Copy 1

NO.84
(1988)

CI/SfB
(Agn)(H16)
UDC 624.042.7:69(931)

BUILDING RESEARCH ASSOCIATION OF NEW ZEALAND

REPRINT

The development of an expert system for seismic loading

J.G.Hosking,
S.Lomas,W.B.Mugridge
and A.J.Cranston.

This report originated in work done in collaboration with BRANZ as part of its ongoing research into knowledge based systems. The views represented are not necessary those of the Association.Reprinted from *Proceedings of Symposium on Knowledge-based systems in Civil Engineering*, August 1988, Monash University, Clayton.

branz
BUILDING RESEARCH ASSOCIATION OF NEW ZEALAND

THE DEVELOPMENT OF AN EXPERT SYSTEM FOR SEISMIC LOADING

J. G. Hosking, S. Lomas, W. B. Mugridge
University of Auckland,
NEW ZEALAND

A. J. Cranston
Worley Consultants Ltd.,
NEW ZEALAND

ABSTRACT

This paper describes Seismic, a system to assist structural designers to check their evolving designs against the seismic loading requirements of the Draft New Zealand Standard for structural design and design loadings, DZ4203:1986. Included in the paper is a discussion of the methodologies found useful in the development of Seismic.

INTRODUCTION

A knowledge-based system, *Seismic*, which assists building designers to meet the requirements of a seismic loadings code, is under development as part of a collaborative programme being pursued by the authors with the Building Research Association of New Zealand (BRANZ). The emphasis in this collaborative programme has been on the development of tools and methodologies useful in the construction of knowledge-based systems for building applications and, in particular, applications based on codes of practice (Whitney, 1987; Hosking et al, 1987a,b; Fowkes et al, 1988; Mugridge and Hosking, 1988; Dechapunya and Whitney, 1988).

This project should thus be seen as one in a series of investigations into the development of code-based expert systems. The progression in this series has been to start with a very explicit, highly structured, code (Hosking et al, 1987b) and to then move on to progressively less explicit, less structured codes. A parallel progression has been to move from systems which are predominantly requirement checkers through to systems where design components could usefully be incorporated (Mugridge and Hosking, 1988). Underpinning the application development has been the concurrent development of Class Language (Mugridge et al, 1987; Hamer et al, 1988) as a vehicle for implementation. Thus the principles and methodologies induced in the course of application development have had a considerable influence on the facilities provided in Class Language.

A byproduct of these projects has been a growing body of knowledge about what types of knowledge are (and should be) encoded in building codes. This pool of *metaknowledge* provides a valuable source of material for future developers of code-based expert systems and also for the developers of the codes of practice.

Following this introduction is a description of the background of the *Seismic* project. This precedes a discussion of the development of *Seismic*, including the methodology behind the development process and, in particular, a description of knowledge acquisition techniques used. A brief discussion of the relationship between knowledge based system construction and codes of practice then leads on to some concluding remarks.

THE CODE AND THE RESEARCH PROJECT

Seismic activity and the philosophy of seismic design

About eighty per cent of the world's earthquakes originate in the circum-Pacific belt (Park, 1981). It is no surprise, therefore, that New Zealand is actively involved in researching the nature of earthquakes and how to avoid the devastation and loss of life typified by such infamous earthquakes as the 1985 Mexico City earthquake. Currently there is no way to accurately predict the occurrence of such earthquakes and no way to avoid the effects of a major earthquake. However we may strive to design buildings which will resist such an event while minimising loss of life and damage to property.

The effect of an earthquake at ground level is a violent shaking of the earth. For example, the horizontal movement of the earth measured in the 1985 Mexico earthquake was about 23 cm every 2 s (Smith, 1986). Considerable vertical movement also occurs; however this is usually ignored when designing buildings to withstand seismic forces because of the large safety margins already built into the structure for withstanding gravitational forces. Hence seismic design is principally concerned with adequate management of seismic lateral loads.

The primary consideration in seismic design is the preservation of life. This of course means that the building must not collapse in any major way. A secondary, but economically important, consideration is that in a minor earthquake the building must remain undamaged, or at most suffer damage that is easy to repair. A major difficulty, however, is that there is no maximum magnitude for an earthquake. Hence most building codes include probabilistic factors which allow an assessment of the *likely* seismic lateral load that a building will meet *in the course of its expected lifetime*. A common formula for evaluating this design force or *base shear* (which the building must then be designed to withstand) is (Smith, 1986):

$$V = Z\ I\ K\ C\ S\ W$$

Where:

V is the base shear
Z is the zone reflecting the severity and likelihood of an earthquake in a given area
I is the importance of the building (eg hospital vs residential)
K is a function of the frame type
C is inversely proportional to the period of the building
S represents the type of soil the building is sited on
W represents the weight of the building

A building designer uses the design base shear as a basis for performing a structural analysis of the building to check that, amongst other things, the deflections produced by that base shear are within reasonable values. Several analysis methods exist. The equivalent static method of analysis, for example, models the dynamic motion of an earthquake as an equivalent set of static forces distributed over the building frame; the resulting static deflections are then assumed to be equivalent to the maximum likely dynamic deflections. This assumption is only valid for a limited range of buildings. For example, the building must be reasonably regular, both vertically and horizontally, otherwise an alternative method, such as spectral modal analysis, must be used.

The code

Building codes are intended to ensure that the results of research are applied to the process of building design. As more research is conducted, the codes are amended or superceded. *DZ4203:1986 General Structural Design and Design Loadings for Buildings* (SANZ, 1986) is a draft New Zealand Standard intended to supercede the older *NZS4203:1984*. Part C of the new draft standard is concerned with earthquake provisions. Other parts of the code relate to general provisions, provisions for dead and live loads, and wind and environmental loads.

