CHAPTER 3

Case Studies

The improvement of understanding is for two ends: first, our own increase of knowledge; secondly, to enable us to deliver that knowledge to others.

John Locke

There are three traditional research paradigms: mathematical, scientific, and engineering. In mathematics, research is derived from constructing concepts, often in the form of formal proofs and reflexive induction and reasoning. In ' social science and scientific fields, research assumes an experimental or empirical slant. The paradigm adopted in this dissertation is the engineering approach, which typically involves studying a problem, proposing solutions, and testing the solution on real problems. Specifically, the paradigm employed is one of conceptualization, empirical exploration, and testing. The Goal-Based Requirements Analysis Method, introduced in the following chapter, was developed and evaluated while working on real problems. This contrasts with other approaches to method development in software engineering research in which methods are developed and later tested on conceptualizations formed in isolation from real applications.

This chapter discusses development of the Goal-Based Requirements Analysis Method (GBRAM) in the context of its application to real case studies. The approach taken has concurrently led to the development of an integrative goal-based requirements method and analyses of real problems. The initial case studies discussed in this chapter enabled the development of a systematic approach to goal identification and refinement, as discussed in Chapter 4; subsequent studies, discussed in Chapter 6, enabled evaluation and refinement of the method. Thus, the case studies presented in this chapter were formative, serving as the origin of the ideas and concepts presented in this thesis. The case studies discussed in Chapter 6 are summative; this distinction is key in that the summative cases previously developed methods were being validated, whereas the formative cases involved the evolution of the methods simultaneously coupled with validation. These case studies unfolded over time and GBRAM evolved as a result of its application to the case studies discussed in this chapter.

Each of these case studies involves a process or system:

- Financial Services Office (FSO) business process (Section 3.1);
- Career Track Training System (CTTS) (Section 3.2); and the
- Meeting Scheduler System (Section 3.3).

These projects are discussed in the following sections. Synopses of each project are followed by discussions of the methodology employed and the lessons learned through application of the method.

Table 3.1 summarizes the availability of the data for each of the case studies discussed in this chapter. Two of the case studies, the FSO and CTTS, are government confidential projects; thus, the raw data cannot be made available. However, the results from both of these studies have been published and are available in [7] and [4], respectively. The raw data for the meeting scheduler case study is currently available via anonymous ftp*.

Table 3.1. Availability of Case Study Data

Case Study	Raw Data	Requirements Document
Financial Services Office	*	*
Career Track Training System	*	*
Meeting Scheduler	√	√

Key:
√ Available from author upon request
* Government Confidential

3.1 Financial Services Office Case Study

The College Financial Services Office (FSO) case study focused on redesigning the FSO business process which is responsible for all of the College's finances. The FSO employs four full time employees, two part time student assistants, and requires the integral involvement of three College administrators.

This study involved the application of goal decomposition and scenario analysis in the context of Business Process Reengineering (BPR) [42, 41, 76, 78]. BPR attempts to avoid simply automating existing processes or tasks in organizations to introduce process efficiencies by questioning the reasons why specific processes and activities are linked together in

*The meeting scheduler data is available via anonymous ftp at ftp.cc.gatech.edu in /pub/groups/SERC/scenario.tar.gz.

support of a business entity [78]. Hammer and Champy observe that BPR requires "discontinuous thinking" so that dramatic performance improvements may be achieved [41]. The requirements for software systems that support reengineered processes must be understood in the context of the goals of the BPR project.

Due to the role of goals and objectives in business organizations and enterprises, the availability of financial processes in need of redesign and the necessary analysis of organizational goals made this project well suited for the development of GBRAM applicable ideas. The FSO case study exemplifies and validates the process of using scenarios in refining business process descriptions.

Methodology and Case Study Artifacts

The FSO case study was conducted for approximately 15 hours a week over a period of 3 months. Two College administrators and four FSO employees, a total of six stakeholders, participated in initial interviews for this case study. These interviews were tape recorded on site and later transcribed for future reference. The interviews served as information gathering sessions, providing an understanding of the organizational structure, the lines of communication within the organization, and the business processes for which the FSO is responsible. The initial elicitation of scenarios was unguided; each stakeholder was asked to explain the business processes for which they are responsible. Stakeholders expressed information in the form of scenarios or illustrations, often freely expressing the processes as scenarios which illustrate a process goal or demonstrate exceptional cases. The transcripts served as a source for goal and scenario identification.

The College administration initially provided a set of very high-level prescriptive goals emphasizing the deliverables for which the unit was responsible. However, upon interviewing the FSO employees, it was clear that employees in non-administrative positions outside the FSO were not aware of the existence of these prescriptive organizational goals and objectives; thus, these goals were not representative of the perceived goals of the stakeholders. The prescriptive goals in this study were analyzed using a top-down approach in order to develop a goal-hierarchy. The interview transcripts were analyzed using a bottom-up approach which also culminated in a goal hierarchy. The characteristics of the different identified goals in the bottom-up descriptive goal hierarchy and the top-down prescriptive goal hierarchy were then compared.

Lessons Learned

This section summarizes the lessons learned from the Financial Services Office case study and addresses the integration of these lessons into the Goal-Based Requirements Analysis Method.

Descriptive goals relate to operational activities

A pragmatic goal classification scheme, which differentiates between prescriptive and descriptive goals, emerged during this case study. *Prescriptive goals* are typically expressed by management-level stakeholders and account for organizational structures and processes that should be observed. Stakeholders are usually aware of goals in terms of operationalizations, expressed in the form of actions performed on a regular basis. *Descriptive goals* are found in the current operational processes of an organization. The prescriptive goals in this case study were provided by management and were codified in the organization's written procedures, whereas the stakeholders responsible for carrying out the FSO processes tended to express goals in a descriptive fashion.

