# A Framework for Transparency in SHGs

A.B. Sagar

Department of Computers and Information Sciences
Hyderabad Central University, Hyderabad
India
bablusagar@gmail.com
www.uohyd.ernet.in

**Abstract.** A Self Help Group [SHG] is a small homogeneous gathering of persons who join on a voluntary basis in order to undertake some common activity through mutual trust and mutual help. SHG system is conceptualized basically to address the problem of rural unemployment, and empowering people to make them economically self-dependent. But, there is a possibility of it being turned into a commercial unit negating the very thesis it espouses.In the last decade we have seen an increased use of the term "transparency" in different contexts such as business, political affairs, education, administration and government. Transparency is considered an indispensable ingredient in social accountability and necessary for preserving and guaranteeing ethical and fair processes. Transparency is related to visibility of information. Lack of transparency leaves the organization and stakeholders in blind states. The growing importance to the requirement of transparency in businesses was the motivation to study Transparency in SHGs. This paper persents a framework for transparency and also outlines the implementation of transparency through Member Behavioral Model (MBM) and Task Execution Cycle (TEC).

**Keywords:** SHG, Transparency, Self Help Groups, social values.

## 1 Introduction

Transparency implies visibility of information related to financial and non-financial matters of the organization and its stakeholders. It should be noted that transparency does not mean opening up the intellectual property files or company's classified documents. Transparency has been, for long, a general requirement for good governance. The right to be informed and to have access to the information has been an important issue in modern societies. Transparency — which can be defined as the accessibility of information to stakeholders of a business, regarding matters that affect their interests[1] — can shape and revolutionize business practice in the present modern society.

Recent observations revealed that there is an increase in societal attention to the issue of transparency. It was also predicted that transparency will become the required

[1] Tapscott and Ticoll 2003, p. 22.

N. Meghanathan et al. (Eds.): CCSIT 2012, Part II, LNICST 85, pp. 612–626, 2012.

premise for gaining and maintaining customer trust and good relationships with all stakeholders. The same is true for SHGs also. Transparency in SHGs can lead to trust of all stakeholders.

## 2 Transparency

Operational functionality of SHG cheifly consists task execution. Every SHG divides the tasks among the members. Tasks execution happens at member level. And these tasks constitute the tasks of the SHG.

Transparency in SHG is related to financial and non-financial reporting. Financial reporting includes tracking of monetary data Non-financial reporting is reporting on task execution.

SHGs' businesses involve several interdependencies. Mutual trust between interdependencies is pivotal. For mutual trust, the interdependencies viz. customers, suppliers and funding agencies connected with SHG, want the SHG information to be open and visible. Also, transparency in a SHGs business builds Trust( between stakeholders and SHG), mutual faith (between SHGs), and confidence (in the involved members of the SHG).

### 2.1 Degree of Transparency

We can classify transparency into three degrees : opaqueness, translucency and clarity. Two more degrees can be described namely "black hole" and "dazzle". They are used to describe the two extreme situations where no information is released by a SHG or too much information is disclosed, respectively.

Opaqueness is when a SHG does not disclose any information to its stakeholders and hence there is no transparency, So a transparent SHG should not have opaque degree in its Transparency.

$$\therefore \ T_{\text{SHG}} \nRightarrow \text{ opaque}$$

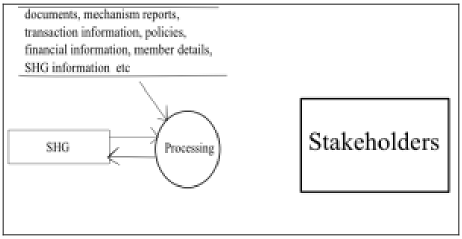

documents, mechanism reports,
transaction information, policies,
financial information, member details,
SHG information etc
SHG
Processing
Stakeholders

**Fig. 1.** Opaque SHG

Translucency is when a SHG discloses its information partially. Therefore a translucent SHG is not a transparent SHG. A SHG cannot be tranparent until it discloses all of its information.

$$\therefore \ T_{\text{SHG}} \nRightarrow \text{ translucent}$$

Clarity is when a SHG discloses all of its information. A SHG having clear disclosure of information is a transparent SHG.

$$\therefore \ T_{\text{SHG}} \nRightarrow \text{ clear}$$

## 2.2 Dimensions in Transparency

Klotz et al. (2008) suggests nine dimensions of transparency. They stratify their nine dimensions into three distinctive groups – recognition, facilitation and enabling. We adopt only six of Klotz et al's nine dimensions, stratifying these by the dimension of time (pre, per, and post). Thus each of the task will have three phases viz. pre-activity, per-activity and post-activity. Pre-activity transparency within and between two SHGs involves actors recognition of responsibilities and interdependencies. Per-activity transparency within and between two SHGs involves actors recognition of status and problems. Post-activity transparency within and between two SHGs involves the facilitation of performance understanding and feedback.