Figure 1 shows two typical clauses from Part C of the code, concerning the appropriate design method to use as a function of the vertical regularity and translational period of a structure. The code also provides a small amount of commentary (not shown) justifying the clause requirements.

> 5.2.1 A building shall be considered to be vertically regular if, when the effect of base rotation is excluded, the difference in horizontal deflection divided by interstorey height between levels i+1 and i+2 as well as (for i not less than 2) between levels i-1 and i is not more than twice the same ratio measured between levels i and i+1. This criterion need be applied only up to a maximum value of i of 0.75n.
>
> 5.2.2 Structures that do not satisfy the criteria for vertical regularity and structures whose natural period of vibration in the fundamental translational mode exceed 1 second shall be analysed for the effects of this motion by the methods of either Section 9 or 10 and the equivalent static method of Section 7 shall not be used.

Figure 1. Two clauses from Part C of DZ4203:1986

The research project

The Seismic project was commissioned by BRANZ in 1987. The immediate aim of the project was the development of a knowledge-based system to assist a building designer to interpret the seismic loading requirements of DZ4203. However, the stated objectives of the project emphasised research rather than product development. These were:

1. To gain further experience in using knowledge-based systems technology for building codes.
2. To identify guidelines for the development of future codes of practice to facilitate their use and conversion to knowledge-based systems.
3. To obtain a better understanding of the resource implications and feasibility of knowledge-based systems based on major parts of New Zealand's building codes.

The *FireCode* project (Hosking et al 1987b) had made some tentative conclusions concerning objectives 2 and 3. It was hoped that the broader experience provided by this project and the *Wallbrace* project (Mugridge and Hosking, 1988) would provide firmer recommendations. In pursuing objectives 1 and 3, a review of the suitability of various approaches to knowledge acquisition was considered important as was the fostering of further development of Class Language by providing experience in a new application area.

INITIAL WORK

Preparation

Initial work on the project aimed to familiarise the developers with the problem domain and then to define a subset of that domain suitable for development as a knowledge-based system. The role of seismic loadings expert was provided by one of the authors (Cranston). The remaining authors (the developers) performed the knowledge engineering and implementation tasks.

The domain background was provided by an initial tutorial session, giving an overview of building design for sesimic safety, together with an ordered series of papers provided by the expert. These allowed the developers to understand the context in which the code had been developed and the intentions and philosophies behind the development of seismic loadings codes.

Tutorial sessions followed where the expert made the link between the background reading and DZ4203:1986, and, in particular, those parts of that code relating to seismic loading requirements. The intention in these sessions was to introduce the developers into the use of the code in practice. It soon became apparent to the developers that the code of practice assumed a large amount of background on the part of its users. The code aims to specify *what* is required of a building but places little emphasis on *how* that is to be achieved. Hence it was felt that the major focus in developing the *Seismic* system would be on identifying a suitable methodology for applying the code. This is in contrast to the developers' experience with the development of the *FireCode* system (Hosking et al 1987b). DZ4226, a draft fire safety code, provides much more *strategic* information, specifying how the code is to be applied, than does DZ4203.

Initial reading by the developers had identified a cycle within the requirements of DZ4203. The definition of translational period is in terms of the Rayleigh formula, where the period is a function of (amongst other things) the distributed shear forces and deflections those forces produce. The base shear is, however, a function of the translational period, as are the deflections (see the clauses in Figure 1). It was not at all apparent to the developers how this cycle was broken in practice.

The tutorial sessions were used to introduce the various components of the building design cycle, and how Part C of DZ4203 related to this cycle. This clarified the problems raised above and made apparent the importance of accurate empirical estimates of quantities, such as the translational period, both in starting off the design analysis process and in minimising the number of iterations required to provide a satisfactory analysis. It was also clear that any practical system could not deal with the requirements of Part C of DZ4203 in isolation and that substantial sections of other parts of the code would need to be integrated.

Specification of the *Seismic* System

The preparatory work clarified the problem domain sufficiently to allow a preliminary definition of what the *Seismic* system could usefully provide and what assumptions could be made of its intended users. The following specification resulted:

The system should be aimed at a relatively novice designer to guide that designer quickly, yet consistently, through the analysis process. However, familiarity with domain terminology could be assumed, together with some familiarity with DZ4203 and the philosophy behind it.

The emphasis of the system should be on providing help in the preliminary design analysis stages, particularly where the user is 'roughing out' designs and wants to experiment with a range of designs. Identification of appropriate empirical estimates to start the analysis process is therefore important; it is this information that is lacking from the code and is possibly unknown to a novice designer. The ability to iteratively develop a design should also be considered important.

The system should concentrate on earthquake loading requirements, but that this should be done in the context of the overall design analysis. Hence construction of the system should take into account possible integration of other parts of the loadings analysis, such as gravitational loading.

The system should concentrate on the equivalent static method of analysis. This is the easiest to perform and, being a linear method, simple scaling of the analysis results is possible if initial estimates of quantities such as the translational period are not too much in error when results are reviewed. Even when the acceptance criteria of the equivalent static method are violated, useful initial design information can be obtained by performing such an analysis. Hence consideration of the requirements in Sections 9, spectral modal analysis, and 10, numerical integration time history analysis, of Part C should be deferred.

The availability of a standard frame analysis package should be assumed. There are many commercially available frame analysis packages, hence the development of a new one is unproductive. Rather, the system should concentrate on providing data in a suitable form for use in such a package (PFRAME was used as an example when developing *Seismic*).

Consideration of Section 8 of Part C, seismic loading for parts and their connections, should be deferred. The requirements of this section are better considered after the member design phase and are hence outside the major scope of the system as a preliminary analysis tool.

In addition to the development of an executable knowledge base, both the expert and the developers felt that the formalisation of a methodology for using the code was useful in its own right. The expert emphasised that such a formal methodology would have considerable quality control benefits. If it was adopted by all designers in an organisation, for instance, review of one designer's work by another would be simpler. By formalising the methodology, resource material would also be available to teach novice designers the correct method of applying the code.

