

PARTICIPANT COURSE MATERIALS

# Information Systems for Microfinance



CGAP CONSULTATIVE GROUP TO ASSIST THE POOR

---

**NOTE** The Participant Course Materials contain the main technical messages and concepts delivered in this course. It is not intended to substitute for the full information and skills delivered through the individual courses in the ***Skills for Microfinance Managers*** training series. During the actual courses, key concepts are presented with case studies, exchange of participant experiences, and other activities to help transfer skills. Users interested in attending a training course should directly contact CGAP hubs and partners for course dates and venues or visit the CGAP website at [www.cgap.org/html/mfis\\_skills\\_microfinance\\_manag.html](http://www.cgap.org/html/mfis_skills_microfinance_manag.html). CGAP would like to thank those who were instrumental to the development and design of the original course that led to this participant summary: Janis Sabetta, Indrajith Wijesiriwardana, Andrew Mainhart, Laura Frederick, Shirley Lunde, Dayo Forster, Ruth Goodwin-Groen, Brigit Helms, Jennifer Isern, Leslie Barcus, Tiphaine Crenn, and all CGAP training hubs and partners. Copyright 2003, The Consultative Group to Assist the Poor (CGAP).

# Contents

<b>Overview and Goals</b> .....	<b>1</b>
Overview .....	1
Goals of the course .....	2
<b>What Is an Information System?</b> .....	<b>3</b>
<b>Components of an Information System for Microfinance</b> .....	<b>5</b>
Core systems .....	5
Additional systems .....	5
<b>Information: Key to Success</b> .....	<b>7</b>
<b>Manual or Computerized?</b> .....	<b>8</b>
<b>System Development Life Cycle</b> .....	<b>9</b>
<b>Preparation</b> .....	<b>10</b>
Can an IS help solve all of an MFI's problems? .....	10
What is the role of the IS task force? .....	10
How do you select who goes on the task force? .....	11
Are you ready? .....	12
<b>Needs Analysis</b> .....	<b>13</b>

Why do a needs analysis? .....	13
How can you do a good needs analysis?.....	13
How should all this information be presented? .....	14
<b>Selection .....</b>	<b>16</b>
1. Review resource requirements for feasibility .....	16
Do you need to automate, and how much?.....	16
How do you prioritize IS components when considering automation? .....	17
2. Conduct a high-level scan of available systems and establish short-list candidates ...	18
What kind of software solution should you aim for? .....	18
3. Make initial recommendation and seek senior management approval .....	19
4. Apply due diligence tools to analyze short-list candidates .....	19
How can you find out more about the software that is available? .....	19
How do you reach a final decision? .....	20
5. Make final recommendation and obtain management decision .....	20
<b>Implementation .....</b>	<b>21</b>
<b>Optimization and Maintenance .....</b>	<b>26</b>
Definitions .....	26
Issues to consider.....	26
<b>Additional Resources .....</b>	<b>27</b>

# Overview and Goals



## *Overview*

As microfinance institutions (MFIs) grow and become more business oriented, managers have found they gradually lose their ability to maintain personal contact with what is happening at the field level. They realize they cannot adequately manage their portfolio and financial operations without better information.

**No single information system (IS) will meet every MFI's information needs.** The IS needs of institutions differ in size and in complexity. These differences are a function of many organizational variables, including volume of transactions, methodology, regulatory environment, infrastructure, and overall readiness for change, as well as the resources available.

This course presents a process that can be used to guide you through the steps needed to understand what your own unique information system will require. Although some parts of this process emphasize computerization, many are equally applicable to manual information systems. Developing and managing an information system is not a simple linear process; it is an ongoing transformation process that requires close examination of what you have now, what you need now, and what you might need for the future.

## ***Goals of the course***

**This course provides guidelines for planning, developing, implementing, and managing the right information system for your microfinance institution.**

- It presents a systematic process to understand what is needed to create and manage an information system for microfinance.
- It defines information systems and explains why they are important to the business of microfinance.
- It identifies specific information needed to make business decisions in microfinance.
- It analyzes critical business processes (in the context of an IS) and streamlining for efficiency.
- It uses a framework tool to evaluate information system software options
- It clarifies the processes required to (1) plan; (2) conceptualize and assess; (3) select; (4) develop and acquire; (5), implement; and (6) manage, maintain, and optimize an information system for your MFI.

