Goal Sketching from a Concise Business Case

Kenneth Boness
School of Systems Engineering
University of Reading,
Berkshire, RG6 6AY, UK
k.d.boness@reading.ac.uk

Rachel Harrison
Dept of Computing & Electronics
Oxford Brookes University
Wheatley, Oxford OX33 1HX, UK
rachel.harrison@brookes.ac.uk

Abstract - **This paper describes how the business case can be characterized and used to quickly make an initial and structurally complete goal-responsibility model. This eases the task of bringing disciplined support to key decision makers in a development project in such a way that it can be instantiated quickly and thereafter support all key decisions. This process also greatly improves the understanding shared by the key decision makers and helps to identify and manage load-bearing assumptions. Recent research has revealed two interesting issues, which are highlighted in this paper.**

Keywords-goal-oriented requirements engineering; project management; agile development.

I. INTRODUCTION

This paper amplifies work originally presented in [1]. Our aim with goal sketching is to help stakeholders who have to make critical decisions in projects that develop evolving systems. We are developing goal sketching through action research to provide an agile way of maintaining a coherent representation of what is known about what the project is to do and how it is to do it. In [2] we state four key objectives to help the decision makers set and manage stakeholders' expectations and nurture shared understanding. Two of these objectives are bringing this help to bear as close as possible to the beginning of the project and ensuring that the methods can be adopted easily by project managers as well as analysts.

In [3; 4] we reported on case studies that suggest that building the stepwise goal refinement arguments that are fundamental to goal sketching can be more difficult than the simplicity of the concept would suggest. In [4] we showed that stepwise refinement of functional goals can be accelerated using simplified activity diagrams provided that due attention is paid to the environment (for example the contexts of construction, commissioning and operation) [2].

Reflecting on our own use of goal sketching in real projects and observing undergraduate students' difficulties it became clear that creating an initial goal sketch, which is useful can also be difficult. In this paper we report on new work to overcome this difficulty. This paper consolidates the material reported in [1] and extends it in the light of further experience.

The basic idea is to determine the roots of the goal sketch from the project's business case. Our new technique takes that part of the business case, which is crucial to the existence of the project and casts it into a structured format of goal oriented propositions (GOPs) that we call a *concise business case*. As our goal sketching is entirely based on GOPs and their refinement [2] the concise business case thus provides a disciplined start to the process. The goal sketch initiated in this way is immediately turned into a structurally complete goal-responsibility (G-R) model by adding suitable additional GOPs; usually with a generous quantity of assumptions. This process quickly brings load-bearing assumptions [5] and constraints to the fore thereby quickly framing a picture of what is known about the requirements and a clear understanding of the current threats jeopardizing the satisfaction of the business case. The aim is to help managers and developers recognize where the project might safely proceed, where it would be prudent to invest more resources into analysis, which assumptions should be "hedged" (mitigated against) and/or "sign-posted" (flagged as early warnings) [5].

It should be noted that there is no presumption in our technique that a single immutable business case is created at the outset. It is simply asserted that the purpose of the project is to deliver products that satisfy the business case at the time.

In standard product based planning (PBP) as espoused in [6; 7] the scope of a project is defined by the sum of its specialist products. Thus with the inclusion of 'management' products (project plans, contracts etc) all the expected contributory effort to a project can be estimated; at least in principle. However this is only true in practice when what is to be done and how it is to be done are both clear (such as the "painting by numbers projects" described in [8]). But when the situation harbors considerable uncertainties about what and how it is then said to be "in the fog" [8]. Setting realistic stakeholder expectations (including the eventual satisfaction of the business case) is then problematical and would need the investigative methods of requirements analysis to discover the *what* while technical invention may be needed to accomplish the *how*.

The methods in this paper concentrate on projects that have invention and/or discovery as prerequisites to their conclusion. In terms of the classifications in [8] they are the projects with a preponderance of "quest" (clear *what* and unclear *how*), "movie" (clear knowledge of *how* but unclear *what*) or "in the fog" (unclear about *what* and *how*). These situations are typical of, but not limited to, projects where Agile methodologies apply. In the wider project management community they can be recognized as *soft* projects [9].

This paper proceeds with Section II putting the work into the context of related work. Section III outlines goal-responsibility (G-R) models as used in goal sketching. Section IV introduces the concise business case template in a simple form that is then expressed as a goal-sketch in Section V. Section VI introduces two refinements of the simple idea. Experiences from a pilot study with students and real-world case studies/y are reported in Section VII. Conclusions are presented in Section VIII.

II. RELATED WORK

The G-R modeling used in goal sketching has an antecedent in KAOS [10], which itself has applications in business process modeling as well as requirements analysis such as illustrated in [11], An alternative goal oriented approach I* [12]has been applied to a wide range of requirements and business process re-engineering. In [13]the authors combine I* with problem frames [14]and business processing to model business strategy with goal oriented analysis. Use-case techniques of goal oriented requirements engineering can offer considerable agility especially when applied with the breadth before depth pattern [15]. However use-cases primarily concern the functional behavior and outcome guarantees [16] and even when the 'wheel and hub' [16] is accounted for many more project concerns remain to be managed. None of these approaches has been specialized for projects and their business cases.

The methods of project management emphasize product delivery, risk and the *raison d'etre* provided by the business case. This is perhaps best exemplified in PRINCE2 and its product based planning [6]. However there appear to be no techniques to assure that the products will actually satisfy the business case; especially when the business case includes a requirement to satisfice the concerns of a complex customer community.