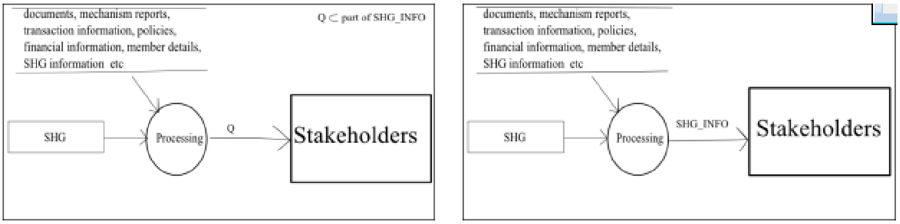

documents, mechanism reports,
transaction information, policies,
financial information, member details,
SHG information etc
Q ⊂ part of SHG_INFO
SHG
Processing
Q
Stakeholders
documents, mechanism reports,
transaction information, policies,
financial information, member details,
SHG information etc
SHG
Processing
SHG_INFO
Stakeholders

**Fig. 2.** Translucent & Clear Transparencies

Let 'w' be the activity to be done by the SHG. Let $w_i$ , $w_j$ , $w_k$ be the pre-activity, per-activity and post-activity phases of the work w. Each phase requires communication with stakeholders. Suppose a SHG's activity (w) is to irrigate and supply five tonnes of corn, then this activity can be divided into three phases: pre-activity ($w_i$ ), per-activity ($w_j$ ) and post-activity ($w_k$ ). Each activity involves communication with entities like governing bodies, subordinates, members, peers or competitors. In $w_i$, communication with supervisor/governing bodies involves describing how much land they are going to use, expenditure estimates, how many members are going to work in it, skills and experience possessed by the members, what fertilizers and chemicals are being used, what techniques are being employed, how waste is disposed, how much each member earns, delivery dates, other responsibilities undertaken by the SHG, etc. And communication with the subordinates involves describing terms and conditions laid down by the supervising/governing bodies, number of hours each member should work, wages for the members, responsibilities of each member, skills and experience required in the members, etc. Communication with the peers and competitors include enquiring regarding problems that are faced in their experience, enquiring how they solved specific issues, seeking suggestions, viewing their data, reports, bills, etc.

The transparency of activity 'w' is $T_w$ . $T_w$ is sum of transparencies of the three phases i.e. transparency of pre-activity ($T_{wi}$ ), transparency of per-activity ($T_{wj}$ ) and transparency of post-activity ($T_{wk}$ ).

$$T_w = T_{wi} + T_{wj} + T_{wk}$$

Interestingly, there is a direct relationship between the degree of transparency and the phases of activity. Degree of transparency is progressive. It progresses with the progressing phases of the activity. Thus, prior to pre-activity the degree of transparency is null (or *opaque*). If the transparency conditions are met in the pre-activity phase, the degree becomes *translucent*. If the transparency conditions are satisfied at per-activity phase and post-activity phase, then degree of transparency becomes *clear*.

**Table 1.** Degree of Transparency & Activity Phases

| Degree | Activity Phases |
|---|---|
| Opaque | Before Pre-activity |
| Translucent | After Pre-activity<br>After Per-activity |
| Clear | After Post-activity |

The below figure shows the progressing transparency degree with progressing activity phases.