# What Is an Information System?



<b>Data</b>	<b>≠</b>	<b>Information</b>
Unprocessed basic facts about the activities of a business that give no insight by themselves		Data transformed into a meaningful form that helps someone to make sound decisions or to gain insight

Any MFI has many users of information, operating at different levels. Also, various stakeholders make decisions based on different information and require different levels of detail. A branch manager may find a client ledger useful for making a decision on a particular loan disbursement, yet at the head office, the financial officer is interested in simply knowing the balances outstanding on all loans given to clients at that particular branch. Information requirements can therefore vary considerably throughout the institution.

Table 1 illustrates the idea that a particular decision may need to be supported by information from different sources.

**Table 1. Decision Based on Information Sources**

<b>Decision</b>	<b>Information Needed to Make the Decision</b>	<b>Information Source (Who/How/Where)</b>
Disbursing a Loan	<ul style="list-style-type: none"> <li>➤ Cash Flow Position</li> <li>➤ Client Profile</li> <li>➤ Client Repayment Record</li> <li>➤ Loan Application Form</li> <li>➤ Loan Officer Details and Recommendation</li> </ul>	<ul style="list-style-type: none"> <li>➤ Accounting</li> <li>➤ Client Records</li> <li>➤ Client Records</li> <li>➤ Loan Management Records</li> <li>➤ Loan Officer Reports</li> </ul>

This table helps us understand that an information system attempts to pull together all these various data sources so that information is readily accessible. We will define an information system as follows.

**An information system is the series of actions involved in**

- **Capturing raw data from various sources**
- **Processing the data into usable information**
- **Storing the information**
- **Disseminating information in the form needed**

# Components of an Information System for Microfinance

A full information system (IS) includes all the systems (both manual and computerized) used by an institution to generate the information that guides management's decisions and actions.

## ***Core systems***

- **Accounting:** Records accounting details and provides complex tools for financial management
- **Portfolio:** The core business for many MFIs, manages all transactions relating to the loan portfolio
- **Deposit tracking:** Manages all transactions related to savings if this product is offered

## ***Additional systems***

- **Customer information:** Detailed information about customers that may be used to understand the customer base
- **Human resources:** As organizations grow, management of information related to staff becomes more complex and can be automated, may consequently be linked to a performance-based incentive scheme.

- **Reporting:** Reports can be generated within each subsystem; it may also be necessary to extract information across subsystems and recombine the information for more complex reporting requirements.

A key success factor for an MFI is making sure all of its systems are suitably linked to allow timely and accurate sharing of data between the various systems.

The linking of all information systems is known as *systems integration*. This process can be both costly and time consuming. Moreover, if any one of the systems is changed or upgraded at a later date, chances are that extensive integration work will have to be done yet again.

# Information: Key to Success

**Good information → Good business → Successful MFIs**

Characteristics of good information systems

- Timely
- Reliable
- Accurate
- Easy to use
- Appropriately detailed reports, with just “enough” information
- Meet needs of various user categories
- Secure
- Good internal controls built in
- Benefit of having information exceeds cost to produce

# Manual or Computerized?



An information system does not need to be fully computerized; all or part of it may be manual. There are advantages and disadvantages, some detailed in Table 2, when considering whether any particular subsystem should be computerized.