Assessing the specification

Prerau (1985) provides a very good checklist of domain and project characteristics that are likely to lead to a successful expert systems project. His discussion is principally aimed at first-time developers in a commercial environment. Nonetheless, the developers found this checklist to be useful both in the development of the specification of the previous section and in the assessment of other projects (Mugridge and Hosking, 1988). Typical items in the checklist (paraphrased from Prerau, 1985) with comments concerning the *Seismic* specification are:

Expert knowledge is needed: This is quite clearly so. To apply DZ4203 considerable knowledge of building design principles and practice is required. Much of this is not included in the written code document.

Conventional approaches are unsatisfactory: Parts of the process of designing a building within the requirements of DZ4203 lend themselves to conventional computing approaches. However, there is sufficient complexity to regard conventional approaches to the entire problem as being inappropriate.

Reputable expertise is available: Suitable experts were available, both for knowledge acquisition and evaluation.

The task is decomposable: The seismic loadings analysis has many, relatively easily isolated, subcomponents. This assists in prototype development as smaller subproblems can be tackled on an individual basis.

Most items on the checklist were answered favourably when matched against the specification. However the answers to two items indicated potential problem areas:

The problem requires a narrow domain of knowledge: This is generally so with the seismic loadings problem. However, it was recognised early on that the system could require extensive spatial reasoning. The risk that this could prove intractable was felt to be justifiable for a research project.

The domain is stable: This is not true for the *Seismic* system. DZ4203 is a draft code - it is a revision of many previous codes and will itself be revised before adoption as a New Zealand standard. However, as the principle aim was research rather than product development it was felt that the risk of developing a system that might not be applicable was worthwhile. In addition, the authors have some interest in using expert system development as a means of testing out and providing assistance in the development of new codes of practice (Hosking et al, 1987b). Hence, from a research sense, we have some interest in investigating the influence domain instability has on successful expert system development.

With these caveats system development began.

SYSTEM DEVELOPMENT

Development methodology

In general terms, the development of *Seismic* involved a cycle of knowledge acquisition, prototype implementation or modification, and review. The prototypes provided a way of demonstrating to the expert the developers' growing understanding of the domain.

Choice of initial prototype

The initial tutorial sessions had given the developers a good overview of the analysis-design cycle. The ready decomposability of the task, identified above, allowed the developers to focus attention on particular subcomponents of the task, while initially ignoring or treating in a shallow manner other subcomponents. Hence initial attention was focussed on the preliminary analysis phases, and in particular the distribution of base shear throughout the structure. Later parts of the task were ignored at this stage.

Within the task of distributing base shear, some subtasks, such as translational period estimation, were treated in depth, while others, such as vertical and horizontal regularity estimation, were initially approached in a very shallow manner. This combination of a broad but shallow approach to a major task, combined with a narrow but deep approach to some of its subtasks is a very typical way of beginning the development of a knowledge-based system.

It is very important to focus on a suitable initial prototype as early as possible and to then commence implementation immediately. In retrospect, we feel that we spent too much time performing knowledge acquisition prior to implementation (Smith and Baker, 1983 note the same problem) when a concrete demonstration of our understanding in the form of a prototype would have been more productive.

Prototype revision

Having chosen the initial focus areas, development of the *Seismic* system proper commenced. Typically, incremental development of the prototype occurred as one of the subtasks initially treated in a shallow manner was explored in more detail in a knowledge acquisition session.

At several points in the development process it became obvious that major revisions to the model embodied in the existant prototype were required. These revisions required major reimplementation of the system to embody the changes. Initially, for example, we had concentrated on examples illustrating the process of design analysis. These examples had emphasised the step-by-step sequence of operations usually performed by the expert to solve the problem. The first version of *Seismic* was thus principally a procedural model of loadings analysis. However, this procedural approach gave no clear reasons for the ordering of activities.

The second version, developed as a result of insight gained from the development of the first version, used a goal-oriented model working back from the required results of the system to investigate the information needed to produce those results. The sequence embodied in the first version was a natural byproduct of the goal-oriented needs of the second version.

IMPLEMENTATION

Modularity and Class Language

Modularity of the knowledge base is critical to the success of a prototyping approach. It must be possible to easily reorganise the knowledge base as the knowledge engineer's understanding of the problem evolves. If there is minimal or inappropriate modularity, it is difficult to restructure the system to keep the design clean and clear.

Seismic is implemented in *Class Language* (Mugridge et al, 1987). Often, major revisions in the overall structure of *Seismic* were very easily achieved due to the modular structuring encouraged by Class Language. Typically such changes involved a 'reshuffling' of the class and object structure, with minimal change to many of the classes. A similar economy of revision was noted in the development of *Wallbrace* (Mugridge and Hosking, 1988).

Modularity in *Class Language* is provided through the following mechanisms:

- classes and associated rules and procedures
- aggregation of objects
- parameterised classes
- classification and inheritance

Using Classes

Class Language is an object-oriented language, so that modularity is achieved through modelling the different classes of objects found in an application. Classes are defined according to the useful aspects that have to be modelled. Some objects correspond to obvious objects in the world, such as *storey*, while others will be rather abstract, such as *direction*.

Rules and procedures are associated with classes, so the information that is tightly coupled is found together. Often changes to a class will not have any effects outside that class.

Aggregation of Objects

Objects of classes may have components which are in turn objects of other classes or sets of objects of another class. This explicit representation of aggregation is extremely important in providing a clear model of, for example, the structure of a building. Figure 2 shows the relationships between classes and objects in the current version of *Seismic*. The solid lines there represent component relationships.