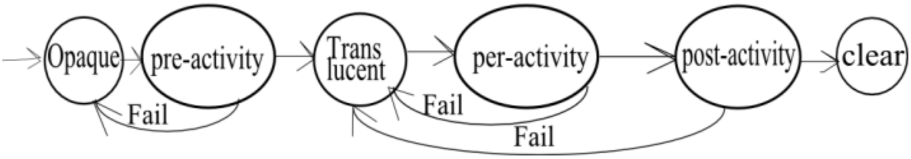


**Fig. 3.** Progressing Transparency Degree With Progressing Activity Phases

It is observed that after each activity phase, the degree of transparency is increasing. So it is of interest to us to define the cheif constituent of each activity phase which affect the degree of transparency. At each activity phases we define mandatory transparency dimensions. Though transparency dimensions are task- and situation-dependent, some dimensions are mandatory. Klotz et. al. (2008) suggested nine dimensions of transparency. We are adopting six of Klotz et al's nine dimensions, stratifying them into the these three activity phases (pre, per, and post)[2]. Recognition of responsibilities and interdependencies are primarily of concern before the realization of a given activity (pre-activity). Recognition of status and problems are primarily related to transparency into an ongoing activity (per-activity). Similarly, understanding of performance and feed-back are related to post-activity transparency.

[2] A Framework for Transparency. Økland Andreas , Lillebo Børge , Amdahl Eva , Seim Andreas.

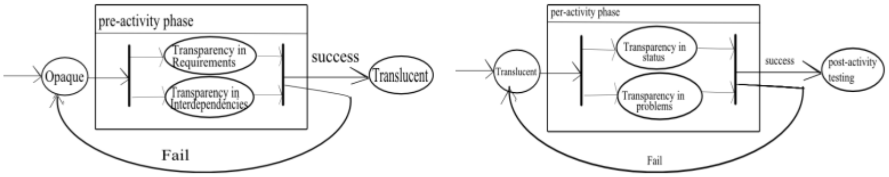


**Fig. 4.** Transparency dimensions in Pre-activity Phase & Per-activity Phases

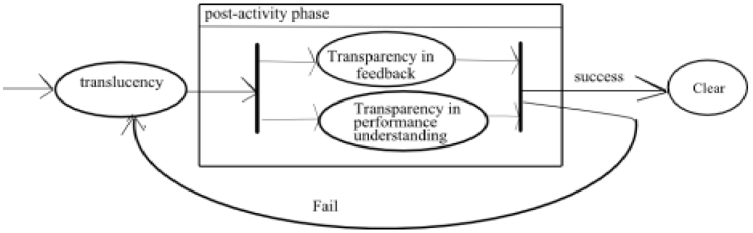


**Fig. 5.** Transparency dimensions in Post-activity Phase

## 2.3 Perspectives in Transparency

There are three main perspectives in Transparency: static transparency, dynamic transparency and radical transparency.

*(1) Static Transparency:* In Static Transparency reporting information is standardized and is done in official formats — such as the financial statements required by various government agencies, comprehensible format for customers, etc. However this is not a very efficient type of transparency because in this type, disclosure is "telling" rather than "sharing", and the flow of information is mainly unidirectional i.e. from the SHG to stakeholders. Since there is no on demand display of information, it does not fully portray the meaning of transparency.

${}_{\text{static}}\text{T}_{\text{SHG}}$ means unidirectional flow of information from SHG to Stakeholders.

$$
{}_{\text{static}}T_{\text{SHG}} \Longrightarrow \forall info \in SHG, S \in Stakeholders,\\
flow(info, SHG, S) \wedge \neg flow(info, S, SHG)
$$

flow(info,SHG,S) indicates that the information ('info') flows from SHG to Stakeholders ('S'). Since this is Static Transparency, the info flow is only unidirectional i.e. SHG to S. But the converse, i.e. flow(info,S,SHG) does not hold true. That is, info could not flow from Stakeholders (S) to SHG. There is no possibility for the stakeholders to request and obtain information of their choice. They only get what the SHG chooses to reveal. This kind of transparency is also called selective transparency. Thus static transparency gives liberty to the SHGs but not to the stakeholders.

A SHG adopts static transparency when its rules for reporting are all predefined (by the governing bodies), and it need not respond to stakeholder's queries.

*(2) Dynamic Transparency* is where the organization and its stakeholders can exchange, share and compare information and adapt its online behavior and electronic requests and queries to the answers and reactions of respective counterparts.

$$_{\text{dynamic}}T_{\text{SHG}} \Longrightarrow \forall info \in SHG, S \in Stakeholders,$$
$$flow(info, SHG, S) \wedge flow(info, S, SHG)$$

*(3) Radical Transparency* refers to the capability of a firm's top management to employ internet-based technologies, such as rss, blogs and collaborative websites, in order to create a direct and continuous dialogue with customers and other stakeholders. Radical transparency may be represented by:

$$_{\text{radical}}T_{\text{SHG}} \Longrightarrow \forall info \in SHG, S \in Stakeholders,$$
$$stream(info, SHG, S) \wedge stream(info, S, SHG)$$

***stream(info,SHG,S)*** is similar to the flow function but the difference being that the flow is continuous.

# 3 Implementating Transparency

The implementation of transparency can be made through member behavior model and task execution cycle. The member behavioral model outlines the behavior of every member and SHG. The Member Behavioral Model (MBM) provides an imprint of generic behavior of each member or SHG with respect to different types of items i.e. tasks, requests, reports, acks, etc.

