THE SELECTION OF PROPERTY MANAGEMENT SOFTWARE

S. van den Berg and C.E. Cloete

Department of Construction Economics
University of Pretoria, South Africa
ccloete@postino.up.ac.za

ABSTRACT

Property management software packages have become essential tools in the management of properties and related facilities. In this paper, criteria for the comparative evaluation of property management software packages are developed and the various software packages available in South Africa are evaluated against these criteria. Recommendations are made as to the appropriateness of the various software packages to the needs of the user.

OPSOMMING

Sagtewarepakkette vir eiendomsbestuur is 'n integrale hulpmiddel in die bestuur van eiendom en verwante fasiliteite. Hierdie artikel het die ontwikkeling van kriteria vir die vergelykende evaluering van eiendomsbestuursagteware ten doel en die onderskeie sagtewarepakkette wat in Suid-Afrika beskikbaar is, word met behulp van hierdie kriteria geëvalueer. Aanbevelings word gedoen ten opsigte van die toepaslikheid van die onderskeie sagtewarepakkette vir die gebruiker se behoeftes.

1. INTRODUCTION

Technology has pervaded all levels and facets of the modern day business environment. The property management industry has not escaped this trend. The speed and accuracy of computerised property management software in reviewing, sorting, filtering and reporting relevant information make it an invaluable tool. Computerised systems make the processing of massive amounts of data easier and quicker. This leads to cost savings through less repetitive work, improved cashflow, accurate information and quicker client response with improved services (Dixon [13]).

2. PURPOSE

The purpose of this paper is two-fold. First, a study was done to determine the factors to be considered when selecting property management software. The systematic approach to a software selection and implementation process in general is discussed. This approach is then applied specifically to property management software, to give the potential buyer insight into the questions to be asked when determining his requirements. The various categories to be used when evaluating computerised property management software are discussed, viz. general, technical, functional, reporting, help functions and operator manuals, user interface, other categories and pricing.

The second part of this paper is devoted to the evaluation of computerised property management software available in South Africa. Suppliers of property management software were visited and demonstrations of their software attended. The various software packages were evaluated for the categories mentioned above. A summary of the findings is listed in the concluding paragraph.

3. LIMITATIONS

This study only evaluated software that is **available and supported** in South Africa. The software evaluated is either developed and supported by companies based in South African or by overseas companies with a local office that supply, install, support and provide training. Although numerous software packages can be purchased from overseas, the nature of property management software is such that implementation, support and training form a major part of the complete package. These critical factors are not available from software available from overseas and such packages were therefore not included in this evaluation.

The software packages considered here are all stand-alone packages and do not form an add-on module of a larger software package.

The purpose of this study is not to arrive at an absolute ranking of the software package from best to worst. This would not be recommended because it would not take into consideration the needs and requirements of the potential user. Each of the software packages has its own strengths and weaknesses and some focus more on specific aspects of the property management arena. This paper rather attempts to guide the potential user, in determining his own unique facility/property management requirements, to ask the correct questions and then to make a decision based on the information presented here, as well as on his/her own analytical process.

4. DEFINITIONS

In this paper, property management refers to the management or administration of a property or portfolio of properties to achieve the goals or objectives of the property owner.

Property management software is software that aids the manager, managing or administration company in achieving the goals or objectives of the property owner by, inter alia, automating mundane tasks and providing management information.

5. SOFTWARE SELECTION PROCESS

The selection of property management software is not much different from the selection of any other type of specialised software, be it CAD (computer-aided design), ERP or hardware simulation programmes. Certain basic rules apply to the selection of any type of software. A systematic and strategic approach to software selection is proposed by Dixon [13] – figure 1.
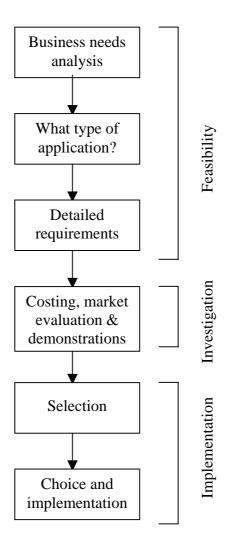