GBRAM recognizes that stakeholders possess different viewpoints and that their manner of expressing those viewpoints varies as well. Scenarios elicited from stakeholders support the descriptive goals and respective operations; however, it was difficult to elicit scenarios to describe activities which support the prescriptive goals. Chapters 4 and 5 explain how GBRAM provides guidance for analysts in identifying the goals as expressed by different stakeholders.

Descriptive goals are a better source of requirements

This is the only case study in this thesis in which a distinction is made between prescriptive and descriptive goals. The interest in descriptive goals for the FSO study stems from the need to analyze descriptive goals. This is salient in that descriptive goals are more indicative of what's going on, whereas management in this study was not cognizant of the real life operational strategies. Thus, descriptive goals were helpful during this case study as a source from which functional requirements could be derived.

Scenario analysis yields concrete process goals

The scenario fragments elicited during the stakeholder interviews were reviewed to identify a concrete set of process goals. Each scenario was analyzed by asking: *"What goal does this scenario fragment either support or satisfy?* and/or *"What goal does this* *scenario fragment prevent the achievement of?"* A goal hierarchy was then constructed using the representation scheme shown in Figures 3.1 and 3.2. This bottom-up approach to goal identification focuses on stakeholder descriptions of activities and operations. It is less structured than other approaches since interview transcripts lack an organizational structure and are very much characterized by stream of consciousness; thus, transcripts do not offer a complete set of goals. Figure 3.1 shows the goal hierarchy constructed using a bottom-up approach.

These goals were defined primarily by the current operating procedures and did not offer a direct correspondence to the prescriptive goal set. However, the use of scenarios and inquiry was beneficial in that the analysis raised exceptional cases and goals which were not apparent in the prescriptive goal set.

The goals shown in Figure 3.1 were formulated upon examination of the scenarios elicited from the stakeholders and documented in the interview transcripts. For example, goal #2 in Figure 3.1 was extracted from the scenarios in NLD #4.2 on page 86. Every goal in Figure 3.1 was identified by examining the entire set of available scenario fragments; therefore, each goal has at least one supporting scenario.

The goal hierarchy in Figure 3.2 is based on the set of prescriptive goals, which were systematically decomposed into subgoals. The scenario transcripts were then reviewed in an effort to identify supporting and/or non-supporting scenario fragments such as those found in the bottom-up approach.

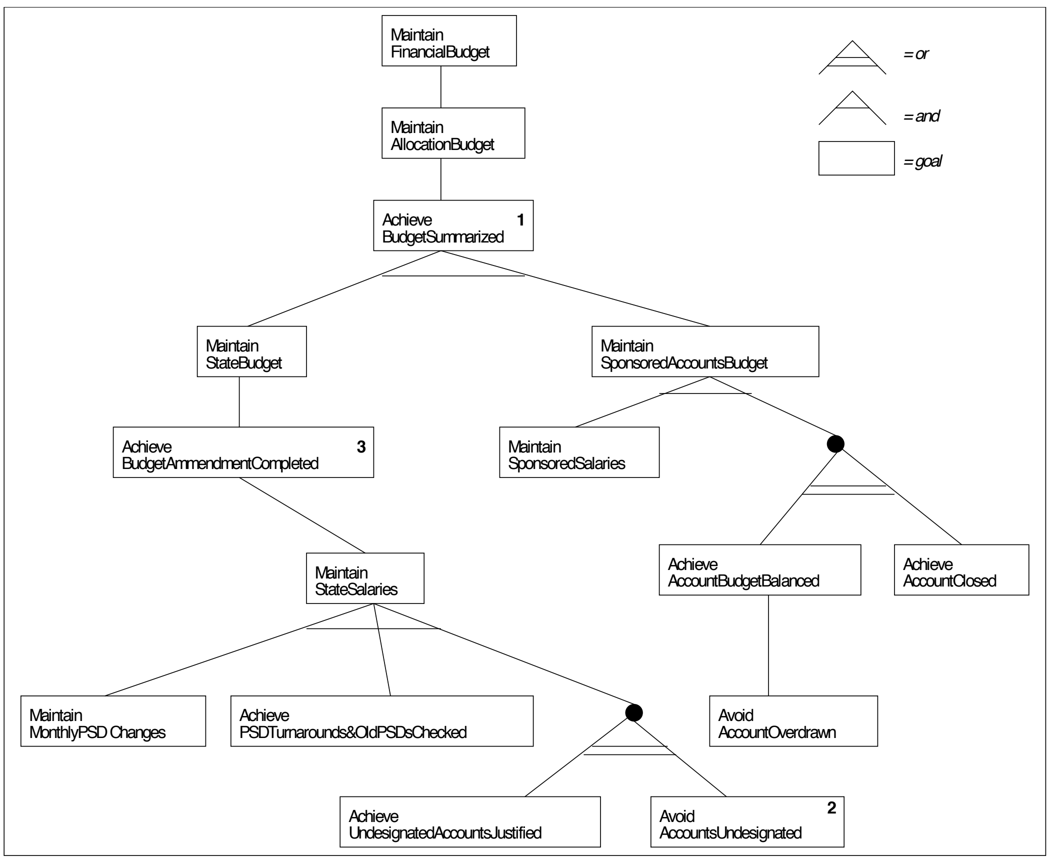

Figure 3.1. Goal Hierarchy for FSO Sponsored and State Accounts. This hierarchy was constructed using a bottom-up approach.