## 3.1 Member Behavioral Model (MBM)

The primitives of the MBM are explained below:

**TMonitor:** The TMonitor is a continuously running process which keeps monitoring for items like tasks, acks, requests, reports, etc. When it receives an item it immediately forwards it to Rpt.

**Rpt:** Rpt has varied functionalities depending on the item it receives. If item is a task, it will notify the task sender about the receipt of the task. And it will forward the task to ExTsk for execution of the task. If the item is an ack, it will store the ack in the Repst against the corresponding task it has dispatched. If the item is a request, it will query the Repst and generate a report and dispatch it to the requester. If the item is a report, it will store the report in the Repst against the corresponding task.

**ExTsk:** The functionality of ExTsk is to log the stages a task passes through when a SHG/member executes the task. If the received task is feasible to that member (or SHG), execution is started. Else, if the task is partially feasible, then task is split into three portions (feasible part, assign part, remainder part). The assign part is assigned to a different member/SHG. Soon after the report is received that the assign part is executed, the remainder part is executed. After completing the exection of the whole task, it sends the task completed notification to Rpt to be forwarded to the notification recipients. While executing the task, it will write the intermediate task status to the store Repst through Rpt. So that the status of task is available at every executing phase of the task.

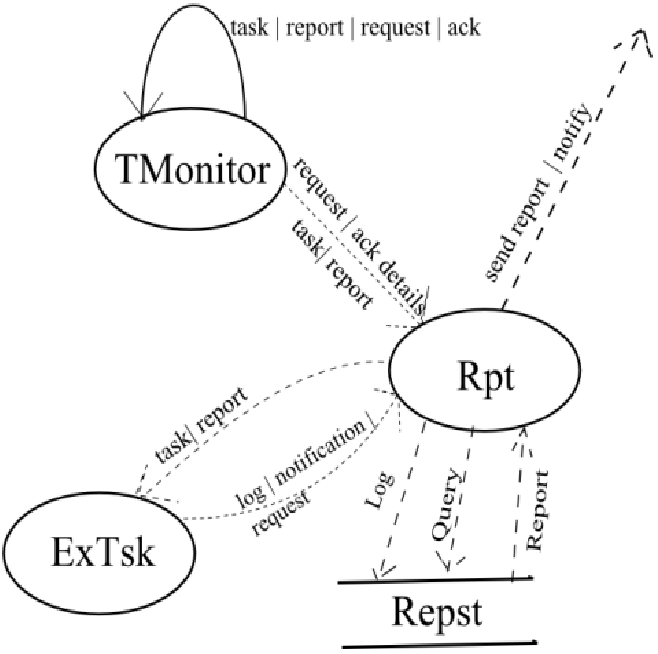


**Fig. 6.** Member Behavioral Model (MBM)

### 3.2 Task Execution Cycle (TEC)

The Task Execution Cycle (TEC) gives the different states that a task undergoes during task execution. The different states are described below:

**Recv:** The task is in Recvd state if it is just received. It will wait there until the member/SHG is ready to take it up and start initiation and execution.

**Initiate:** The task moves to Initiated state if the member/SHG has considered execution.

**Execute:** The task moves to Executing state when the member/SHG has started execution or is continuing exectuion after waking up from sleep.