Parameterised Classes

The calculations for the loading ALONG and ACROSS the building follow the same logic. It is inappropriate to duplicate this logic for the two directions, so a class was defined which generalises the aspects of both directions, having suitable parameters to distinguish the two directions for the user. Classes allow suitable abstractions, reducing the overall detail and making explicit the commonality between different parts by creating a more general class.

Classification and Inheritance

In addition to the aggregation abstractions discussed above, class-subclass abstractions are also important. For example, Table B-1 of DZ4203 encodes appropriate live load figures for different categories (such as Institutional and Residential) and subcategories (such as operating theatres and dining rooms) of room usage. It is useful to model these categorisations explicitly and this is possible in *Class Language*. In Figure 2 the dotted lines represent class-subclass relationships.

Computationally these relationships are used by the inheritance and classification mechanisms of *Class Language*. Inheritance allows objects of a subclass to inherit features described in a more general class. This allows considerable economy by eliminating unneccessary redundancy. Classification permits an object to be defined to be of a general class (such as table-b1) and to be dynamically classified to one of the subclasses (such as agriculture). More details on these mechanisms may be found in Mugridge et al (1987) and Hamer et al (1988).

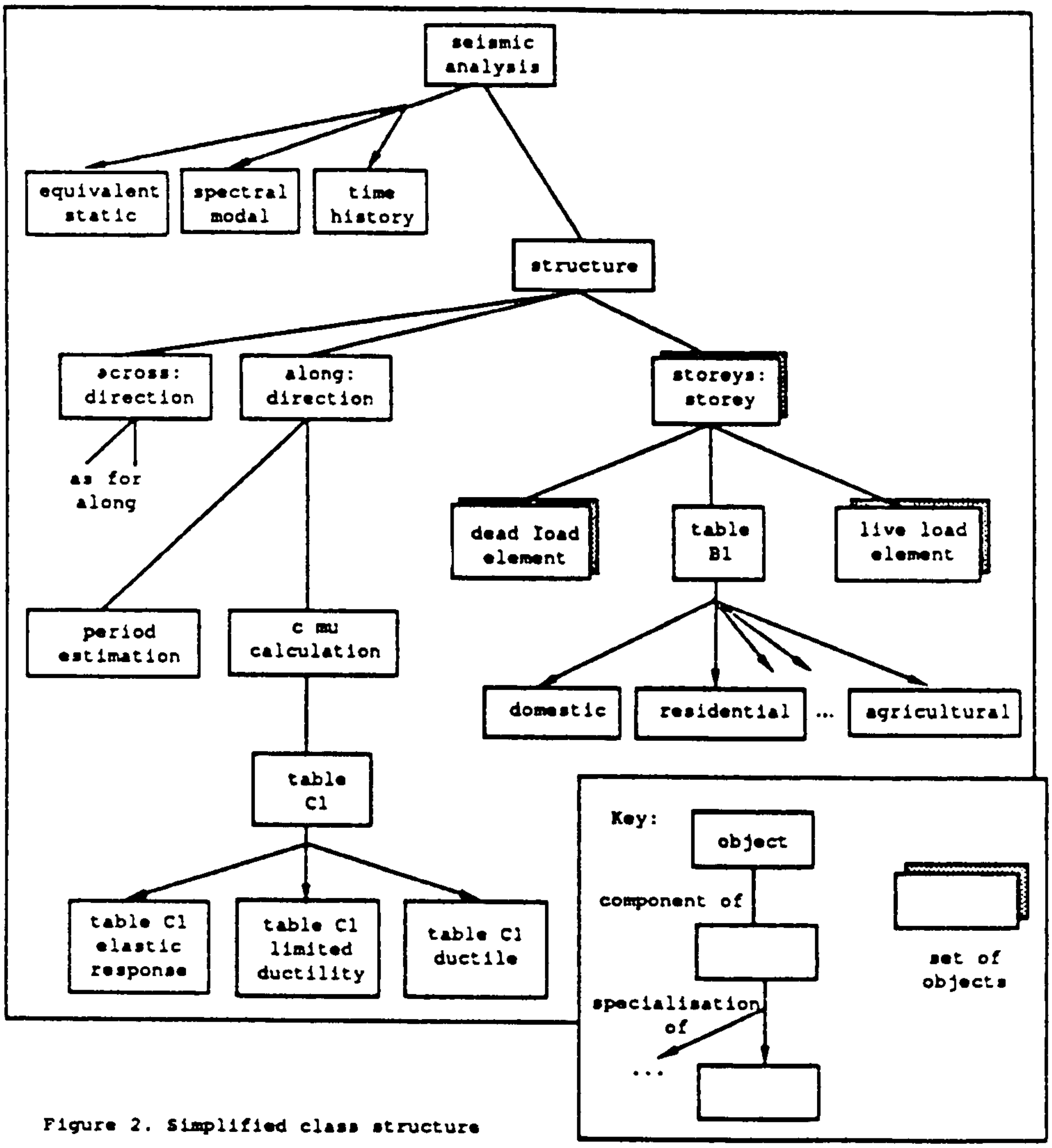


Figure 2. Simplified class structure

Handling short cuts

In many cases, it was desirable to avoid asking many questions when it was inappropriate. For example, in seeking information about each level of a structure it was tedious to seek the same information repeatedly for levels which had the same values. Handling such shortcuts proved to be the most difficult part of system development as, by their nature, the shortcuts tend to cut across the neat structures set up to handle the more general case.

The current prototype

Functional decription

The current prototype performs the following functions:

1. An estimate of the translational period of the building in two orthogonal directions.
2. An estimate of whether the acceptance criteria of the equivalent static method have been met.
3. An estimate of the base shear coefficient in two directions.
4. An estimate of the seismic weight of the building.
5. An estimate of the base shear in two directions for use in an equivalent static analysis.
6. A distribution of the sesimic lateral load for use in an equivalent static analysis.