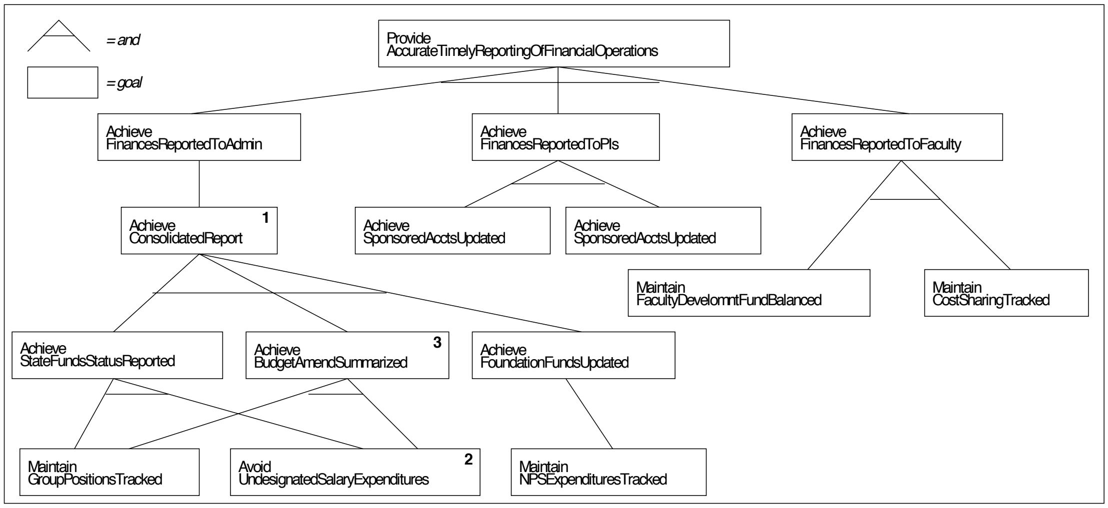

Figure 3.2. Goal Hierarchy for FSO Prescriptive Goal Set. This hierarchy was constructed using a top-down approach.

Scenarios were identified for three of the prescriptive goals: 1) `Achieve Budget Summarized`, 2) `Avoid Accounts Undesignated`, and 3) `Achieve Budget Amendment Completed`. These goals correspond to goals 1, 2, and 3 in Figure 3.1. The top-down approach requires an initial set of prescriptive goals. In the FSO, these goals displayed a definite emphasis on deliverables. Only four goals in Figure 3.2 had supporting scenarios.

Taking a bottom-up and top-down approach yields multiple viewpoints

The FSO goals were first identified by analyzing a list of process inefficiencies identified during the initial bottom-up analysis of the FSO business processes. Each process inefficiency was examined by asking *"What goal is prevented from being satisfied by this inefficiency?"* While this approach is structured and facilitated the identification of scenarios that satisfy system goals, it did not provide a complete set of goals; furthermore, the goals identified in this manner did not offer a direct correspondence to the set of prescriptive goals. Thus, the results of this analysis were not very useful for the consideration of information system requirements.

Employing only a bottom-up approach in goal identification fails to provide a sufficiently high-level view for reorganizing/restructuring the organization. Similarly, sole employment of a top-down approach prevents analysts from developing an understanding of the actual current processes in an organization. However, a top-down approach coupled with a bottom-up approach offers a more complete view of an organization and its processes. These views complement each other, resulting in the identification of multiple viewpoints which may then be resolved.

Synonym identification facilitates viewpoint resolution

Contrasting both of the goal hierarchies gives rise to issues pertaining to synonymous goals and their identification. Consider what the FSO employees refer to as the "Allocation Budget" in Figure 3.1. The College administration refers to the allocation budget as the "monthly consolidated report" as shown in Figure 3.2. Clearly, this is indicative of conflicting viewpoints. If this difference in nomenclature were resolved, it would probably be possible to identify a few more supporting scenarios for the prescriptive goals in Figure 3.2. Some of the scenarios which support the goal `Achieve Monthly Report Consolidated` in Figure 3.1 would also support the goal `Maintain Allocation Budget` in Figure 3.2.

This case study was initially motivated by the study of scenario analysis for the meeting scheduler [64]. The goal and scenario analysis of the FSO business process exemplifies the desirability of combining both a top-down and bottom-up view. The goals from the bottom-up analysis were much more concrete, process/task oriented, and descriptive in nature, while the goals from the top-down approach were prescriptive in nature. This difference led the analysis to focus on understanding the relationships between the two sets of objectives. Since scenarios played a role in refining the FSO goals, subsequent case studies sought to understand the relationship between goal refinement and scenarios.

The following section discusses an analysis of an Air Force base training acquisition process.

3.2 Career Track Training System

The Career Track Training System (CTTS) case study focused on an Air Force Base (AFB) responsible for sending employees to Air Force and Department of Defense (DoD) training. The training acquisition process was observed to be extremely fragmented, necessitating an enormous amount of time and effort involving dozens of people across numerous organizational boundaries. Approximately 155 AFB employees are enrolled in official training each year. The Goal-Based Requirements Analysis Method was employed to analyze the goals for a Career Track Training System (CTTS).

The CTTS business process was appealing and well suited for investigation of the GBRAM for several reasons. First, the requirements for the continuing education process and desired system illustrate requirements problems typically exhibited by information systems. For example, the need to register persons for training sessions in the military is analogous to the need to register students for university classes in a course registration system. Second, the problem can also be considered a resource management (and scheduling) problem since it concerns the provision of courses for employees. As such, it addresses issues which are relevant to many systems and is, thus, widely applicable. Third, due to the involvement of different units in the AFB, and the DoD, the likelihood of conflicting goals among those units is high, making it an appropriate case to consider since one of the GBRAM goal refinement approaches employs conflict identification strategies. The following section explains how the case study was performed and provides an overview of the project artifacts.