**Table 2. Comparison of Manual and Computerized Systems**

	<b>Advantages</b>	<b>Disadvantages</b>
<b>Manual information system</b>	<ul style="list-style-type: none"> <li>• Less expensive initially</li> <li>• No computer literacy requirements</li> <li>• Adaptable</li> <li>• Places few demands on infrastructure</li> </ul>	<ul style="list-style-type: none"> <li>• Increasingly expensive as the MFI grows</li> <li>• More subject to human error</li> <li>• Somewhat limited growth potential</li> <li>• Often less productive</li> </ul>
<b>Computerized information system</b>	<ul style="list-style-type: none"> <li>• Less expensive as the MFI grows</li> <li>• Less subject to human error</li> <li>• Facilitates growth</li> <li>• Higher productivity</li> <li>• Permits more sophisticated business processes</li> <li>• Enhances data security</li> </ul>	<ul style="list-style-type: none"> <li>• More expensive initially</li> <li>• Requires computer-literate users and support personnel</li> <li>• Software may not be adaptable</li> <li>• Places demands on infrastructure (power, temperature controls)</li> </ul>

# System Development Life Cycle



The System Development Life Cycle (SDLC) is a process that helps to formalize the steps involved in planning an information system project, understanding its goals, selecting a solution, and then putting it in place and successfully managing it over its lifetime.

- **Preparation** Establish a good foundation by planning well
- **Needs Analysis** Understand your requirements and prioritize
- **Selection** Choose the optimal solution given your resources
- **Implementation** Make it work
- **Optimization and Maintenance** Continue to improve on its performance



### ***Can an IS help solve all of an MFI's problems?***

An IS cannot solve all of an MFI's business and operations problems. Some may be due to lack of staff training or internal controls.

As a first step, an institution needs to determine reasons for needing to change. It then needs to set specific measurable goals for the new system, for example, greater efficiency by decreasing costs by 10 percent. If the MFI decides to change its IS, the next step is to create a task force to lead the process through planning and implementation.

### ***What is the role of the IS task force?***

An effective task force should

- Determine frequency of its meetings and how it will function
- Involve all the main stakeholder groups fully
- Ensure that the objectives of the projects are communicated to staff
- Assess current practices
- Lead the development of the system
- Document all the main processes and decisions made

## How do you select who goes on the task force?

- **By organizational structure:** select from branch staff, head office staff, senior management
- **By experience:** project planning, change management, information systems, operations, communication, human resources
- **By task force role:** need a project leader to drive the work, need a project champion to rally the staff together and promote the project's benefits

### ***More on running task forces***

- ***How many people should be on it? Suggest 4 to 10.***
- ***How long should it run? Suggest 6 months to a year.***
- ***Can the members change? Yes, especially as project progresses and functions of task force change.***
- ***How do you keep people motivated? Make sure they have enough time (reallocate job responsibilities if necessary) and are recognized and rewarded for their effort.***

## ***Are you ready?***

Is your institution ready with...

- **A business plan and budget?** The business plan and budget include details on planned growth rates, changes in methodology, new products and services, branch expansion, financial projections. All of these influence the requirements of the system and the resources that can be made available for the project.
- **Documented policies, procedures, and practices?** An IS cannot outperform the business operations it is intended to model. You need to make sure that the current procedures are designed to work efficiently, or update and streamline them.
- **Accurate, complete historical data?** Correct client balances and transaction histories must be available.
- **Internal controls?** Supervisory checks and balances should be in place and monitored adequately.
- **Infrastructure?** Consider where the new computers go and what they need to function properly: air conditioning, power-surge protection, generator, new building.
- **Personnel?** Review staff requirements: hire new staff, reallocate duties of existing staff, use consultants.

# Needs Analysis



## ***Why do a needs analysis?***

You need to understand thoroughly your organization's needs. Once you have a physical document available, you can also use it to help evaluate potential software or determine specifications for a custom system.

## ***How can you do a good needs analysis?***

It is recommended that you undertake the following steps:

- Document and review current business practices, including information flows
- Redesign any inefficient procedures
- Analyze current and future needs
- Assign priorities of needs

### ***More on redesigning inefficient procedures***

- ***Continuous Process Monitoring:*** Advocates incremental changes to an existing—and working—process
- ***Business Process Re-engineering:*** In the extreme, assumes the current process is irrelevant and a “clean slate” perspective is required to redesign business processes

## ***How should all this information be presented?***

A needs analysis report should include the following sections:

- Overview of MFI
- Goals and scope of system
- Executive summary of information-system priorities
- Detailed description of needs

### ***More on information flows***

Define and diagram:

- **Data sources**
- **Points of transformation (for example, updating client's last payment date)**
- **Where information is used for decision making**
- **Where and how information is stored**
- **How, when, and to whom information is communicated**

Some techniques that could be used:

- **Flowcharts—graphical representation of business processes**
- **Process mapping—framework analyzing processes, broken down by tasks**
- **Activity theory—can be used in process mapping to break down descriptions of actions even further**

## A word on security

Security is an important issue to think through at this stage. It is important to consider what internal controls would be adequate to deal with each type of security risk. There are additional kinds of risks that do not relate directly to the information system, but are broader, involving risks to the financial institution as a whole. You need to design a security strategy to cope with the following events:

- Unauthorized access and changes to system data
- Employee fraud
- Theft of client data files
- Erroneous transaction entries as a result of miskeying of account numbers
- Loss of system data because of equipment failure or power failure
- Posting transactions on wrong date
- Loss of mission-critical system because of fire, flood, or other disaster
- Failure of hard disks in system servers
- System failure caused by loss of communications lines
- Loss of funds resulting from fraudulent simultaneous withdrawals on a single client account at multiple branches

Security will always be important, but especially so during implementation and optimization.



When deciding on an information system, consider the following steps:

1. Review resource requirements for feasibility
2. Conduct a high-level scan of available systems and establish short-list candidates
3. Make initial recommendation and seek senior management approval
4. Apply due diligence tools to analyze short-list candidates
5. Make final recommendation and obtain management decision

## ***1. Review resource requirements for feasibility***

### **Do you need to automate, and how much?**

Consider the following factors:

- **Staff.** Do staff have the right skills and experience for an automated IS? What kind of training do they need? Will there be much resistance or is there a willingness to change?
- **Technology.** What infrastructure is required for network and communications? Are branches to be decentralized or fully integrated? How can the technology be supported?
- **Time.** When do you need it? How does your business plan to grow? What impact could growth have on your IS requirements?

- **Costs.** What are the short-term costs for buying hardware and software? What are the longer term recurrent costs for supplies, connectivity, staff, facilities? What is the available budget? How do you get the best value for money?

## Common Difficulties

Underestimate time → Project behind schedule  
Underestimate cost → Project over budget

### How do you prioritize IS components when considering automation?

- Consider the following components essential: accounting, portfolio management, and deposit tracking (if savings products are offered).
- Consider a customer information system if you have a lot of clients in several different geographic areas who are involved in different types of businesses. It will be particularly useful if your institution is interested in customer-focused product development, planning to grow its size and range of products, or is facing competition.
- Consider a reporting system if your institution consistently misses out on making good management decisions due to a lack of timely information or is unable to provide its staff with all the information they need to perform their jobs.
- Consider a human-resource system if your institution is having trouble attracting and retaining the type of staff it desires or is spending more time than it wishes on keeping track of basic staff information (holidays accrued, incentive systems, job descriptions, performance reviews, etc.).

## 2. Conduct a high-level scan of available systems and establish short-list candidates

### What kind of software solution should you aim for?

With a realistic view of requirements and resources, the MFI can consider various alternatives: buying an off-the-shelf package, finding one that can be modified, or building one from scratch (see Table 3).

**Table 3. Comparison of Buying, Modifying, and Building Software**

	Advantages	Disadvantages
<b>Buy</b>	<ul style="list-style-type: none"> <li>• Low to medium cost</li> <li>• Likely to operate relatively error free</li> <li>• Short schedule for implementation</li> </ul>	<ul style="list-style-type: none"> <li>• Dependent on outside technical support</li> <li>• Unlikely to fully match institution's policies and procedures</li> <li>• Cannot be modified as institution changes, unless upgraded by vendor</li> </ul>
<b>Modify</b>	<ul style="list-style-type: none"> <li>• Likely to operate relatively error free</li> <li>• Medium schedule for implementation</li> <li>• Can be closely adapted to institutions' policies and procedures</li> </ul>	<ul style="list-style-type: none"> <li>• Medium to high cost</li> <li>• Dependent on outside technical support</li> <li>• Future modification costly</li> </ul>
<b>Build</b>	<ul style="list-style-type: none"> <li>• Technical support is in-house</li> <li>• Can be fully adapted to institution's policies and procedures</li> <li>• Can be modified to match institutions' changes</li> </ul>	<ul style="list-style-type: none"> <li>• High cost</li> <li>• Will require debugging</li> <li>• Long development schedule</li> </ul>