We accept that the above approaches offer potential rigor and precision in their specialized ways but none provide a combined model of business and technical requirements analysis that meets the agile aims of goal sketching [2]; speed and a concise capture of rationale in just enough precision for managing expectations and enriching stakeholder negotiation. Our new technique of goal sketching from the business case complements, and can be used with, the appropriate best practice requirements engineering and project management techniques.

Pursuit of alignment and shared understanding leads to complexities known as “problematical situations” in soft systems methodology (SSM) [17] Therefore soft systems thinking could be used for investigating soft projects. However this would tend to be cumbersome in the face of demand for agility. Nevertheless the attention to *weltanschauungen* (world views) and human activity systems (*holons*) in SSM [17] has informed our approach to goal sketching. Our goal sketching may be related to the nesting of *holons* in SSM but with the simplification of a focus on project appraisal as a problematical situation. We are currently exploring this relationship more deeply.

III. GOAL-RESPONSIBILITY MODELS

Goal sketching is a lightweight technique for producing goal-responsibility models. An example goal-responsibility model is shown in Figure 1. The figure and the explanation provided here is abstracted from [2].Each box in Figure 1 is a 'goal oriented proposition' (GOP). There are goals, assumptions and constraints. In this example P is satisfied by the combined soundness of A,Q and R. R is satisfied by actors 1 and 2 taking necessary joint and collaborative responsibilities. Similarly Goal Q is satisfied by C,S and T where S and T are satisfied by actors 1 and 3 respectively. C is a constraint that will be satisfied by the definition of a 'rule' for cross-cutting the responsibilities of actors. [2].

In this example P is a single root and A,C,S,T and R are the leaves of the G-R model. Note that the necessary behavior (and other qualities) that must be instantiated is described only at the leaves of the model; it is not distributed across the model.

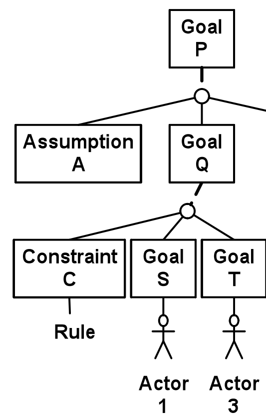

Figure 1. Goal-Responsibility model.

A *structurally complete* G-R model is one (such as Figure 1) where: all goal leaves are guaranteed by responsible actors and constraints are guaranteed through cross-cutting rules. The only leaves not guaranteed are assumptions, that must be trusted.

The skill of the analyst is to organize the GOPs into a structurally complete and persuasive stepwise argument. This discipline is a powerful aid to understanding what is known about the requirements and their preferred satisfaction. It allows the analyst to guide the setting of expectations among the stakeholders. For example in order to achieve structural completeness the analyst may need to add GOPs as “TBD” (to be determined) or to add one or more assumptions. These moves may reveal a lack of information as well as vulnerable assumptions and thus point to the risks surrounding expectations on the current understanding.

IV. THE CONCISE BUSINESS CASE (CBC)

Major project management methodologies emphasize the temporary nature of projects and how a project’s continued existence can be justified by a viable business case; see [6; 18; 19]. Taking PRINCE2 as an exemplar, a project is defined as:-

"a temporary organization that is created for the purpose of delivering one or more business products according to a specified business case." [6].

Business products (aka specialist products [6]) define the intended outcome of the project. A product may be all or part of what Alexander calls *kit* [20] or an accomplishment such as completing the training for a group of staff who will be served by the kit. At the heart of the above definition is the imperative that these products satisfy a business case. It follows that the requirements, or acceptance criteria, for the products should be traceable to the business case. One way of assuring this is to create a G-R model such as Figure 1 to represent the business case roots, constraints and assumptions and after suitable stepwise GOP refinement placing the products that are to participate in the live system among the actors.

A business case typically includes a promissory part and a rationale justifying the investment needed to accomplish it. The promissory part will include benefits that are both direct (i.e. immediate) and indirect (realized later). The project is obliged to deliver only the direct benefits. Under the definition (above) it is the promissory part of the business case that concerns the project. Bearing this in mind and considering projects that we have observed (system & product development projects and investigation projects) we have postulated certain characteristics and summarize them in what we call the *concise business case template,* as described below.

CBC Template: Subject to the validity of certain <u>assumptions</u> it is agreed by the project owners that it is a sound investment proposition to realize certain direct <u>benefits</u> and enable other indirect benefits to the project owners through the development of products that will <u>satisfice the concerns</u> of a given community of 'customers'. This is to be accomplished within defined <u>constraints</u> on time, cost and prescribed <u>approach</u>.

The terms owner and customer are adopted from [17]. The term owner thus stands for those people, or their representatives, sponsoring the project so long as they can expect a satisfactory return on their investment. The term customer stands for someone (or agency) that will be a beneficiary or potential victim of the results of the project fulfilling its obligations. The community of such customers includes all the 'on-stage' and 'off-stage' actors [16] such as users and regulators.

The underlined text in the CBC template affords a basis for structuring the promissory part of the business case as a set of goal oriented propositions; the motivations (/m/), behaviors (/b/), constraints (/c/) and assumptions (/a/) described in [2]. This may be more easily visualized through a Goal Frame [2] as shown in Figure 2.

In Figure 2 the large box represents the target domain of the project, which here (and according to the CBC template) contains two sub-domains: the products to be produced and the customer community. Usually in practice both of these domains are decomposed into their own sub domains.

The underlined terms in the CBC template are represented in Figure 2 as follows: The <u>benefits</u> appear mostly as motivation goals at the top of the frame but there may also be motivations involved in satisficing [21] the concerns of customer community. The <u>assumptions</u> appear mostly as load-bearing assumptions (holding up the frame at the bottom) but there may be further assumptions involved.