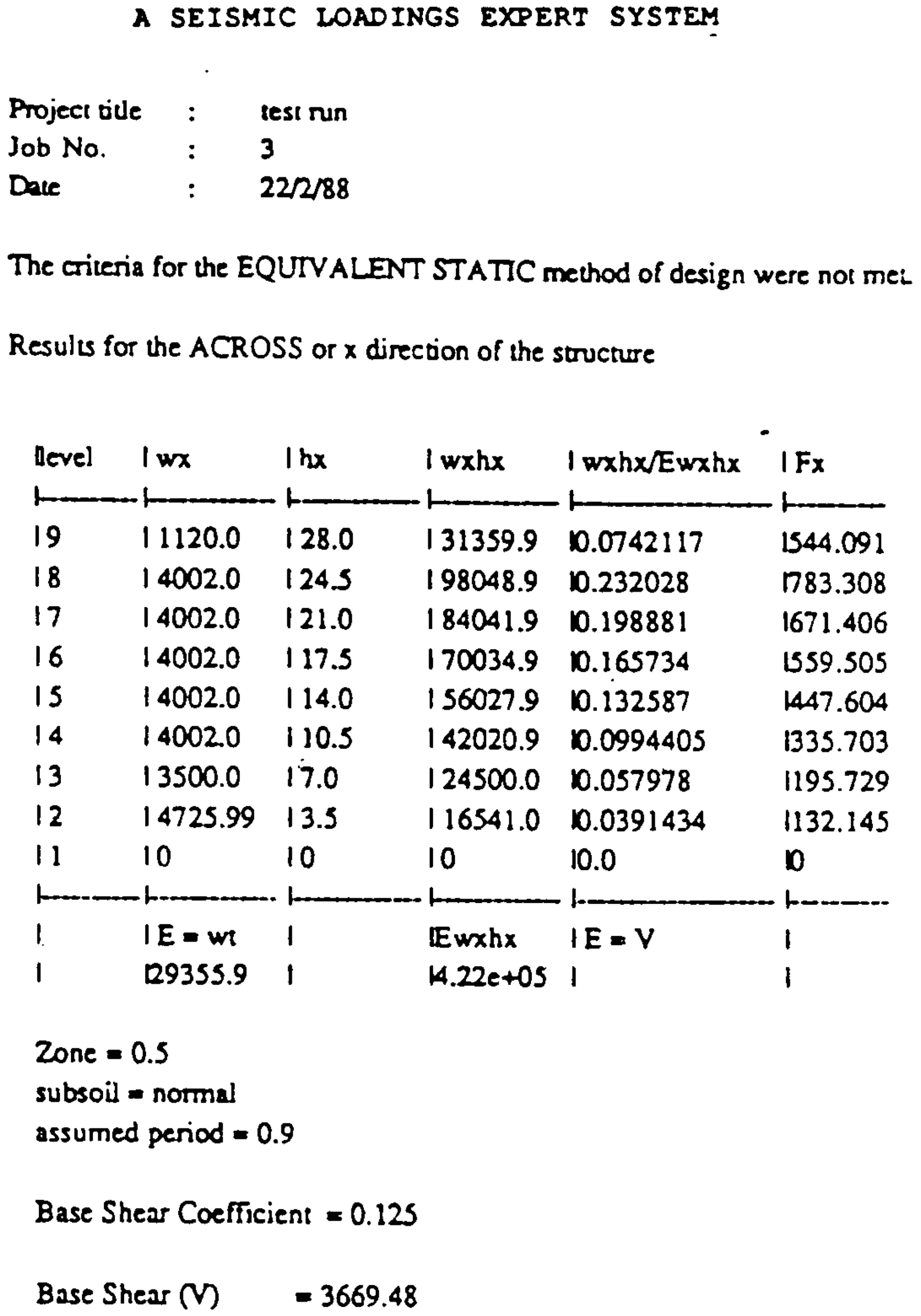
The results are presented in a tabular fashion suitable for use in analysing the frame model. A typical example is shown in Figure 3. In fact the goal of producing this table drives the execution of *Seismic*. For example, in order to generate the table, the analysis method must be known, as must the number of levels and their seismic weights, heights, etc.

Extending the prototype

The current prototype is incomplete with respect to the specification given earlier. Knowledge acquisition has proceeded beyond the current implementation. Hence good estimates can be made about the ease or difficulty of extending the functionality of *Seismic*.

One class of extensions is quite straightforward and should involve few problems. This class involves expanding some of the subtasks which are currently handled in a shallow manner so that they are more competently dealt with. A representative examples is the estimation of horizontal and vertical regularity of structures. The current prototype simply asks the user whether the structure is horizontally and vertically regular. The expert has provided background material to assist the user better in empirically esimating structural regularity.

The second class of extensions may involve significant effort. The principal problems in this class are a requirement for more complex spatial reasoning than the current model would permit and incomplete knowledge acquisition. An example is performing torsional analysis. This would involve, amongst other things, a distribution of the torsional moment to the structure elements requiring a more complete representation of the structure frame than currently exists. As there are potentially a large number of structural elements this could cause severe problems with spatial representation.



```
A SEISMIC LOADINGS EXPERT SYSTEM

Project title   :   test run
Job No.         :   3
Date            :   22/2/88

The criteria for the EQUIVALENT STATIC method of design were not met.

Results for the ACROSS or x direction of the structure

|level  |wx        |hx       |wxhx       |wxhx/Ewxhx     |Fx
|-------|----------|---------|-----------|---------------|-------
|9      |1120.0    |28.0     |31359.9    |0.0742117      |544.091
|8      |4002.0    |24.5     |98048.9    |0.232028       |783.308
|7      |4002.0    |21.0     |84041.9    |0.198881       |671.406
|6      |4002.0    |17.5     |70034.9    |0.165734       |559.505
|5      |4002.0    |14.0     |56027.9    |0.132587       |447.604
|4      |4002.0    |10.5     |42020.9    |0.0994405      |335.703
|3      |3500.0    |7.0      |24500.0    |0.057978       |195.729
|2      |4725.99   |3.5      |16541.0    |0.0391434      |132.145
|1      |0         |0        |0          |0.0            |0
|-------|----------|---------|-----------|---------------|-------
|       |E = wt    |         |Ewxhx      |E = V          |
|       |29355.9   |         |4.22e+05   |               |

Zone = 0.5
subsoil = normal
assumed period = 0.9

Base Shear Coefficient  = 0.125

Base Shear (V)          = 3669.48
```

Figure 3: Sample summary output produced after running Seismic. Approximately 60 questions preceded this summary.