Methodology and Case Study Artifacts

Two analysts, the author of this thesis and an analyst from the AFB, conducted the CTTS case study for approximately 10 hours a week over a period of four months. One member of the team was familiar with the application domain and was also a stakeholder in the system; the other possessed a knowledge of analysis methodologies and was not a stakeholder in the system. It was observed that having both areas of expertise was essential for performing an effective analysis. Given a one page textual description of the current training acquisition process, the analyst and stakeholder conducted an analysis of the process in order to identify the system goals. The information from the analysis was reviewed, system goals were identified using inquiry and action word identification, and a goal structure was constructed using the representation scheme presented in [8]. In-depth interviews were conducted with AFB personnel and professionals in the training acquisition process to develop a comprehensive understanding of the current process. This interaction resulted in the construction of a detailed, informal flow chart model of the current process, from which an artifact of the redesigned/reengineered model with a statement of the system goals and requirements was derived. The following subsection discusses the lessons learned during this case study.

Lessons Learned

As with the previous case, the lessons learned from this case study served as an incubator for the ideas that influenced the evolution of the method. This section discusses the lessons learned during the CTTS case study and is a high-level explanation of how these lessons were integral to the formation of the method. Extensive examples from this case study are presented in the exposition of the method in Chapter 4; thus, to avoid repetition, the discussion in this section is brief.

Stakeholders express activities more readily than they express goals

Stakeholder identification allows the consideration of activities in which agents stand to gain or lose, thereby facilitating the identification of conflicting viewpoints. However, in the CTTS, stakeholders rarely gave consideration to the actual goals of a system without some form of prompting; instead, they were more likely to focus directly on operationalizations and actions which they rely on the system to perform. Since stakeholders did not volunteer goals, goals had to be elicited through prompting. This illustrates the importance of allowing stakeholders to state goals descriptively so that the analyst may synthesize these viewpoints and incorporate them into a representative set of goals. GBRAM allows analysts to share knowledge about the different stakeholders, their viewpoints, and their responsibilities so that inconsistencies may be detected. The role of stakeholder viewpoints is elaborated upon in Chapters 4 and 5.

Multiple sources yield better goals

The goals for a desired system are not always clear at the outset of analysis and must be extracted from diverse representations of information (e.g. stakeholders are not considered to be "representations" in the context of this discussion). It is unlikely that analysts will produce a complete goal set for a system given only one information source, but the combination of goals extracted from various information sources does produce a *more* complete set of goals. This is especially probable when analysis of both the current and desired systems is incorporated into the goal set.

During this case study a direct correlation was observed between the different types of goals identified and the nature of the available documentation. Three sources of information were available for this analysis: a textual introductory statement, a textual process scenario, and flow charts of the existing process. Analysis of these three sources yielded the identification of 36 goals. The introductory statement and process scenario comprise a one-page textual description detailing the high-level mandates that drive the system and organization, and the employee certification acquisition process. The flow charts represent the current process, described during interviews with stakeholders. It is important to note that the medium, or language, in which the information is expressed is less important than the actual information provided. The introductory statement for this case study is the most declarative of the three sources, while the process scenario exhibits commonalities with flow charts in its step-by-step description of the process.

The heuristics for identifying goals in GBRAM depend on the various sources with which an analyst works. Process descriptions in flow charts tend to be much more clear and succinct than are transcripts of someone speaking, since transcripts lack an organizational structure which minimizes tangential comments. These observations all played a formative role in the development of the heuristics presented in Chapter 5. Consequently, GBRAM provides guidance for analysts to identify and extract goals from information sources such as process diagrams and interview transcripts.

Categorizing goals suggests operationalizations

It is useful to differentiate among types of goals by noting the target conditions of the goals. In GBRAM, goals are classified as either achievement or maintenance goals. An *achievement goal* is satisfied when the target condition is attained. A *maintenance goal* is satisfied as long its target condition remains true*.

*The classification of achievement and maintenance goals is discussed in greater detail in Chapter 4.2.

While both achievement and maintenance goals were identified in the CTTS, maintenance goals are more likely to appear in organizational and policy level descriptions than are achievement goals since maintenance goals tend to delineate the objectives of an organization. Consider, for example, that nine maintenance goals were extracted from the CTTS prescriptive description, while none were extracted from the process description or identified from the current process descriptions. This juxtaposition supports the findings reported in [7] that organizational goals are often not reflected in operational strategies. Achievement goals are more likely to arise when exploring the process descriptions, which tend to contain more action words than prescriptive descriptions. Thus, GBRAM uses action word identification as a technique for identifying goals; the role of action words in goal identification is discussed in Chapters 4.2 and 5.2. In the CTTS study achievement goals were more readily extracted from flow charts than from, for example, the interview transcripts of the Financial Service Office study (See Section 3.1).

Given that high-level maintenance goals are not often reflected in operational strategies, it is useful to differentiate them from achievement goals. An efficient enterprise and use of information technology/systems depends on a close correspondence between the supporting systems and the enterprise goals and objectives; this thesis subdivides enterprise goals and objectives into maintenance goals, representing organizational goals, and achievement goals, representing operational strategies.