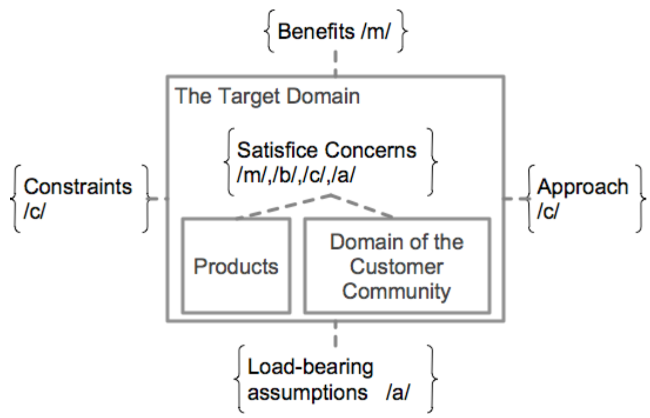

Figure 2. Concise Business Case as a Goal Frame.

The <u>constraints</u> and <u>approach</u> appear mostly as the constraints on each side of the frame (containing it) but again there may be further constraints emerging through the concerns of the customer community.

A simple illustration adapted from the zoo turnstile example in [22] serves to demonstrate the above ideas: The sponsor is the management of a zoo who believe that it is worth the investment to develop a computer-controlled turnstile guarding the entrance to their zoo. Their concerns therefore relate to an application domain involving the public and their interaction with the zoo. The GOPs in the business case could be those shown in Figure 3.

Assumptions:
- Admission to the zoo is through one gate alone. /a1/
- *Revenue is being lost by visitors evading payment. /a2/*

Benefits:
- *Increased profit for the Zoo /m1/*
- *Control of admissions /m2/*

Satisfice Customer Community:
- *Safety of the visitors (Emergency services) /m3/*
- *No additional workload (Staff)./m4/*
- *Easier reporting of visitor statistics (Staff) /m5/*

Defined Constraints:
- *The new system shall be operational by 1st April 2009. /c1/*

- *The development resources are X. /c2/*

Approach:

- *Develop a computer-controlled turnstile guarding the entrance to the zoo. /c3/*

Figure 3. The GOPs for the Zoo project.

The approach presented in Figure 3 is what we refer to as the *simple* form of the CBC. Experience has shown that certain additional concerns may need attention. These are introduced in section V following an illustration.

V. GOAL-RESPONSIBILITY MODEL FOR THE BUSINESS CASE

A structurally complete goal refinement model for the concise business case template is shown in Figure 4. Because of the lack of detail provided the completeness of the G-R model depends, as anticipated in Section III, on added assumptions and TBDs.

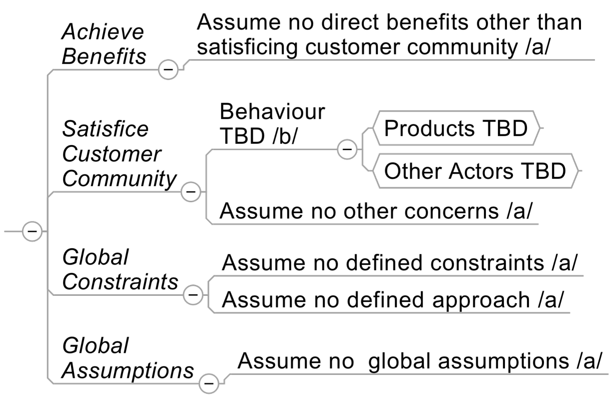

Figure 4. Structurally complete Goal-Responsibility Model for the Concise Business Case.

The goal responsibility model in Figure 4 reads from left to right. The nodes without type indicators (such as 'global constraints') are inserted as grouping nodes to make the reading easier.

In order to make Figure 4 structurally complete a set of assumptions were added to the effect that there are no other known concerns at each level of refinement over and above the concerns explicitly addressed. Such assumptions can provide a powerful challenge to the stakeholders and this helps the elicitation of technical and project requirements. A single behavior goal (/b/) has been added as a place marker and is yet to be determined (TBD) in detail. This behavior must be terminated with defined responsibilities to guarantee it.

If the project board trust the assumptions in Figure 4 and believe that its TBD can be safely resolved some time later they may judge that enough analysis has been completed; there is enough precision where it is needed and the assumptions are sound. However it would be difficult on the basis of Figure 4 as it stands to have any confidence in setting the stakeholders' realistic expectations. It is more likely that further analysis would be undertaken to validate or replace the assumptions and clarify the TBD. Completing a structurally complete G-R model with just enough detail and precision to satisfy the project board is an iterative process.

Returning to the turnstile example, the GOPs of Figure 3 are laid out in the G-R model Figure 5, which owing to space limitations is not shown in structurally complete form.

In Figure 5 the assumptions are satisfactory from a structural completeness point of view; though they are probably not persuasive.

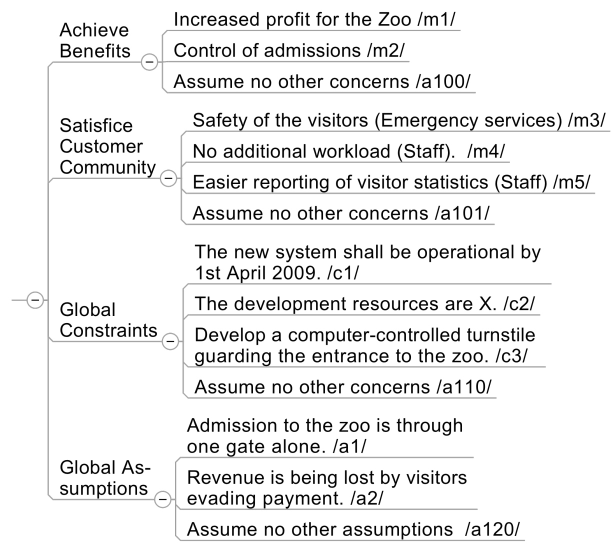

Figure 5. Structurally incomplete G-R model for the Zoo Turnstile project.