KNOWLEDGE ACQUISITION

Background

An important component of this research project was the examination of different knowledge acquisition methods, and their usefulness for developing knowledge-based systems for building codes. Some useful work has been done on the development of tools to assist knowledge acquisition "in the small", where the emphasis is on polishing an existant system (e.g. Davis, 1979). However, very little work has been reported on knowledge acquisition in the large, i.e. techniques for gaining enough information to develop a conceptual model from scratch. As Hoffman (1987) points out most papers with 'knowledge acquisition' in their title turn out to have very little to do with knowledge acquisition and rather more to do with implementation technology.

Two principle issues were focussed on in the course of the *Seismic* project: completeness of record and notation. The former issue involved the need to record all of an interview between the expert and the developers for later analysis (since it was often not obvious during an interview what was unimportant). The latter involved deciding how best to represent the information gathered for easy inclusion in the model and for feedback to the expert. It is usual to separate these issues, and concentrate on completeness of recording during an interview with later *transcription* and *analysis* being used to change the notation.

Recording techniques

An audio tape recorder is commonly used to obtain a "verbal protocol" of an interview session. However, this method misses body language (which can indicate inference-reasoning) and may suffer from recording quality problems (um's ahs' etc tend to lose their semantic meaning and fade into the background) (Hoffman, 1987). Prerau (1987) made intial use of audiotape in the Compass project. However, it was soon abandoned as the expert was able to proceed slowly enough to permit satisfactory note-taking.

An alternative to audiotape is videotape which was used extensively to record interviews during the Seismic project. This method has the advantage of recording body and hand movement and also transient 'white-board' diagrams. The following observations resulted from our experience (many of which relate to other complete recording methods):

Use a cameraman who is not part of the interview team.

The operation of a camera involves considerable effort, particularly as the aim is to capture both body movement (particularly pointing) and diagrams. With only one camera, as was the case with *Seismic*, constant changes in speaker location and focus meant that the cameraman had to concentrate almost exclusively on filming. Initially one of the developers acted as cameraman, however that person found that he was unable to take an active part in the interviewing and had difficulty remembering the detail of what was discussed. As a result another person was employed specifically to film the interviews.

This problem has a parallel in the case of written transcription. Early experience in the Seismic project showed that one cannot be both transcribing an interview and taking an active part in it. Either the interview must be interrupted for periods of transcription or a *scribe* must be used to make the written transcription. The former option was taken by Prerau (1987).

Videotaping can be intrusive.

Videotaping may be unsuitable with some experts as they clam up in front of the rather obvious video equipment. Although this was not a major problem in the case of the Seismic project, it was noticeable that the expert occasionally talked 'to the camera', and that sessions without videotaping seemed 'freer'. Hart (1986) noticed the same phenomenon.

*Videotape provides a **real time** record of an interview.*

The whole point of using video (or audio) tape is to obtain a complete real time recording of a knowledge acquisition session allowing the transcription to a suitable notation to be deferred till after the session. This means that maximal use is made of the expert's time. However, later transcription of the video (or audio) tape takes *much longer* than the time taken for the original session. This extra time may be sufficient to encourage use of written transcription methods.

In the case of *Seismic*, after initial experiments, a judicious combination of methods was found to be best with written notes supplementing the video recording. The written notes were used to crudely index the videotape (or the videotape was reviewed at speed to create such an index). The videotape was then used (1) as an archival record (2) to review poorly understood parts of the interview (using the crude index) (3) as a record of the transient diagrams.

Notation

The notation used to record information has a significant bearing on its long-term usefulness. Recorded information should be available in a form to provide feedback to the expert in later sessions. Hence a notation that the expert is comfortable with is necessary. The implementation languages available currently are pitched at the wrong level to allow direct transcription. Also, using one implementation language (i.e. notation) is likely to be very restricting (see for example Smith and Baker, 1983). For most problems (or subproblems) there is a fairly 'natural' notation that fits the problem well and provides a good focus for further information gathering; using any other notation slows progress. This notation may not be one the expert uses routinely and so a major job of the knowledge engineer is to be inventive in the selection or development of appropriate notations that allow the problem to be explored.

Prerau (1987) takes a somewhat different (although fairly common) approach to recording information. He believes that quasi-English if-then rules and procedures (where appropriate) are a good way of recording information. His arguments are that this makes translation to a rule-based implementation easy and that the rule-based paradigm is probably easier than other AI paradigms for an expert to understand. However his arguments assume a rule-based implementation and hence the acquisition process is immediately constrained by the known implementation possibilities. It may be that quasi-if-then rules worked well for the Compass project, because they were in some way natural for it, but this is not an argument for using them generally.

In the course of the *Seismic* project a number of notations were found useful. At the start of the project procedurally oriented charts (describing the sequence of events to be performed in designing a building within the requirements of the loadings code) were found useful (see Figure 4). However, as noted earlier, it soon became apparent that this notation was limited. It was difficult to understand why a particular order was imposed on operations - that was just the way it was.