### ***3. Make initial recommendation and seek senior management approval***

Generally, if you find a software application that meets at least 75 percent of your needs, buy rather than build it, even though you may need to change some of your internal processes and policies. The resulting software solution is likely to be cheaper and more likely to meet industry standards.

### ***4. Apply due diligence tools to analyze short-list candidates***

#### **How can you find out more about the software that is available?**

The evaluation framework (see [“Additional Resources”](#) for more information) provides a way of assessing the fit of a software for your organization, considering the following factors:

- **Functionality and expandability.** What subsystems are included, what kind of methodologies does it handle? What options are available for payments, branches, languages, currencies?
- **Usability.** How easy is it to use? What kind of user interface is offered?
- **Reporting.** What reports are available? Can users design new reports?
- **Standards and compliance.** Does it meet accounting standards? Does it comply with government and regulatory requirements?

- **Administration and support.** What security options are offered? How robust is the software? What technical support is available? What is the strategy to upgrade software?
- **Technical specifications and correctness.** What is the architecture? What kind of technology is required? How well does it handle large numbers and date formats?
- **Cost.** What is the purchase price? What are the annual support costs?

**Use independent reviews such as those available at the CGAP Information Systems Services ([http://www.cgap.org/iss\\_site/](http://www.cgap.org/iss_site/)) to get basic information about a possible product.**

### **How do you reach a final decision?**

- Make a short list of products to investigate further
- Collect information from the vendor, in writing
- Review a demonstration version of the software
- Check references with other clients using the product and/or the vendor
- Visit an MFI using the software or see a live demo using your own data

## ***5. Make final recommendation and obtain management decision***

Get the task force to agree on one product and make a formal recommendation to management.

# Implementation

## Proper Planning Prevents Poor Performance

In designing an implementation plan, consider the following:

- **Hardware procurement.** What hardware should you buy and what are the memory, speed and space requirements?
- **Infrastructure development.** What type of network is required? What facilities are needed, for instance, generator, telecommunications?
- **Software installation.** Where should it be installed first?
- **Testing.** How do you ensure the system works as you require it to?
- **Software modifications.** Are any further changes required to the system?
- **Documentation.** Are user and administrator manuals available?
- **System configuration.** How should the system parameters be set?
- **Data transfer.** How will you transfer old data to the new system?
- **Staffing.** Do the staff have adequate training and time to learn the new system?
- **“Live” date.** Will the old system and the new system run at the same time during the changeover? If so, for how long? When will the new system be considered “live”?
- **Internal controls.** Are there adequate checks such as audit and access trails? What manual or electronic confirmations are needed?
- **Institutional interface.** What about procedural issues that the software cannot help to manage?

- **Data security.** Are there measures in place to adequately protect the data? Is there a plan for how to operate if the computers are down?

***Make sure you specify what the MFI is responsible for and what you expect the vendor to do.***

### ***More on implementation strategies***

There are different ways to carefully phase in your new information system:

- ***Functional:*** Phase in the system over time by installing one business function at a time, for example, portfolio tracking, accounting, deposit tracking, human resources
- ***Branch:*** Phase in the system by installing it at one geographic location at a time
- ***Feature:*** Phase in the system by activating core features initially, then activating additional features when appropriate or necessary

## ***More on testing***

Computer programs are incredibly complex constructions. The only way to make sure they will do what they are designed to do is to confirm the functionality through testing. (Table 4 lists the differences in scope for two software solutions.)

Activities performed during testing include both a review of code printouts and dynamic testing. During dynamic testing the code is executed under controlled conditions (often one program instruction at a time) that permit the tester to verify that the program performs as expected.