As hypothesized prior to this study, achievement goals are best mapped to actions that occur within the system, while maintenance goals tend to be nonfunctional (e.g. constraints that prevent things from occurring). Maintenance goals are helpful when operationalizing achievement goals because they can point to previously overlooked goals. In the CTTS, analysis of new relevant information providing additional descriptions of maintenance goals resulted in refinements to the set of achievement goals, thus yielding more meaningful operationalizations. By categorizing goals, analysts may also begin bridging the communication gap between stakeholders and developers. The integration of these concepts and experiences is evidenced through the discussion in Chapters 4 and 5.

Diversity of goal information gives a rich picture

System requirements information tends to be much more specific and more implementation-dependent than domain theory information. Application domain information may appear more useful for identifying the high-level goals, since the main concern is the overall problem and not a proposed solution. However, more concrete goals and goal obstacles* can be identified from system requirements, which primarily focus on actual functionality and system performance. This identification ultimately leads to a better understanding of the system due to the availability of more concrete descriptions which allow analysts to more clearly envisage the desired system. Consider, for example, the different types of goals identified in the CTTS. In the CTTS, maintenance goals are extracted from policy level descriptions and achievement goals are extracted from both policy level and process descriptions; thus, when operationalizing† an achievement goal, it is helpful to analyze the goals extracted from different sources. This analysis enables analysts to clarify their understanding of the goal they seek to operationalize so that it may be specified in more detail.

* *Goal obstacles* prevent or block the achievement of a given goal.
† *Operationalization* refers to the process of translating a goal into an operational requirement.

Constraints indicate requirements and point to new goals

Constraints provide additional information about requirements that must be met for a given goal to be completed, providing insight into issues that must be considered when goal priorities change. For example, a constraint may indicate *when* a goal can be completed. Consider a constraint which specifies that a meeting must be scheduled on a specific day. If a room is not available or no one can attend the meeting on that day, the goal priorities must be reexamined. By examining the constraints in the CTTS, new goals were identified that would otherwise have been overlooked. GBRAM provides heuristics which aid analysts in identifying and uncovering hidden goals and requirements via constraint analysis. This process is further elaborated in Sections 4.3 and 5.2.

Exceptions can be explained via goal obstacles

Goal obstacles are an effective mechanism for the anticipation of exception cases that must be handled by system operations. Some requirements arise from analysis of obstacles and are thus not obvious to stakeholders. In the CTTS, obstacle analysis forced the consideration of reasons that could prevent an agent from achieving a goal. In Chapter 4, Examples 4.14 and 4.15 on page 98 illustrate how GBRAM aids in obstacle analysis by indicating the need for the proposed system to be prepared to handle possible obstacles and goal failures. Another benefit of obstacle analysis is the ability to more easily identify scenarios* to determine why a goal could be blocked and when the relationship between goals and scenarios deserves further research. This is useful in that obstacles indicate which scenarios, if elaborated, would ensure coverage of exception cases. This relationship be-

* *Scenarios* are behavioral descriptions of a system and its environment arising from restricted situations.

tween goal obstacles and scenarios is explained in Chapter 4 and is detailed in Chapter 5.

Scenarios play a major role in uncovering issues

Scenarios offer a natural and concrete way to describe the circumstances in which a goal may fail or be blocked, facilitating the discovery of new goals and the consideration of alternative mappings from goals to operations. In the CTTS, the use of scenarios led to the uncovering of hidden goals and obstacles as well as the identification of circumstances leading to the occurrence of goal obstacles. By considering the CTTS obstacle `Submitted paperwork not reviewed`, issues were uncovered that may have otherwise been overlooked. For example, if paperwork is not complete, the employee must be notified and asked to resubmit their paperwork or risk losing their ability to improve their certification status. It was clear during the CTTS that the identification of pre- and post-conditions for each goal is important to allow each goal to be operationalized into the requirements, as discussed in Chapter 4 on page 88. Analyzing scenarios in the CTTS enabled the identification of possible postconditions for various behaviors and goals. Experiences and observations from both the meeting scheduler and the CTTS demonstrate that scenarios are useful for uncovering and elaborating requirements, checking for completeness and conflicts, and communicating with stakeholders. These concepts are addressed and supported by GBRAM, as explained in Chapter 4.

Goal Evolution

Goal evolution concerns the refinement of goals from the moment they are first identified to the moment they are translated into operational requirements for the system specification. Evolution of the CTTS goal set was marked by modifications and additions to the goals themselves. In order to examine the evolution of the CTTS goal set, the size of the goal set was tracked during each stage of the analysis. The results of the goal analysis are seen in Figure 3.3. The ovals show the number of identified instances of achievement and maintenance goals, constraints, goal obstacles, and scenarios. The alpha-numeric combinations represent the evolution of the initial goals into a more refined and elaborated set of goals.