A rationale for enforcing the constraints needs to be added. They will be handled differently: /c1/ and /c2/ cross cut the project plan and impact on the feasibility of the production of project products (see [2; 23] and Cockburn's 'wheel and hub' [16]); /c3/ is a design constraint that would be testable in any products developed by the project. The motivation goals /m1/ through /m5/ will require refinement into behaviors guaranteed by appropriate actors; as in Figure 4 some of the actors will be those of the application domain and some will be the products.

As a temporary measure Figure 5 could be made structurally complete by adding TBD behaviors and suitable assumptions. As an example, a speculative first analysis is provided in Figure 6 for /m2/.

The behavior /b1/ in Figure 6 is described by a use case, which is indicated as TBD. If the project board are content that this can safely be left to the future or to chance in the hands of the developers then no more precision is needed even though the actual project products that will provide machines or props are also TBD.

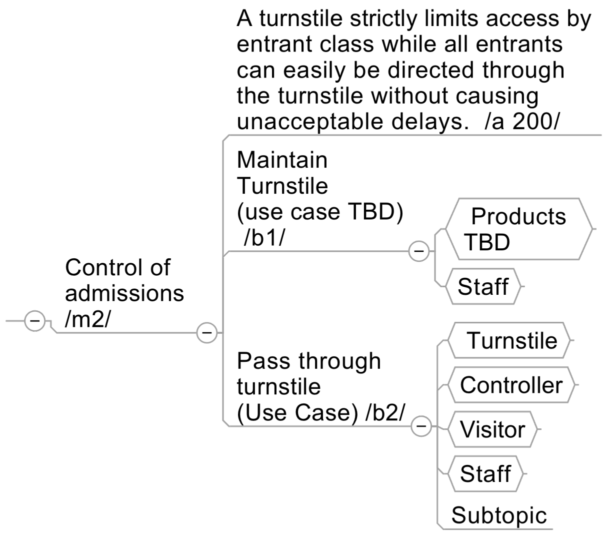

Figure 6. Structurally complete refinement to /m2/.

The goal /b2/ has definite assigned responsible actors in the form of two products (Turnstile and Controller) and people. Again if the project board is satisfied with its use case then no more analysis is needed on this matter and attention can be directed to the outstanding /m2/ through /m5/ and the composition rules for /c3/.

VI. Two Practical Issues

Practical experience has drawn our attention to two issues. The first of these relates to the context of project instantiation and execution.

A. *Direct and Indirect Benefits*

Turner [24] points out that (as mentioned earlier) projects are temporary organizations within a regime of project based management. Projects deliver products but the benefits sought might depend upon their subsequent exploitation (see [24] figure 1). This implies that the 'benefits' in the CBC may be indirect pending their exploitation. Thus entries listed in the CBC (such as benefits and assumptions) may need to be adjusted in order that the CBC expresses the promissory obligations on the project alone. If this observation is overlooked it can be the cause of confusion by practitioners of our method; this was found in the experience reported below.

To help circumvent the confusion we strongly advocate that the CBC is reviewed and adjusted to ensure that it expresses the project viewpoint and differentiates between direct and indirect expectations. The adjustments tend to be that those benefits that are indirect are commuted to assumptions.

B. *Efficiency*

The second concern pertains to the efficiency of goal sketching from the CBC.

Refining the CBC towards concrete responsibilities reveals a set of project acceptance criteria based on constraints, behaviors and assumptions. In simple examples such a refinement offers obvious 'stems' from which systems requirements can be expressed as refinement trees. For example Figure 6 takes /m2/ as a stem from which the turnstile system requirements can be obtained by refinement. This was assumed in [1] and has been assumed in all sections above.

It is more often the case that the branch from which a system refinement would logically follow will not be obvious. We have recently adjusted our technique to overcome this difficulty. Two goal graph trees are now used. The first representing the project CBC and the second representing a presumed goal of 'sound solution architecture'.

The 'sound solution architecture' goal provides a systematic opportunity to represent the full scope of the required products and the full criteria by which they would be acceptable. This point is easily overlooked. Direct refinement of the CBC can be expected to provide the ultimate acceptance criteria. However in a systems or software development project these are satisfied through the construction of a new structure. The project owners might not be able to verify the structure and it might not be their primary concern so long as the direct CBC derived acceptance criteria are satisfied. However to the project the structure is the means to satisfying the CBC goals but will itself need to be a complete and coherent structure that must be acceptance tested. It is as if the CBC presents 'black box' acceptance criteria and the second tree offers 'white box' acceptance criteria. For a project to be successful both must be satisfied. Separating problem (CBC) and solution (the solution architecture) in this way is a manifestation of the well-known fact that requirements and design so often intertwine [25; 26] as do problem and solution [27].