**Assign:** The task moves to assigned state if the task was assigned to a different member/SHG.

**Sleep:** The task moves to sleep state once the task was assigned.

**Abort:** If the task was found to be infeasible or if the assignment fails, then the task moves to aborted state.

**End:** The final state of every task is the End state.

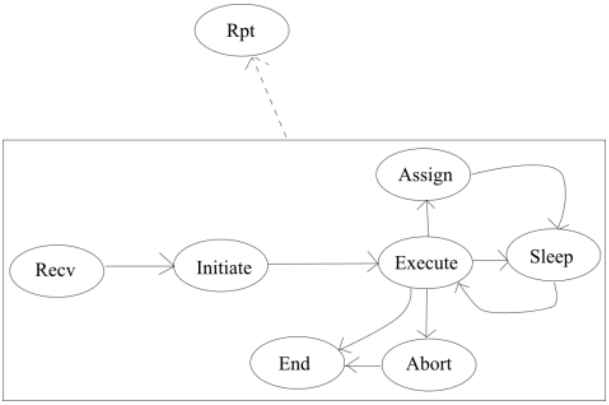


**Fig. 7.** Task Execution Cycle

Now implementation of transparency in task execution can be done by transparency messages defined on the TEC. Every task passes through the states given in Task Execution Cycle. For each state, transparency messages provide financial and non-financial information. When the task reaches a state, a transparency message is sent to the notification recipient. The transparency messages carry both predefined information and also dynamic information.

### 3.2.1 Task Properties

Executing a task is the basic operation of member and SHG. So we start by defining tasks. A task has all required information about the financial and non-financial details. Every task, besides other attributes, must have the following attributes:

*Task Id:* Identification token of the task

*Parent Task:* If the task is subtask of a bigger task, then this is the id of the parent task.

*IsPartitionable:* Yes, if the task can be split and distributed; No, if it has to be executed at only one place.

*Risks:* Risks involved with the task.

*Task Type:* Whether the given task is technical or non-technical.

*Creation Date:* Date when the task was created.

*Excluded Executors:* Restricted executors list.

*Task Stakeholders:* All the stakeholders who are related to the task.

*Task Status:* Status of the task, indicating how far the task was completed.

*Skills Needed:* List of skills needed to execute the task.

*Skills Levels Needed:* Level of skills needed for executing the task; excellent, fair, medium, poor skill levels.

*Task Infrastructure:* Infrastructure that would be required for the execution of the task.

*Task Priority:* Priority of the task; Low, medium, high.

*Start By Date:* The date by which one has to start the execution of the task.

*Interdependencies:* Interdependencies of the task.

*Task Supervisor:* The member / SHG in-charge of the task.

*Complete By Date:* The date by which one has to complete the execution of the task.

*Notification Recipients:* Stakeholders that are to be notified about the status of the task.

*Responsibilites:* Responsibilites that are to be undertaken by the task executor.

*Task Initiator:* Identity of the one who initiated the task.

*Potential Executors:* Executors for whom the task is meant.

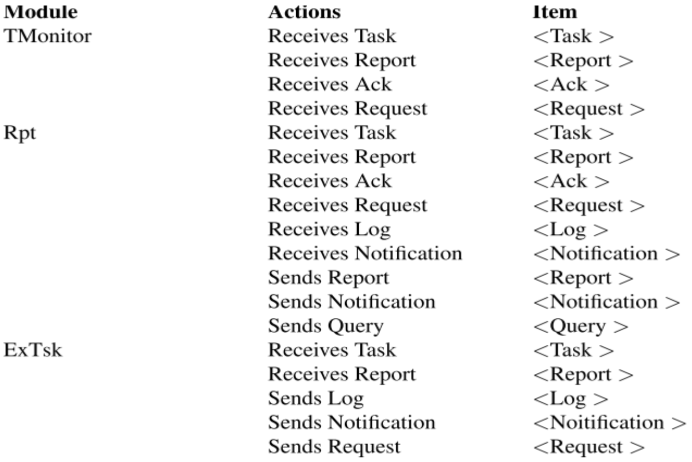
*Cost of Task:* Cost of the task estimated using variables like the resources and skills it involves and its significance (priority)

Other operations related to tasks are sending reports, ack, requests, notifications,etc. When a request is received, it has to be processed and response (report) must be sent to the requester. When a task is received, an acknowledgment must be sent to the sender. During the execution of the task, task status is to be notified to the notification recipients and requeters. And when a log or report is received, it is stored in the repository.

In MBM, each module has different functionalities.