ANALYSIS PHASE			DESIGN PHASE	
1. DATA ASSEMBLY	2. ANALYSIS	3. REVIEW RESULTS	4. MEMBER DESIGN	5. PARTS & PORTIONS
1.1 STRUCTURE FORM & GEOMETRY 1.2 LOADINGS 1.2.1 Assemble Load Combinations 1.2.2 Gravity Loads - Dead Loads - Live Loads 1.2.3 Wind Loads 1.2.4 Environmental Loads 1.2.5 Earthquake Loads 1.3 MEMBER PROPERTIES 1.3.1 Section Properties 1.3.2 Material Properties	2.1 GRAVITY LOAD ANALYSIS 2.2 LATERAL LOAD ANALYSIS 2.2.1 Method of Analysis Model of Analysis 2.2.2 Wind Loading 2.2.3 Earthquake Loading: Translational Load & Torsional Load Analysis	3.1 STIFFNESS 3.1.1 Structure Period 3.1.2 Interstorey Deflections		

Figure 4. Procedural chart

A better notation was then adopted, essentially a goal-oriented description of the task, (Figure 5). This allowed attention to focus on the reasons for choosing a particular strategy. While the goal-oriented approach was new to the expert, it was not awkward or uncomfortable for him to use. In fact he adapted the original method as explained to him to suit his own needs. This new method seemed to interest him, and provided him with a different perspective to the problem. After one meeting in which the expert was introduced to the concept of a goal-oriented approach, he candidly talked about the illogical nature of his method of describing and using the (loadings) code and indeed of the way the code is organised. I.e. this new notation was somewhat more 'natural' to the problem, even though it was a novel notation for the expert.

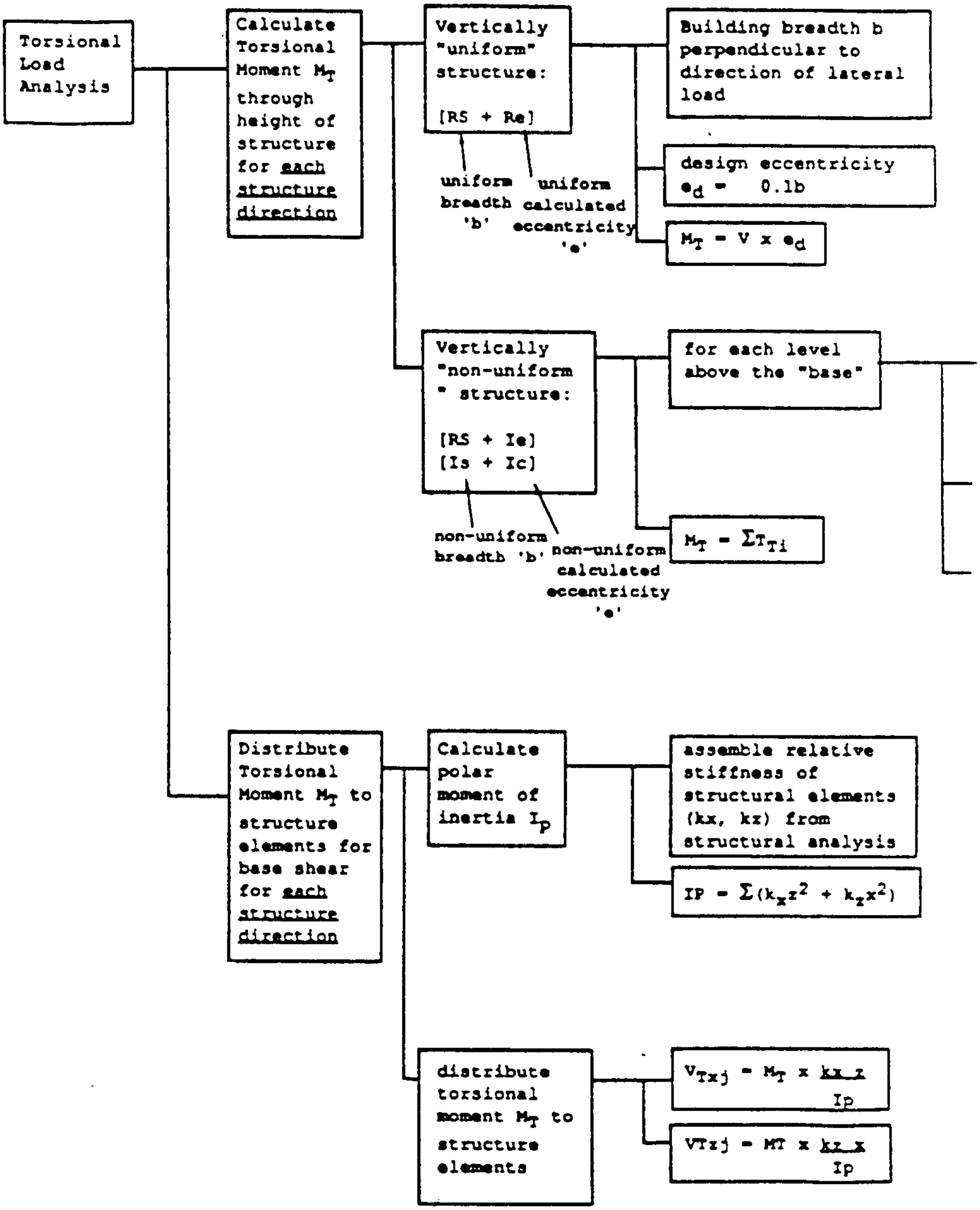


Figure 5. Part of a goal-oriented chart

THE RELATIONSHIP BETWEEN KBS AND CODES OF PRACTICE

One of the aims of the authors' programme of work has been to investigate the strengths and weaknesses of codes of practice as knowledge sources. The hope is that this information will be of benefit to future developers of the codes of practice.