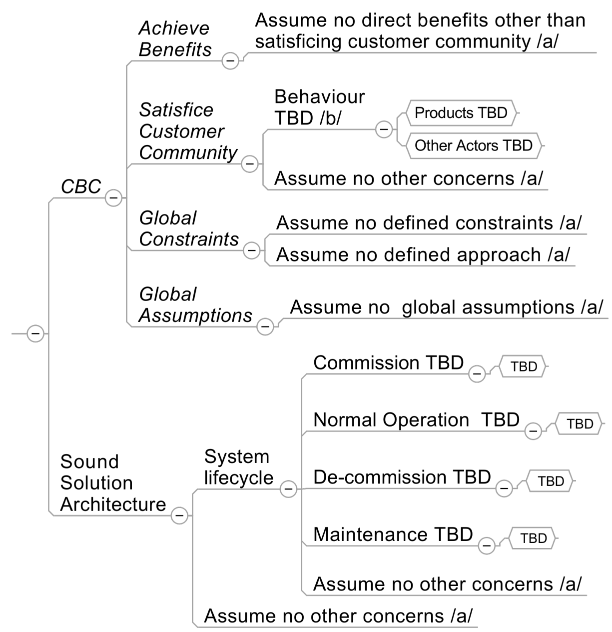

Figure 7. Showing the introduction of the sound solution architecture.

The principles are illustrated in Figure 7 where the original CBC of Figure 4 is supplemented with a 'sound solution architecture' goal.

The sound solution architecture is refined into system lifecycle concerns and an assumption of no others. In practice this assumption is unlikely to hold as there often are other parts of the solution architecture such as enterprise architecture phasing and personnel training. All of these entail products to be supplied by the project. However for the present audience the main interest will be the system lifecycle. Here we have assumed a lifecycle pattern that we often find. There is a main system with its normal operation and system actors to support that operation (all TDB at this stage as indicated by the TBD). There are also systems for commissioning and maintenance. In general there will be cross-cutting acceptance responsibilities between the actors of the different systems and between the systems and the CBC criteria.

VII. Case Study

The example application is taken from a project undertaken by a small software product development company supplying tools for use in the UK medical primary care sector. The example is generalized to illustrate the fact that it has already become a reusable analysis pattern [28] to the company. This company is referred to as the supplier in the following example where company and organization names have been changed to safeguard confidentiality.

Example: A Pharmaceuticals company (PCo) wants to provide a software tool that can be installed and used in general medical practices in the UK as a supplement to their usual medical systems. The tool is intended to access and analyze the electronic records for patients registered with the medical practices who have a particular condition (the cohort of interest in each medical practice). The analysis will show compliance and deviations with nominated best practice care guidelines published by a College of Physicians (CP) and will provide data to be analyzed in a research department at the University of X (UoX) supporting the guidelines. It is a part of the business justification that this will obtain the endorsement of the National Society for the Condition (NSC). A hidden justification is that such acts of educational contribution improve the standing of the PCo among the healthcare professionals. A pharmaceutical industry regulatory body (RB) gives strict rules that the PCo must obey when interacting with the practices and the National Health Service (NHS) regulates codes of confidentiality in regard to the access and use of patients' data.

The sponsor is thus the PCo and the Customer Community includes the doctors, the CP, the Regulators (RB and NHS). The application domain comprises the medical surgeries with their staff and standard medical computer systems. The complete Customer Community Domain (see Figure 2) is shown in Figure 8. Each of the sub domains harbors people with concerns that will be satisfied by the project's products alone and/or in collaborations with actors from the sub domains.

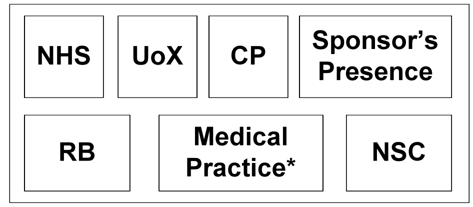

Figure 8. The Customer Community Domain for the PCo project.
*(Note: the * indicates multiplicity)*

The sponsor requires the supplier to provide the tool but, crucially, is not acquainted with the normal working activities associated with a medical practice nor with the different medical computer systems in use. The sponsor relies on the supplier for this knowledge. Thus sponsor's business case requires the satisficing of a customer community's concerns that are not appreciated by the sponsor. Figure 9 shows the CBC.

Benefits:
- *Enhance the PCo's standing appropriately with healthcare professionals /m1/*
- *Contribute to the evidence base for the Guidelines /m2/*

Satisfice Customer Community:
- *Satisfy the regulators concerns. /m3/*
- *Provide a practical service to help the doctors manage the care for their patients in the cohort of interest. /m4/*
- *Collect suitable data for onward supply to the Guidelines research centre./m5/*
- *Satisfy all brand and commercial presentation concerns /m6/*

Global Constraints:
- The tool shall be operational by 1^{st} April 2009. /c1/
- A fixed price development fee of £X. /c2/

Approach:
- Develop an independent software tool that can be worked cooperatively with standard medical computer systems /c3/

Global Assumptions:
- *The best practice guidelines would be adopted more rigorously in the medical centres if they could be made more accessible. /a1/*
- *Support for the guidelines is note provided as a part of the normal behavior of the standard medical systems. /a2/*
- The supplier knows how to satisfice the normal working needs of the intended users /a3/

Figure 9. The GOPs for the PCo project.