### 3.2.2 Actions Performed by Modules of MBM

**Table 2.**



| Module | Actions | Item |
|---|---|---|
| TMonitor | Receives Task | <Task > |
| | Receives Report | <Report > |
| | Receives Ack | <Ack > |
| | Receives Request | <Request > |
| Rpt | Receives Task | <Task > |
| | Receives Report | <Report > |
| | Receives Ack | <Ack > |
| | Receives Request | <Request > |
| | Receives Log | <Log > |
| | Receives Notification | <Notification > |
| | Sends Report | <Report > |
| | Sends Notification | <Notification > |
| | Sends Query | <Query > |
| ExTsk | Receives Task | <Task > |
| | Receives Report | <Report > |
| | Sends Log | <Log > |
| | Sends Notification | <Noitification > |
| | Sends Request | <Request > |

### 3.2.3 Messages

As discussed earlier, transparency is implemented using transparency messages. These transparency messages are application specific. However, some of the transparency messages are mandatory and they are given below.

Ack
RecvdReport
ExecutingTask
AbortedTask
RecvdRequest
RecvdTask
InitiatedTask
AssignedTask
EndedTask
RecvdReport

### 3.2.4 Messages Descirption

<*Ack* >- Ack to be sent to the Repository for logging about receipt of task, request, or report.

<*RecvdTask* > - Message to be sent to the task sender. This is the acknowledgment to the sender that the task was received.

<*RecvdReport*> - Message to be sent to the report sender. This is the acknowledgment to the sender that the report was received.

<*InitiatedTask*> - Message to be sent to the task notification recipients that the given task execution is initiated.

<*ExecutingTask*> - Message to be sent to the task notification recipients that the given task execution is being executed.

<*AssignedTask*> Message to be sent to the task notification recipients that the given task execution was assigned to another member/SHG

<*AbortedTask*> Message to be sent to the task notification recipients that the given task execution is Aborted.

<*EndedTask*> Message to be sent to the task notification recipients that the given task execution has ended.

<*RecvdRequest*> Message to be sent to the requester that the request was received

<*RecvdReport*> Ack sent to the report sender that the report was received successfully.

### 3.2.5 Message Formats

<*RecvdTask*>- <RecvdTask> <task Id> from <task assigner Id> on <date> <time> <RecvdTask >

<*InitiatedTask*> - <InitiatedTask> <task Id> on <date> <time> with <responsibilities> and <interdependencies> <Initiated-Task>

<*ExecutingTask*> - <ExecutingTask > <task Id> <date> <time> <status> <problems> <ExecutingTask>

<*AssignedTask*> -<AssignedTask> <task Id> <assigned to> on <date> <time> <AssignedTask>

*<SleepTask>* - <SleepTask> <task ID> <wake up upon> <event> <SleepTask>

*<AbortedTask>* - <AbortedTask> <task Id> aborted due to <reason> on <date> <time> <AbortedTask>

<EndedTask > - <EndedTask><task Id> on <date> <time> <ended with success/failure/abort> <feedback> <performance understanding> <EndedTask>

<Ack> <Ack> <ack Id> on <date> <time> for <task Id/request Id/report Id> from <member Id> <shg Id><Ack>

*<RecvdRequest>* - <RecvdRequest> from <member Id> <shg Id> regarding <task Id> on <date> <time> <to be responded by date & time > <RecvdRequest >

*<RecvdReport>* <RecvdReport> <Report Id> on <date> <time> from <member Id> <shg Id> for <task Id > and <request Id > sent on <date > <time > <RecvdReport >

*<Log>* - <Log> <task Id> <task status> <problems> <other info> <date> <time> by <member ID> <shg Id> <Log>

### 3.2.6 Databases

During the implementation of transparency using MBM and TEC, we will be needed to maintain the following databases.

TASKS_DB,
ASSIGNED_TASKS_DB,
ABORTED_TASKS_DB,
INITIATED_TASKS_DB,
REQUESTS_DB,
TMONITOR_DB,
RPT_DB,
RECVD_TASKS_DB,
SLEEPING_TASKS_DB,
ENDED_TASKS_DB,
ACK_DB,
REPORTS_DB,
EXTSK_DB,
LOG_DB

**TASKS_ DB :** contains table which has information about every task.
*TASKS_ DB (Task Id , Task Type , Task Status , Task Priority , Task Supervisor , Task Initiator , Parent Task , Creation Date&Time , Activation Date&Time , Start By Date&Time , Complete By Date&Time , Potential Executors , IsPartitionable , Excluded Executors , Skills Needed , Skills Levels Needed , Notification Recipients , Cost of Task , Risks , Task Stakeholders , Task Infrastructure , Interdependencies , Responsibilites , Notification Recipients )*