As an example, one would expect that it would be relatively simple to encode the information from a table in a code of practice into a knowledge based system. Unfortunately, this is often not the case. The sorts of problems that arise are:

- The purpose of the table is unclear. This occurs, for example, in the draft fire code DZ4226 (Hosking et al, 1987b).

- Information is missing. This occurs, for example, in Table C-1 of DZ4203, which defines the ductility for different categories of structure. Some structure types do not have values associated.

- There are ambiguities in the terms used. This occurs in all the regulation-based systems we have constructed.

- It is not clear whether it is possible to interpolate or extrapolate the figures given. This occurs in Table 11 of NZS 3604, in the wall bracing provisions (Mugridge and Hosking, 1988).

- A set of examples is given and the user is expected to choose the example most similar to their case. Often it is not spelled out that the user is expected to interpret the table in this way. This occurs, for example, in the draft fire code DZ4226 (Hosking et al, 1987b).

- Important information is "hidden" away in footnotes to tables.

The authors believe that such problems arise through: general cases not being considered by the code writers; a conflict between the need to be simple for the simple cases, and the need to have a complete specification for all cases; and unstated assumptions being made about what is usual.

CONCLUSIONS

Summary

Seismic is still very much a prototype and is incomplete with respect to the specification described earlier. However, being primarily a research project, the aim was not *product* development, but rather an examination of the process of development. Hence, we have taken pains to discuss the methodology used in the course of developing *Seismic* as an aid to those embarking on the construction of their own systems.

Future work

An interesting future project will be to examine the difficulty of modifying the current *Seismic* implementation to match the requirements of the impending NZ Standard. The authors (as yet unsubstantiated) feeling is that the major structure of the system will be unaffected as the major design strategies are mostly independent of the code requirements.

The authors plan to extend their consideration of the strengths and weaknesses of codes of practice as knowledge sources. Our feeling is that committees developing codes of practice could usefully include trained knowledge engineers whose task would be to provide expert assistance on the organisation of the knowledge to be encoded (in written form) in the code of practice. We also feel that concurrent development of a knowledge base system with a code of practice would be a valuable way of providing immediate feedback to the code developers about the information they include in the codes.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the long-term collaboration and financial assistance provided by the Building Research Association of New Zealand, and in particular, the support of Dr Haris Dechapunya. We also gratefully acknowledge John Hamer for implementing *Class Language*.

REFERENCES

Davis, R. (1979). "Interactive transfer of expertise: acquisition of new inference rules", Artificial Intelligence, Vol. 12, 121-158.

Dechapunya, A. H. and Whitney, R.S. (1988), "Knowledge-based systems for building technology in New Zealand", to be presented to Symposium on knowledge-based systems in civil engineering, Monash, August.

Fowkes, A.H.R., Sharman, W.R., and Dechapunya, A.H. (1988) "The development of marketable knowledge-based systems" to be presented at The 3rd New Zealand Expert Systems Conference, Wellington, New Zealand, May.

Hamer, J., Hosking, J.G. and Mugridge, W.B. (1988), "The evolution of Class Language", to be presented at The 3rd New Zealand Expert Systems Conference, Wellington, New Zealand, May.

Hart, A., 1986: "Knowledge Acquisition for Expert Systems", McGraw-Hill, New York.

Hoffman, R. R. (1987). "The problem of extracting the knowledge of experts from the perspective of experimental psychology", AI Magazine, Vol. 6(Summer), 53-67.

Hosking, J.G., Mugridge, W.B. and Buis, M. (1987a), "Expert systems for regulations and codes", Proc. of the 2nd New Zealand Conf. on Expert Systems, Auckland, New Zealand, Feb, 37-47.

Hosking, J.G., Mugridge, W.B. and Buis, M. (1987b), "FireCode: a case study in the application of expert system techniques to a design code", Environment Planning and Design B, Vol. 14, 267-280.

Mugridge, W.B., Hosking, J.G., and Hamer, J. (1987), "Class Language - a language for building expert systems", Proc. of the 2nd New Zealand Conf. on Expert Systems, Auckland, New Zealand, Feb, 173-180.

Mugridge, W.B. and Hosking, J.G. (1988), "The development of an expert system for wall bracing design", to be presented at The 3rd New Zealand Expert Systems Conference, Wellington, New Zealand, May.

Park, R. (1981), "Review of code design for earthquake resistant design of concrete structures in New Zealand", Bulletin of NZ Nat. Soc. Earthquake Engineering, Vol. 14, 177-208.

Prerau, D.S., (1985), "Selection of an appropriate domain for an expert system", AI Magazine, Summer 1985, 26-30.

Prerau, D.S. (1987), "Knowledge acquisition in the development of a large expert system", AI Magazine, Vol. 6(Summer), 43-51.

SANZ (1986), "DZ4203:1986 General Structural Design and Design Loadings for Buildings", Standards Association of New Zealand, Wellington, New Zealand.

Smith R.G. and Baker, J.D. (1983), "The Dipmeter Advisor system - a case study in commercial expert system development", Proc of the 8th IJCAI, 122-129.

Smith, R.J.(ed) (1986), "Steel Structures for Seismic Safety", Interntaional Iron and Steel Institute, Brussels.

Whitney, R.S. (1987), "Knowledge-based systems for building technology: strategic aspects", Proc. of the 2nd New Zealand Conf. on Expert Systems, Auckland, New Zealand, Feb, 5-11.

Copy 1

Copy 1

B17182
0025521
1988
The development of an expert system for wall braci

Replacement copy

BUILDING RESEARCH ASSOCIATION OF NEW ZEALAND INC.
HEAD OFFICE AND LIBRARY, MOONSHINE ROAD, JUDGEFORD.

The Building Research Association of New Zealand is an industry-backed, independent research and testing organisation set up to acquire, apply and distribute knowledge about building which will benefit the industry and through it the community at large.

Postal Address BRANZ, Private Bag, Porirua

ISSN 0111–7459