An initial structurally complete G-R model was constructed from Figure 8. Six low precision TBD and assumption GOPs were needed to establish the initial structural completeness. In general the assumptions and

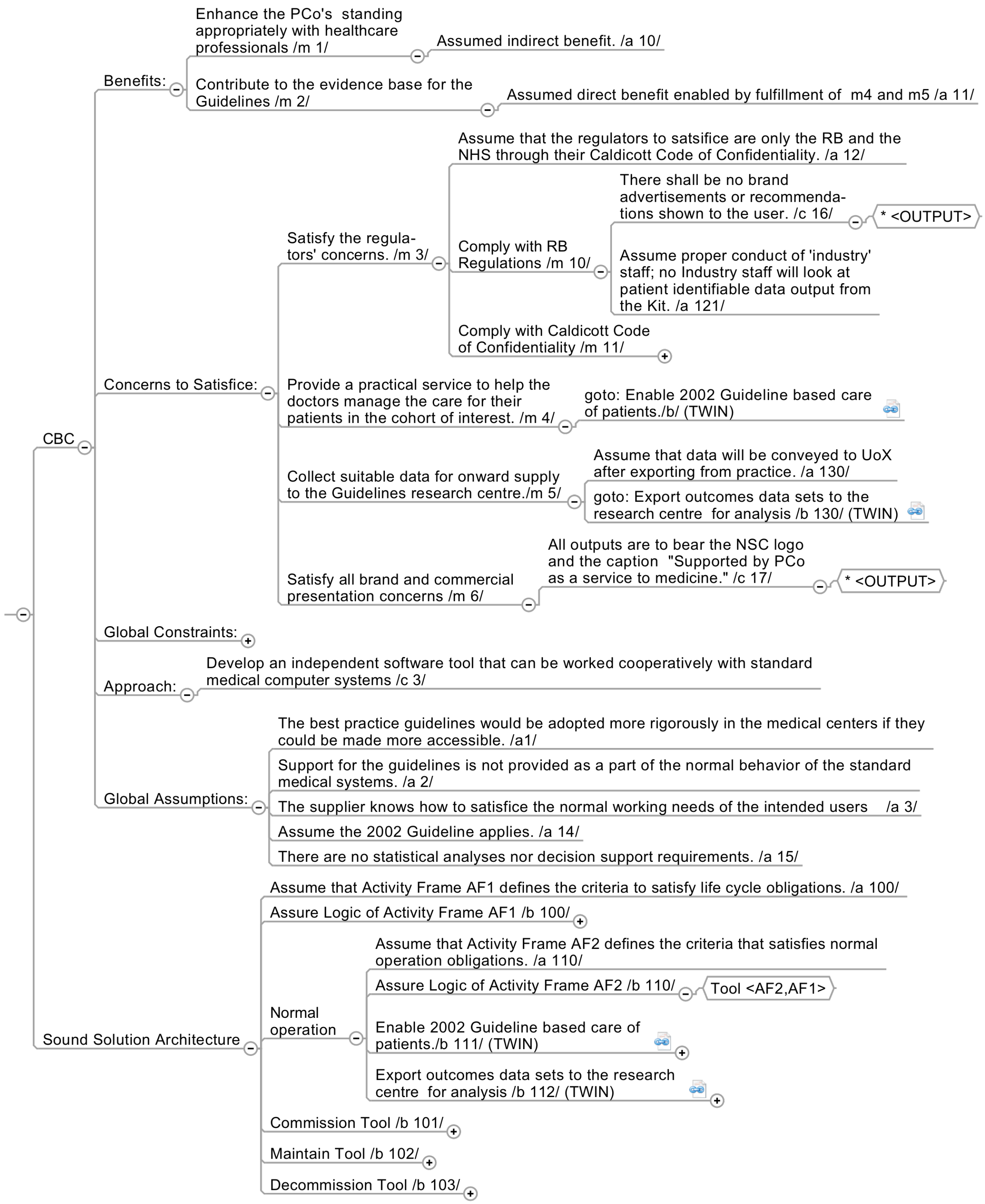

Figure 10. Partially expanded G-R model for the CBC in Figure 9.

TBDs could not be accepted by the stakeholders; however after a couple of cycles of iteration involving discussion and the goal sketching techniques outlined in [2] just enough precision was established to make proceeding on some parts of the development acceptable to the stakeholders (e.g. the refinements of /m3/,/m5/ and /m6/) whilst other parts (e.g. the refinement of /m4/) needed to be analyzed further before proceeding. The developed G-R model is shown in Figure 10.

The figure is only partially expanded because of space limitations. The (+) marked at leaves indicates hidden ('rolled-up') detail. If all of the (+) are expanded the reader would see that the graph is indeed structurally complete. All of the elements of Figure 9 are transcribed onto the graph. Benefit /m1/ is an example indirect benefit as it depends on exploitation and effect outside the competence of the project itself. On the other hand /m2/ is accepted as a direct goal, which is assumed to be satisfied as a consequence of satisfying the two customer community GOPs /m4/ and /m5/. These two GOPs imply complex requirements. These requirements require the vehicle of a coherent and structurally complete system to support their satisfaction. Such a system must be self-consistent and complete in its own terms hence it is more appropriate to apply the 'sound solution architecture' method described in section VI.II above. The /m4/ and /m5/ branches therefore point to /b111/ and /b112/ in the normal operation branch of the sound solution architecture GOP. (The link between pairs such as /m4/ and /b111/ is indicated by a hyperlink icon and the supplementary test '(TWIN)'.)