**RECVD_TASKS_DB :** Contains information about all received tasks.
*RECVD_TASKS_DB(Task Id , Received From , On Date&Time , Start Date&Time , Completion Date&Time , Notification Recipients , Potential Executors , Infrastructure Requirements , Skill Requirements , Responsibilities , Interdependencies , Risks , Priority , Parent Task)*

**INITIATED_TASKS_DB :** Contains information about all initiated tasks.
*INITIATED_TASKS_ DB(Task Id , Initiation Time , Initiation Date , Notification Recipients , Initiating Member Id , Proposed Executors , Proposed Completion Date&Time , Skills Available For Task , Infrastructure Available , Priority Given )*

**EXECUTING_TASKS_DB:** Contains information about all executing tasks.
*EXECUTING_TASKS_DB(Task Id , Status , Problems , Notification Recipients , Members Involved , Skills Involved)*

**ASSIGNED_TASKS_DB:** Contains info about assigned Tasks
*ASSIGNED_TASKS_DB( Task Id , Assigned To , On Date&Time , Start Date&Time , Completion Date&Time , Notification Recipients , Infrastructure Requirements , Skill Requirements , Responsibilities , Interdependencies , Risks , Priority , Parent Task )*

**SLEEPING_TASKS_DB**: Contains info about sleeping Tasks
*SLEEPING TASKS DB ( Task Id , Slept On Date&Time , Waking Up Event , Woke Up Date&Time, Notification Recipients )*

**ABORTED TASKS DB:** Contains info about aborted Tasks
*ABORTED TASKS DB (Task Id , Aborted On Date&Time , Reason , Notification Recipients , Aborted By Member Id)*

**ENDED_TASKS_DB:** Contains info about ended Tasks
*ENDED TASKS DB ( Task Id , Notification Recipients , Ended On Date&Time , Feedback , Performance Understanding)*

**ACK_DB:** Contains info about acknowledgments
*ACK_DB ( Ack Id , Notification Recipients , Sent On Date&Time , Corresponding Id of Task/Request/Report , Success / Failure , RecvdSenderDetails , PreparingAck , SendingAck , AbortingAck , EndedAckSend )*

**REQUESTS_DB:** Contains info about requests.
*REQUESTS_DB ( Request Id , Notification Recipients , Sent On Date&Time , Corresponding Task , Success / Failure , PreparingRequest , SendingRequest , AbortedRequest EndedRquesting )*

**REPORTS_DB :** Contains info about reports..
*REPORTS_DB ( Report Id , Notification Recipients , Sent On Date&Time , Corresponding Task , Success / Failure , QueryingRepst GeneratingReport SendingRequest Abort/Fail EndReporting )*

**LOG_DB:** Contains info about logs.
*LOG_ DB (Log Id , Made By , Date&Time , Log Data , Notification Recipients )*

**TMONITOR_DB:** Contains info about reports..
*TMONITOR_DB( Received Item , Sender's Id , Date&Time )*

**EXTSK_DB :** Contains info about ExTsk
*EXTSK_DB(Received Item , Senders Id , Date&Time , Sending Log/ Notification/ Request Id , Date&Time)*

**RPT_DB:** Contains info about Rpt.
*RPT_DB (Received Item , Senders Id , Date&Time , Sending Report / Task / Query / Log /Notification Id Date&Time )*

### 3.2.7 Rules of the Model

1. ∀ Task t ∈ ASSIGNED_TASKS_DB ∃ t ∈ RECVD_TASKS_DB ∩ t ∈ INITIATED_TASKS_DB ∩ t ∈ SLEEPING_TASKS_DB
2. If t ∈ ENDED_TASKS_DB, then t ∈ RECVD_TASKS_DB ∩ t ∈ INITIATED_TASKS_DB ∩ t ∈ EXECUTING_TASKS_DB
3. The different possible paths and corresponding states of Task's lifecycle are:
   - Recvd → Initiate → Execute → End
   - Recvd → Initiate → Execute → Assign → Sleep → Execute → End
   - Recvd → Initiate → Execute → Abort → End
   - Recvd → Initiate → Execute → Assign → Sleep → Execute → Abort –→ End
4. If t ∈ ABOTRED_TASKS_DB, then t ∈ RECVD_TASKS_DB ∩ t ∈ INITIATED_TASKS_DB ∩ t ∈ EXECUTING_TASKS_DB ∩ t ∈ ASSIGNED_TASKS_DB ∩ t ∈ SLEEPING_TASKS_DB
5. If t ∈ SLEEPING_TASKS_DB, then t ∈ RECVD_TASKS_DB ∩ t ∈ INITIATED_TASKS_DB ∩ t ∈ EXECUTING_TASKS_DB ∩ t ∈ ASSIGNED_TASKS_DB
6. Every state has an associated transparency message.