- The most common test strategy is to start testing as early as possible, often testing only a specific section or subroutine of the program. This is called **unit testing**.
- Once unit tested, pieces of the program are joined, like pieces of a puzzle. The process of joining units of code is called integration. As units are joined, testing continues (now called **integration testing**), until eventually all units of the program have been joined, or integrated. The focus at this stage is to confirm that the syntax of the programming language has been correctly applied. After the program has been fully integrated, the programmers perform a complete test of the application (a **full system test**). Before starting a full system test, the program should be initialized with a sample of representative data loaded into the database in order to simulate actual operating conditions.
- A final test phase is the **user acceptance test**. At this point, the testing is focused on confirming that the programmers have followed the customer's design specifications. (Operational functionality should already have been confirmed during the full system test.)

Unit testing and integration testing are typically done by the programmers. A staff member, preferably assisted by a programmer, should perform user acceptance testing.

**Table 4. Scope of Testing for Different Software Solutions**

<b>Software Solution</b>	<b>Scope of Testing</b>		
	<b>Unit Testing and Integration Testing</b>	<b>Full System Test</b>	<b>User Acceptance Test</b>
Off-the-shelf software package	Not included (done by software publisher's programmer before releasing the software for sale)	Included (done by end-user's software evaluator prior to purchasing the software)	Included (done by an end user before system goes live)
Rewrite of existing code, or custom coded solution	Included (done by end-user's programmer while developing the program)	Included (done by end-user's programmer while developing the program)	Included (done by an end user before system goes live)

## ***More on documentation***

Types of documentation that may be required:

- **Source code:** An overview of the software design, a detailed description of the variables, high-level description of each routine.
- **System administration manual:** Explains the design of the software in lay and technical terms, has step-by-step instructions on how to install the software, and how to manage the software (including re-indexing the database, setting up a new user, changing the configuration).
- **Users' manual:** Should be presented in easy lay terms and explain methodically how to use the software, providing examples where appropriate. Much of this should also be available as online help.
- **Training manuals:** Similar in language to the system administrator's and user's manuals, but more like a workbook and should also have practice data available so users can perform the different transactions and procedures on their own.
- **Institution documents:** Process and information flows that show how staff interact with the actual system. Internal control manuals and other policy and procedure guidelines are included here. Also, keep notes on the processes and decisions made by the task force as a record for any new staff members responsible for overseeing IS.
- **Contract:** All licensing agreements should also be kept safely.

# Optimization and Maintenance



## *Definitions*

- **Optimization:** Ensure the system is used to maximum advantage
- **Maintenance:** Keep hardware and software functioning and updated

## *Issues to consider*

- **End-user documentation** to be updated whenever a policy, process, or procedure changes and may be kept as a “living document” online.
- **System administration log** to keep detailed records of routine tasks such as backups and all other significant system events. A hard copy and an online version must be available (in case the system is not functioning for the very reason that has been documented electronically).
- **Hardware maintenance** to ensure that hardware and networks function as well as they should.
- **Software upgrade documents** to explain how to install new versions of software to computers and replicate to all end-user workstations.
- **Requests for software modifications** to collect and collate users’ requests for changes to the software, to forward to vendor.
- **Staff training** to accommodate additional training programs that must be designed for new employees or staff who change their responsibilities, or simply to reinforce or deepen existing skills.

## Additional Resources

- Information Systems Services, CGAP, [http://www.cgap.org/iss\\_site/](http://www.cgap.org/iss_site/)
- The following three documents can be downloaded from the CGAP site at [http://www.cgap.org/iss\\_site/documents.html#mis](http://www.cgap.org/iss_site/documents.html#mis):
  - *MIS for Microfinance Institutions: A Handbook* by C. Waterfield and N. Ramsing, CGAP, Washington, DC: 1998.
  - *MIS for Microfinance: An Evaluation Framework* by A. Mainhart, Microenterprise Best Practices Project, Development Alternatives/U.S. Agency for International Development, Bethesda, MD: 1999.
  - "Management Information Systems Between Salvation and Frustration," *Nexus* (Fall 2000), Small Enterprise Education and Promotion Network, Washington, DC.