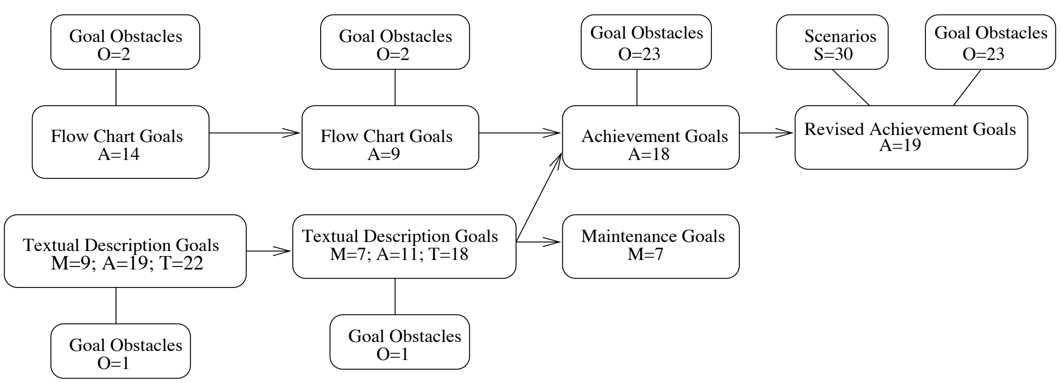

Figure 3.3. Evolution of CTTS Goal Set

Given 36 lines of text (470 words) in the CTTS textual description (Introduction and Process Scenario), 22 goals were identified. During the case study, three approaches for reducing the size of the goal set were uncovered: eliminating duplicate goals, refining goals based on system entities, and consolidating nearly synonymous goals. Using these refinement techniques, discussed in Chapter 3, the size of the goal set, derived from the textual description, was reduced from 22 to 18 goals. The flow chart goals were refined and reduced from 14 to 9 goals. Once the duplicate goals were eliminated and the synonymous goals reconciled, the goal obstacle and scenario analysis elaboration techniques were employed, yielding the identification of 23 obstacles and 30 scenarios for the achievement goals. Finally, as a result of refinements during operationalization, the size of the achievement goal set (18 goals) increased in size from 18 to 19 goals. This analysis of the evolution of the goal set facilitated early reasoning regarding the scope of effect using the refinement and elaboration techniques. More detailed discussion of these refinement and elaboration techniques is found in Chapters 4 and 5.

3.3 The Meeting Scheduler

An initial meeting scheduler case study [64] was conducted in an effort to validate the inquiry cycle model [61]; this analysis predates both the FSO and the CTTS. The meeting scheduler is a theoretical problem which may be applied to real development projects. The system resulting from the application of this problem provides a rich combination of challenging features (e.g. interfering goals, real-time aspects, and multi-agent cooperation and communication). Since the starting point for this analysis effort was a short requirements document that had to be understood, clarified, and refined, the meeting scheduler system also illustrates contractual requirements. Additionally, the meeting scheduler illustrates market-driven project issues, providing a great deal of latitude to make decisions regarding which features or components developers should implement. This study entailed an investigation of the types of questions that analysts ask about a set of written requirements and how they tend to be answered as well as the role scenarios play in the process.

The Goal-Based Requirements Analysis Method was not complete after the FSO and CTTS case studies leaving questions to be answered. At this juncture, we revisited the meeting scheduler from a different angle for observation of goal identification and refinement.

Methodology and Case Study Artifacts

The goal-based method was applied during a reevaluation of the meeting scheduler problem. Two sources of information were available for this analysis: the existing two-and-half page "Preliminary definition at the meeting scheduler system" written by Axel van Lamsweerde and his students at the Catholic University at Louvain [87], and transcripts of interviews* with three administrative assistants responsible for scheduling departmental meetings and maintaining the calendars of several professors in an academic college. Efforts were focused on two (of three) specific sections in the preliminary definition: Meeting Scheduling Domain Theory and System Requirements. Due to the preliminary nature of this study, we did not focus on the information in the third section, which pertains to system extensions. The domain theory, system requirements, and interview transcripts served as information sources for goal and scenario identification.

*These interviews were conducted by Idris Hsi, a graduate student at Georgia Tech.

Lessons Learned

This section summarizes the lessons learned from the Meeting Scheduler case study and addresses the integration of these lessons into the Goal-Based Requirements Analysis Method.

Goals may be identified using an inquiry-driven approach

The initial meeting scheduler case study included an investigation of the types of questions that analysts ask about a set of requirements [64]. In the meeting scheduler revisited, the focus was on instantiating this inquiry model for a goal-based approach. During this case study, it was evident that an inquiry-driven approach aids in goal identification.

The domain theory description explains how meetings are typically scheduled. It contains 22 lines of text which comprise the domain theory portion of the preliminary definition. Goal identification yielded 12 goals which were identified by asking *"What goal(s) does this fragment exemplify?"* The identification of these goals was relatively straightforward. Consider the following description:

> Domain Theory Description: *Meetings are typically arranged when the meeting initiator asks all potential meeting attendees for the following information based on their personal agenda:*
>
> - *a set of dates on which they cannot attend the meeting (hereafter referred to as "exclusion set")*
> - *a set of dates on which they would prefer the meeting to take place (hereafter referred to as "preference set")*

The goals G_1, G_2, G_3, and G_4, shown in Table 3.2, were identified from the Domain Theory Description. It should be noted that GBRAM does not require that goals be expressed in a specific format; throughout this thesis, goals will be displayed in a tabular format, as in this table, or in a hierarchical format, as shown in Figures 3.1 and 3.2.

Table 3.2. Goals Identified from "Scheduling Meetings: Domain Theory"

Goals	Agent
G_1: Meeting requested	Initiator
G_2: Potential attendees specified	Initiator
G_3: Exclusion set requested	Initiator
G_4: Preference set requested	Initiator
G_5: Date range specified	Initiator
G_6: Equipment requirements requested	Initiator
G_7: Location preference requested	Initiator
G_8: Exclusion set provided	Attendees
G_9: Preference set provided	Attendees
G_{10}: Equipment requirements provided	Active participant
G_{11}: Location preference provided	Important participant
G_{12}: Meeting arranged	Scheduler

Multiple sources yield a more complete set of goals

In the FSO, it was observed that stakeholders tend to describe goals in a descriptive fashion. As observed in the CTTS system, the goals that are extracted from process descriptions when used as the exclusive source of information are insufficient for achieving throughness and completeness. Therefore, the 'meeting scheduler revisited' example relies on additional transcripts of interviews conducted with three persons responsible for coordinating meetings in a university department. Table 3.3 shows the goals identified from one of the three interview transcripts. Several of the goals (e.g. G_2: `Decentralized participants coordinated` and G_7: `Overhead time reduced`) suggest areas of personal concern to individual stakeholders. In a BPR effort, such goals suggest process inefficiencies which are candidates for redesign in the new system.