### 3.2.8 Transparency Metric

Transparency metric quantifies the level of transparency that is maintained by the SHG or a member. Transparency metric can be computed using TEC from the number of transparency messages transmitted by the task executor. Since each transparency messages is attributed with a value, depending on the number of messages sent and the value associated with the sent messages, transparency can be computed.

## 4 Conclusion

The proposed model successfully implements Transparency as defined in this paper. Degree of Transparency can be assessed using the number of transparency messages corresponding to a task. For e.g., if a task has hundred transparency messages associated with its execution, the task initiator can define custom thresholds for opaqueness, translucency and clarity. During execution of the task, as the task executor sends the transparency messages, the count of the messages gives the degree of transparency. The dimensions of transparency are realized by defining various transparency messages in the phases of the task i.e. pre-activity, per-activity and post-activity phases. In the pre-activity phase, transparency messages related to requirements and interdependencies are defined. In the post-activity phase, transparency messages related to status and problems are defined. And in the post-activity phase, transparency messages related to feedback and performance understanding are defined. Thus the dimensions of Transparency are implemented. The perspectives of transparency are assessed based on the operational behavior of the task executor.If the task executor operates on simplex mode of communication, i.e. only sends messages but does not receive any queries from the stakeholders, then the transparency is Static transparency. And, if the task executor maintains duplex mode i.e. not only sends messages but also receives queries and responds to them, then the transparency is Dynamic transparency. Radical transparency is when the task executor streams the messages and maintains duplex mode of communication. Radical transparency is a form of static transparency with the interval between the transmission of messages being very small. Thus perspectives of transparency are implemented. Thus the proposed model successfully implements the degree, dimension and perspective of transparency as defined in this paper.

**Acknowledgments.** I would like to thank Prof. Hrushikesha Mohanty whose guidance and thoughts have become the source for my research work and life. This work is funded by Rajiv Gandhi National Fellowship.

## References

1. Andreas, Ø., Børge, L., Eva, A., Andreas, S.: A Framework for Transparency. In: 21st Annual Conference of the Production and Operation Management Society, USAID, CARE, GTZ/NABARD, Vancouver, Canada, May 07- May 10 (2010)
2. Vaccaro, A., Madsen, Æ.: Corporate dynamic transparency: the new ICT-driven ethics? Ethics Inf. Technol. 11, 113–122 (2009)
3. Tapscott, D., Ticoll, D.: The naked corporation. Free Press, New York (2003)
4. Transaction Model, Network and Information Systems Management, University of Paisley (2000)
5. Madsen, P.: Dynamic Transparency, Prudential Justice and Corporate Transformation: Becoming Socially Responsible in the Internet Age. Journal of Business Ethics 90, 639–648 (2009)
6. Vaccaro, A., Madsen, P.: Transparency in business and society: introduction to the special issue. Ethics Inf. Technol. 11, 101–103 (2009)
7. Parikh, T.: Rural Microfinance Service Delivery: Gaps. Inefficiencies and Emerging Solutions (2007)
8. Sa-Dhan: The Association of Community Development Finance Institutions, A Study Report On SHG Federation In India: Emerging Structures and Practices (July 2004)
9. Bhawan, V.: National Workshop on SHG Federations, New Delhi (May 2008)
10. Kuo, H.-C., Tsai, T.-H., Huang, J.-P.: Building a Concept Hierarchy by Hierarchical Clustering with Join/Merge Decision (2008)

11. APMAS, Optimizing SHGs (October 2005)
12. APMAS, SHG Federations in India: A Status Report (2007)
13. EDA Rural Systems and APMAS, Self-help groups in India: A Study of Lights and Shades, A joint initiative of CARE, CRS, USAID and GTZ, New Delhi (2006)
14. MYRADA, Impact of Self Help Groups (Group Processes) on the Social/Empowerment Status of Women Member In Southern India. Paper presented at the Seminar on SHG-bank Linkage Programme, New Delhi (November 2002)