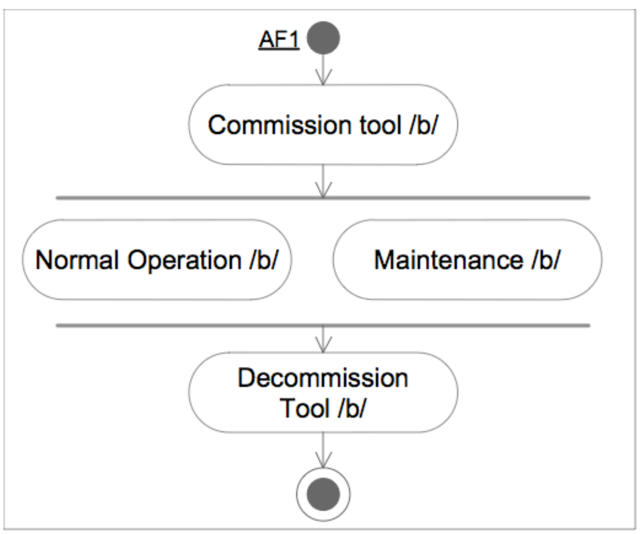

Figure 11. Activity Frame AF1 referenced in Figure 10
It depicts a life-cycle pattern.

The sound solution architecture GOP has been refined using a life-cycle refinement pattern that we frequently find appropriate. The pattern is shown in Figure 11. It has the form of an activity frame and has been applied as a refinement device in the manner we describe in [2; 4]. By applying this pattern (or an alternative) we can trust that the additional 'systems' are not overlooked.

The activity frame of Figure 11 appears in Figure 10 as AF1 in /a 100/ and /b 100/. Another, more complex activity frame has been used called AF2, which appears in /a110/ and /b111/. We refer the reader to [2; 4] for details of the technique of working with activity frames and goal refinements.

Some of the leaves of Figure 10 show operationalization by responsibility assignment. /c 16/ represents a constraint on all (signified by *) system elements that involve outputs to the screen, file or paper. The semantic tag <OUTPUT> indicates cross-cutting to all other responsibilities marked also with the tag (see [2] for details of the method). Similarly the system element 'Tool' in /b110/ will cross-cut all others with tags <AF1> and <AF2>.

This example and others have led us to the following interesting observations:-

1 Although the technique was established to accelerate goal sketching on new problems this approach has (as mentioned above) become the standard pattern used by the company on its development projects.
2 The resulting G-R models appear to focus from the start on the assumptions that are load-bearing and vulnerable [5] and this can readily lead to assumption based planning [5] with its recommended 'hedging' and 'sign-posting' tactics.
3 As anticipated when discussing Figure 4, the assumption 'no known further assumptions' and the need to impose provisional TBDs provoked keen attention to the assumptions and consequently increases the understanding shared by the stakeholders.

The effectiveness of this approach is difficult to quantify as it is uneconomic to execute a project twice concurrently; one for control and the other for comparison. Nevertheless looking at three projects (of similar size and complexity) where we used goal sketching prior to the incorporation of the CBC it can be said with some confidence that there was an appreciable acceleration brought about by the use of the CBC. Qualitatively this appears to be due to increased confidence brought about by the focus of the CBC. There were also benefits due to achieving higher that usual early shared understanding.

Pilot study: At the University of Reading each final year undergraduate of the School of Systems Engineering has an individually assigned project. 44 such students who attended an optional short module on requirements analysis were set an exercise where they would have to use a CBC. The students represented a spread of discipline from IT with management through to computer science. They were all novices at goal argumentation (such as goal graph refinement) but had a basic grasp of the principles. Most students had a year in industry and a large proportion of the students were on degrees that require a grasp of business imperatives. It was felt that this group would make an interesting test of the ease of learning and applying the

CBC technique. Despite their inexperience they are typical of the kind people who will eventually be involved in managing software and systems development projects.

In the exercise the students were instructed to prepare by writing a brief summary (100 words) of their project and then were instructed to develop a mind map of the stakeholders to the project and their respective concerns (limited to what they perceived to be the 20 most important stakeholder-concern items). They were instructed to create a CBC in their own time up to a deadline of three weeks. Again a budget as given to focus attention on expressing the full scope through the discipline of condensing what they perceived as the most important of 20 entries on the CBC. This occurred during a period that coincided with a high load of other academic work on each student.

We found that 30% (13) of the students created CBCs that were good enough to take directly to their project stakeholders for clarification and improvement by discussion and to be used as the basis for full G-R model refinement. 20% were adjudged as not understanding the technique as they needed more mentoring on the use of the technique before they could redraft their CBC and reach the standard of the higher cohort. The remainder 50% were judged as not suitable to be shown to stakeholders without prior intensive mentoring and rework.

Our technique has given us an insight that helps to distinguish those students who have the skills and ability to formulate abstract concepts about a future world. This discrimination correlated well with the more general observations of the students' tutors; people who have observed the students over several years.

The skill to formulate abstract concepts about a future world is crucial to prospective requirements analysts and it is one that industrial practitioners and students find difficult to acquire.

Looking at the CBCs produced by the upper 50% was instructive. It illustrated some weaknesses in the original formulation of the CBC, which led to the changes discussed in the above section. When using the original formulation it is easy to confuse whether a requirements statement refers to a benefit, assumption, constraint or satisfied concern. Sometimes the classification does not matter as the key outcome is the elicitation of important concerns.

Further Examples: The concise business case has been applied to other soft projects. For example a recent project between a major enterprise architecture service company and the University of Reading showed that the methods described here can be used to bring focus to a project as a whole and to stages (e.g. sprints) of the project. We can also report that in a dozen real projects considered the concise business case template proposed here in every case provides a robust and suitable template to commence a goal sketch. We have also observed its successful use in drawing up proposals and contracts.

Early indications suggest that the application of the CBC could be formulated as number of requirements analysis patterns. This matter is being investigated as further work.