Table 3.3. Goals Identified from Interview #1 (JM)

Goals	**Agent**
G_1: Meeting scheduled	Scheduler
G_2: Decentralized participants coordinated	Scheduler
G_3: Potential attendees specified	Initiator
G_4: Participant schedules requested	Initiator
G_5: Participant schedules received	Participants
G_6: Meeting requested	Initiator, Scheduler
G_7: Overhead time reduced	Scheduler
G_8: Date range specified	Scheduler
G_9: Preference set requested	Scheduler
G_{10}: Preference set provided	Attendees
G_{11}: Preference sets compiled and formatted	Scheduler
G_{12}: All preference sets consolidated	Scheduler
G_{13}: Important attendees preference sets requested	Scheduler

Goal obstacles & conflicts may be identified by using an inquiry-driven approach

By asking *"What goal(s) does this fragment obstruct or block?"* it was possible to identify goal obstacles. Table 3.4, shows that two conflicts were identified which may potentially block goal G_{12} (`Meeting Arranged`).

Table 3.4. Goal Obstacles for G_{12} from "Scheduling Meetings: Domain Theory"

Goals	Agent	Goal Obstacles
G_{12}: Meeting arranged	Scheduler	1. Date conflict 2. Room conflict

Both of these conflicts (`Date conflict` and `Room conflict`) were identified by analyzing the following description:

> Date range and conflict description: *The proposed meeting date should belong to the stated date range and to none of the exclusion sets; furthermore, it should ideally belong to as many preference sets as possible. A "date conflict" occurs when no such date can be found. A conflict is strong when no date can be found within the date range and outside all exclusion sets, but no date can be found at the intersection of all preference sets.*

For each goal obstacle (conflict) identified for the meeting scheduler, an additional table was constructed. The date conflict and room conflict tables are shown in Tables 3.5 and 3.6. Each table describes the conflict and enumerates the possible resolution strategies. In the Date conflict case (Table 3.5), four resolution strategies were specifically abstracted from the text.

Table 3.5. Date Conflict Description and Resolutions

Meeting Arranged: Date conflict
Strong Conflict: no date can be found within the date range & outside exclusion sets
Weak Conflict: dates found within the date range & outside all exclusion sets; but at intersection of all preference sets
Resolutions:
R_1: Initiator extends date range
R_2: Some participants remove some dates from their exclusion sets
R_3: Some participants withdraw from the meeting
R_4: Some participants add some new dates to their preference set

In the room conflict case (Table 3.6) two possible resolution strategies were extracted from the domain theory. The identification of such conflicts suggest candidate scenarios for analysts to elaborate. For example, consider R_2 in Table 3.5; if some participants remove dates from their exclusion set after the meeting is scheduled, this may require the meeting to be rescheduled. The inquiry approach thus aids in identifying exceptional cases which the system must be prepared to handle. Chapters 4 and 5 discuss how GBRAM supports this process and provide a catalog of questions to guide analysts through this inquiry process.

Table 3.6. Room Conflict Description and Resolutions

Meeting Arranged: Room conflict
Strong Conflict: No room available at meeting date
Strong Conflict: No room available with proper equipment meeting date
Weak Conflict: No room available at intersection of Location preference sets
Resolutions:
R_1: New round of negotiation
R_2: Active participant modifies Equipment requirements

Concepts expressed in concrete terms are easily understood by stakeholders

The most significant difference observed between the domain theory and the system requirements was the number of goal obstacles. Two potential goal conflicts (shown in Table 3.4) were identified from the domain theory, whereas sixteen goal conflicts were identified from the system requirements (shown in Table 3.7). The identification of 16 concrete goal obstacles, or conflicts, supports the theory that concepts expressed in concrete terms are more understandable to stakeholders and analysts. Although more goals and goal obstacles were identified from the system requirements, it was observed that at times it was more difficult to abstract functional goals from the system requirements by considering each statement and simply asking *"What goal(s) does this statement exemplify?"* Many of

Table 3.7. Goals Identified from "System Requirements"

Goals	Agent	Goal Obstacles
G_1: Meeting organization supported	Scheduler	
G_2: Meeting date determined	Scheduler	
G_3: Meeting location determined	Scheduler	
G_4: Meeting arranged	Scheduler	1. Participant modifies preference set(s) 2. More-important meeting called
G_5: Potential participants notified	Scheduler	
G_6: Overhead time reduced	Scheduler	
G_7: Typical meeting management reflected	Scheduler	
G_8: Conflict resolution supported according to client's resolution policies	Scheduler	
G_9: Interactions among participants supported	Scheduler	1. Participants ignore requests 2. Notification not delivered
G_{10}: Interactions among participants minimized	Scheduler	
G_{11}: Multiple meeting requests satisfied in parallel/concurrently	Scheduler	1. Meeting requests compete for overlapping time 2. Meeting requests compete for space
G_{12}: Decentralized requests satisfied	Scheduler	
G_{13}: Physical constraints maintained	Scheduler	1. Meeting room allocated to more than one meeting 2. Participant scheduled for two meetings at same time
G_{14}: Drop-dead date kept short	Initiator	
G_{15}: Participants notified soon after meeting arranged	Scheduler	1. Notification not delivered 2. Notification ignored
G_{16}: Lower bound between drop-dead date & date range specified	Initiator	
G_{17}: Privacy maintained	Scheduler	1. Non-privileged participant becomes aware of another's constraints
G_{18}: Usable by non-experts	Scheduler	
G_{19}: Customizable	Scheduler	
G_{20}: Evolving data accommodated	Scheduler	1. Participant failed to send notification of new address
G_{21}: Explicit priorities among dates in preference set satisfied	Scheduler	1. Priorities not specified
G_{22}: Participants represented by substitute	Participant	1. Delegate unavailable
G_{23}: Meeting attended in full	Participant	1. Participant arrived late 2. Participant left early

the statements exemplified goal obstacles; from those obstacles it was possible to infer the actual goals.

