutions providing standardized interfaces.

**DMO-IFMIS**
Greater interactivity with other Information Systems in use in the DMO or within the context of Public Administration procedures.

**DQ & SECURITY**
Integrate a variety of software solutions to reinforce data quality and security.

**ITS**
Standard features and data structures to support system integration.

**Advice**
Increased support and advice for systems integration and reporting, enhancing data analysis and communication.
Solutions provided

Increased support and advisory services for integration processes and reporting, enhancing data analysis and communication

- Analytical tools (DSM, MTDS, DSF), to export information from the Debt Management System
- Standard processes (e.g. budget, drawings, payments, etc.)
- Auction platforms, sharing information to Debt Securities modules
- Market platforms (Bloomberg, Reuters), import reference data
Benefits of Integration

Integrating the development of interfaces within the context of the Debt Management Software project implementation and development facilitates:

1. **Security**
   - as the new modules do not bypass any of the standard security checks, authorizations, etc.

2. **Transparency**
   - All data is consistent and trustworthy through all the integrated applications.

3. **Simplicity**
   - Easiest development of country specific functions to support integration and data migration.

4. **Adaptability**
   - Easier maintenance and adaptation of existing links to newer versions of the DMFAS software.

5. **Capability**
   - Building local capacity for the maintenance of interfaces.
Benefits of Integration

In addition to technological advantages of integration, we can also consider the compliance with most recent definitions of best practices in Debt Management, particularly those related with Data Quality Dimensions such as:

1. **Accuracy**
   - the extent to which data reflects the reality, reinforcing the reliability of the information

2. **Completeness**
   - All recorded data is present wherever needed through all the integrated applications

3. **Timeliness**
   - Ensures data is processed in due time, once the required processes are triggered, helping to ensure deadlines are respected

4. **Uniqueness**
   - Facilitates the unique recording principles, making transactions to be processed efficiently

5. **Auditability**
   - Ensuring that any transaction can be tracked to its originating transaction.
**Needs Assessment**

On demand, provide advisory services for assessing the integration of IFMIS with the Debt Management Software.

- Assess the requirements for integration and eventual modifications (process & applications)
- Agree with government on requirements for implementation.

**Analysis of local system**

Environment and needs for the interface (Conceptual Design), in close collaboration with IFMIS technical and functional teams.
Physical design
Creating the technical definitions of the interface between systems, including a Workshop on Procedures, Database Structures, tables and relationships.

Development support
The local IT staff will maintain the detailed design of the interface; Validate the modifications and updates to the detailed design of the interface.

Recommended activities
IFMIS Integration

- Assist local IT staff in the elaboration of the detailed design (Offer technical template)
- Preparation and presentation of detailed design in form of a workshop
- The local IT staff developing the programs for the link/interface between the systems.
- UNCTAD providing remote support and guidance through its helpdesk.
Support during development process
Development and maintenance of interface performed by IFMIS IT staff
Remote support and guidance through helpdesk.

Supervision/Coordination
Systems Analyst/Designer supervising the work of developers and consultants while coordinating ensuing activities

Validation of the Interface
Onsite review and test of the link/interface in coordination with local IT staff and the related users

Successful implementation
IFMIS Integration
Planning and communication
Creating plans allowing information sharing and coordination among parties during all phases of the project

Capacity building
Developing capacity on local IT teams in order to ensure maintenance and evolution of the interfacing modules

Standardization
Use agreed ITS standards to maximize flexibility and data exchange compatibility. Ensure understanding by all parties

Contingency planning
Developing an emergency response plan that includes coordination and communications

Apply Industry patterns
Meet user needs and follow accepted usability engineering practices for interactive systems
Lessons learnt

Failure factors

Outsourcing
- Lack of expertise regarding functionality, making the modules to be underdeveloped and failing on the functionality tests or in a live situation.
- Poor analysis of requirements creating unresolved integration issues during early phases of the project

QA
- Non Accurate development time and thorough system testing to ensure systems are working properly after integration
- Poor users and experts participation in the different project phases

Project Plan
- Non accurate estimations of resources;
- Badly defined system requirements;
- Inability to handle project's complexities

Implementation
- Poor project management
- Unmanaged risks
- Use of immature technology
- Poor development practices and standardization
- Political pressures

Communication
- Poor communication among management, users and developers
- Poor reporting of project's status
Lessons learnt
Reducing failure factors

Outsourcing
- Ensure experts availability to validate OC deliverables (documentation, specification, etc.)
- Create management committee to periodically review the resolution of integration issues raised

QA
- Allow resources for development and testing to ensure systems are working properly after integration
- Active participation of users, experts during project phases
- Managerial support to project

Project Plan
- Review project planning against results of Needs Assessments;
- Validate system requirements with users and experts;
- Periodically review project’s milestones

Implementation
- Perform periodic risk evaluations
- When using new technology ensure validation of Proof of Concepts
- Perform periodic reviews of development practices and standards in use
- Plan realistic schedules

Communication
- Ensure periodic communication among management, users and developers
- Create and share periodic reports of project's status, following standard templates
Future developments

The evolution of technologies and requirements require solutions for interfacing systems within new configurations.

1. Reporting
   Increasing support for Business Intelligence platforms, enhancing data analysis and communication.

2. Technology advances
   Cloud computing & SaaS contexts, providing high availability, scalability and cross platform support.

3. Capacity Building
   Continue improving on capacity building activities for local IT and IFMIS staff.

4. Reinforce Security
   Enhance standard interfaces for integrating a variety of software solutions reinforcing data quality and security.
Thank you

Questions?