VIII. CONCLUSION AND FURTHER WORK

We have shown how goal sketching can be accelerated by introducing a template *concise business case* and have corroborated our expectation using industrial case studies. The template will be one in a family of templates. We are also confident that there is an underlying analysis pattern: choose a template, map the business case to it and transform that into a structurally complete G-R model by adding such assumptions as necessary. The pattern also appears to have a fractal nature as it can be applied to the whole project or to its stages (or agile sprints). More work is needed to clarify and document the pattern.

G-R modeling with the concise business case is most appropriate to soft projects with uncertainty about *what* and *how*. Otherwise best practice project management methods (e.g. PBP) would be advised as more cost effective.

The use of the concise business case begins a goal refinement process in which techniques such as use-case goal refinement and KAOS can be used for additional rigor with regard to operationalizing functional requirements.

It was expected that the process of building a structurally complete model from the concise business case would nurture improved shared understanding among the stakeholders. Early signs are that this is indeed the case. This is apparent in the value of the assumptions identified when attempting to build a structurally complete model from the concise business case. Additionally we identify a potential synergy with assumption based planning [5] and its 'hedging' and 'sign-posting' tactics.

Further, the elicitation of assumptions can be helped by the identification of *weltanschauungen* using soft systems methodology [17].

Creating a structurally complete model based on the concise business case might best be considered as a digest of what is known and provides a project board's viewpoint. It compliments (and does not compete with) best practice requirements engineering and project management.

ACKNOWLEDGMENT

The authors would like to acknowledge their industrial collaborators. In particular colleagues in OSKIS Informatics Ltd and Secerno Ltd.

REFERENCES

[1] Boness, K. and Harrison, R. 2009. Goal Sketching and the Business Case. Software Engineering Advances, 2009. ICSEA'09. The Fourth International Conference on.

[2] Boness, K., Harrison, R., and Liu, K. 2008. Goal sketching: An Agile Approach to Clarifying Requirements. International Journal on Advances in Software, IARIA. 1, 1.

[3] Boness, K. and Harrison, R. 2007. Goal sketching: Towards agile requirements engineering. Software Engineering Advances, 2007. ICSEA 2007. International Conference on. 71-71.

[4] Boness, K. and Harrison, R. 2008. Goal Sketching with Activity Diagrams. Software Engineering Advances, 2008. ICSEA'08. The Third International Conference on. 277-283.

[5] Dewar, J. A. 2002 Assumption-based planning: a tool for reducing avoidable surprises. Cambridge Univ Pr.

[6] OGC 2002. PRINCE2: Managing Successful Projects. London: HMSO.

[7] Taylor, M. D. 2003. How to Develop Work Breakdown Structures.

[8] Checkland, P. and Winter, M. 2006. Process and content: two ways of using SSM. Journal of the Operational Research Society. 57, 12, 1435-1441.

[9] Crawford, L. and Pollack, J. 2004. Hard and soft projects: a framework for analysis. International Journal of Project Management. 22, 8, 645-653.

[10] Dardenne, A., Lamsweerde, A. V., and Fickas, S. 1993. Goal-directed requirements acquisition. Science of computer Programming.

[11] Koladis, G. and Ghose, A. 2006. Relating Business Process Models to Goal-Oriented Requirements models in KAOS. Advances in Knowledge Aquisition and Management.

[12] Yu, E. 1997. Towards modelling and reasoning support for early-phase requirements engineering. Proceedings of the 3rd IEEE International Symposium on Requirements Engineering (RE'97). 226.

[13] Bleistein, S. J., Cox, K., Verner, J., and Phalp, K. T. 2006. B-SCP: A requirements analysis framework for validating strategic alignment of organizational IT based on strategy, context, and process. Information and Software Technology. 48, 9, 846-868.

[14] Jackson, M. 2000 Problem Frames: Analyzing and structuring software development problems. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

[15] Adolph, S., Cockburn, A., and Bramble, P. 2002 Patterns for effective use cases. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

[16] Cockburn, A. 2002 Writing effective use cases. Addison-Wesley Reading, MA.

[17] Checkland, P. 1998 Systems thinking, systems practice. Wiley.

[18] APM. 2006 Project management body of knowledge. Association for Project Management (APM).

[19] PMI 2000. Guide to the Project Management Body of Knowledge, PMBOK Guide 2000 edition.

[20] Alexander, I. F. 2005. A Taxonomy of Stakeholders: Human Roles in System Development. International Journal of Technology and Human Interaction. 1, 1, 23-59.

[21] Simon, H. A. 1996 The sciences of the artificial. The MIT Press.

[22] Zave, P. and Jackson, M. 1997. Four dark corners of requirements engineering. ACM Transactions on Software Engineering and Methodology (TOSEM). 6, 1, 1-30.

[23] Boness, K., Finkelstein, A., and Harrison, R. 2008. A lightweight technique for assessing risks in requirements analysis. IET Software. 2, 1, 46-57.

[24] Turner, J. R. 2008 The handbook of project-based management. McGraw-Hill Professional.

[25] Krabbel, A., Wetzel, I., and Züllighoven, H. 1997. On the inevitable intertwining of analysis and design: developing systems for complex cooperations. Proceedings of the 2nd conference on Designing interactive systems: processes, practices, methods, and techniques. 205-213.

[26] Nuseibeh, B. 2001. Weaving together requirements and architectures. IEEE Computer. 34, 2, 115-117.

[27] Hall, J. G. and Rapanotti, L. 2009. The discipline of natural design. Undisciplined! Proceedings of the Design Research Society Conference 2008. 2.

[28] Alexander, C., Ishikawa, S., and Silverstein, M. 1977 A pattern language: towns, buildings, construction. Oxford University Press, USA.