Conditions imposed on goals suggest constraints

Some of the statements neither exemplified a goal or obstructed a goal. Instead, they exemplified constraints which must be met in order for a goal to be achieved. For example, several constraints were identified from the 'Date Range & Conflict Description', shown on page 60 (e.g. C_3, C_4, C_5, & C_6 shown in Table 3.8). These statements are classified as constraints and not as goals. Ideally, goals are high level objectives such as 'Meeting Scheduled.' The textual fragments of the description impose constraints on the objective of scheduling a meeting. The description delineates certain requirements and conditions that must be met. It is interesting to note, however, that G_5 (`Date range specified`) in Table 3.2, which had already been identified in a previous description fragment, was identified for a second time in the Date range & conflict description. Since the goal was already specified, the second occurrence was considered redundant and was thus not added; Chapters 4 and 5 explain GBRAM heuristics that aid analysts in the identification of redundancies. The nine constraints identified for goal G_{12} (`Meeting Arranged`) are shown in Table 3.8. It should be noted that the constraints did not stem from the goals, but were instead identified from the textual descriptions in parallel to the goal identification process.

Extraordinary circumstances can be identified by considering scenarios

Scenarios facilitate the identification of special or extraordinary circumstances which occur so that goal and requirements information may be elaborated. Scenarios are identified by considering the goals and goal obstacles previously identified in order to determine the reasons why a goal may fail and the circumstances under which a goal may fail. By asking *"Why?"*, *"What are the circumstances under which this obstacle can occur?"*, *"Why did this obstacle occur?"* and *"Why was this goal not achieved?"* scenarios which address the reasons for and consequences of failure may be identified. Consider goal G_{16} (`Meeting date determined`) in Table 3.9. By asking *"What are the circumstances under which this obstacle can occur?"* four scenarios were identified for obstacle #1 (`Meeting date not determined`).

Table 3.8. Constraints for Goal G_{12}: Meeting Arranged

Constraints
C_1: Meeting date defined by pair: calendar date and time period
C_2: Exclusion sets are all contained in the date range
C_3: Preference sets are all contained in the date range
C_4: Proposed meeting date belongs to date range
C_5: Proposed meeting within no exclusion set
C_6: Proposed meeting belongs to as many preference sets as possible
C_7: Meeting room available at Meeting date
C_8: Meeting room meets Equipment requirements
C_9: Meeting room belongs to as many as possible Important Participants' Location Preferences

Table 3.9. Scenarios for Meeting Scheduler Goal G_{16}

Goal	**Goal Obstacles**	**Scenarios**
G_{16}: Meeting date determined	1. Meeting date not determined	1.a Date conflict
	2. Preference sets not consolidate	1.b Room conflict
	3. Preference sets not provided	1.c Equipment not available
		1.d Important participant pref set not provided

Results

Goals were also extracted from the problem definition document produced by Axel van Lamsweerde et. al. [87] which is referred to as the "System Requirements." The textual statements of need provided in the System Requirements were of a different flavor than those provided in the domain theory. The domain theory information is more widely applicable in that it is much more generalizable than the system requirements information. The system requirements information was much more specific and more implementation dependent. The argument may be made that domain theory information is more useful for identifying the high-level goals for the system due to its concern with the overall problem; in counterpoint, it may be said that more concrete goals and goal obstacles were identified from the system requirements, ultimately leading to a better understanding of the system. This may be due to the System Requirements' foundation on principles of actual functionality and system performance, which leads to more concrete descriptions and allows the system to be more clearly envisaged. However, it may also be the case that more goals and goal obstacles were identified from the system requirements simply due to the difference in length (52 lines of text versus 22). Table 3.7 shows the 23 goals identified from the system requirements.

One important issue in the meeting scheduler problem is the notion and occurrence of conflicts. From a general perspective, the concept of conflicts is specific to the domain of meetings and is not particular to the ontology of goals. However, in the domain of scheduling meetings, conflicts are analogous to goal obstructions in the general ontology of goals.

Chapters 4 and 5 demonstrate how the lessons learned from the meeting scheduler revisited were integrated into the Goal-Based Requirements Analysis Method.

3.4 Summary

This chapter presented the three case studies which served as the conceptual origin for the Goal-Based Requirements Analysis Method. Each case study detailed in this chapter involved a particular system:

- the Financial Services Office;
- Career Track Training System; and
- the Meeting Scheduler.

In the Financial Services Office process, goal decomposition and scenario analysis were investigated in the context of business process reengineering. The Career Track Training System (CTTS) required the reengineering of business processes spanning several inter-organization boundaries within a large enterprise. The meeting scheduler case required an analysis of user needs for a multi-user office application.

A synopsis of each project detailed in this chapter was followed by a discussion of the methodology and the lessons learned. The case studies served as a source of early validation, shaping the GBRAM through the lessons learned. The next chapter introduces the Goal-Based Requirements Analysis Method in detail.