Figure 1: Stages in the selection process

As shown in Figure 1 the selection process comprises three main stages, namely:

- Feasibility
- Investigation
- Implementation

Feasibility

Feasibility involves a full analysis of applications in order to quantify the cost and benefits of a new software system. "*The prerequisite for success is an exact definition of the tasks to be performed and the objective to be achieved. The essential ingredient is the specialist knowledge of the decision maker: knowing **what** needs to be done and **how**. Once initiated the analysis will also examine the current running of the organisation including*

- *detailed investigation of existing information systems and professional activities in the firm*
- *shortcomings and problem areas, which could include data duplication, excessive information costs and lack of information for key areas of management*
- *human and organisation costs of the system, including training, staffing and ergonomic studies.*" (Dixon [13])

Keep in mind that a computerised system will not clean up a chaotic manual or hand-driven system. In most cases, a computerised system will add to the chaos as the saying warns: "Garbage in, garbage out." Certain procedures, efficiencies and work structures must be in place in order to fully benefit from a computerised system.

It is very important to consult the end user and staff using the system during this analysis stage. They are the people using the existing system and they will provide invaluable information as to what processes and procedures work or how it should function in the computerised system. Also keep in mind that the computerised system may require changes to the way the business is currently being run. "Software vendors typically draw ideas from a large pool of customers. This lets them refine how things are done by selecting those processes that result in greatest efficiency. This process can be to your benefit if you are willing to change your present operations. Here are a few phrases you might catch yourself saying during this process that can cost you time and money: 'We have always done it that way; We have to do it this way; We can't afford to change right now.' The primary benefits of a computerised accounting and management system are efficiency gains from automating routine tasks, instant access to information and the ability to perform tasks that were previously considered too impractical" (Pedersen [25]).

At the detailed requirements list stage, a prioritised list is very helpful. List the functions that are required, need improvement or could be made more efficient. The next step is to prioritise this comprehensive list into "must haves", "desired" and "would be nice". With this in hand the next step of investigation can be undertaken.

Investigation

The investigation phase starts off with the consideration of the main options for acquiring software, namely:

- In-house program
- Off-the-shelf program (not customisable)
- Off-the-shelf program (configurable/customisable)
- Custom-built application

A certain level of computer awareness is needed to formulate a realistic software strategy. Moreover, cost will need to be fully investigated, including the cost of purchase, upgrade cost for new versions of the software and training. The above options all have their advantages and disadvantages that must be taken into consideration.

In-house program. This is designed and programmed by the user himself and usually by someone with basic programming skills. It involves keeping records and doing calculations on a spreadsheet or small database type program. The range of information kept is generally limited. Functionality is limited, the user interface very basic and the program is not user friendly and usually only used by one or two persons in the company. Advantages are that the program development is cheap and the user can alter and add to the application easily, immediately and cheaply. These programmes are suitable for small companies with limited requirements.

Off-the-shelf programs (not customisable). This refers to an application designed and written by professional programmers which any user can buy and use as is. There are usually limited or no customisable features and the user has to make do with the features and reports bundled with the program. These types of program are written to fulfil a general need rather than a specific requirement and they are sometimes too general to meet the exact needs of certain companies. Advantages of these programs are their moderate cost, they generally are easy to learn, they have been tested for bugs and errors and are usually user friendly. These programs are suitable for small to medium companies with larger requirements.

Off-the-shelf programs (configurable/customisable). These types of program are similar to the ones described above but with the added advantage that they can be tailored and customised to a user's specific needs and requirements. Not only can the screen layout and reports be tailored to the user's needs but unique functionality can be added to the basic programme structure. Some of these programmes are modular in design which enables the user to add the required different modules to the basic package. The user interface, help functions, user and operating manuals are of excellent standard. These types of program also usually require a dedicated implementation team drawn from the supplying company or its reseller and personnel from the client company. The costs of buying and implementing these programs are usually substantial and implementation time can range from months to years, depending on the size of the system. These programmes are suitable for medium to large companies with specific requirements.

Custom-built application. Custom-built applications are designed and built to the customer's own and specific needs and requirements. This can be done either in-house or be

contracted out to a particular software house. For custom-built applications the user prepared detailed specifications of all the functional, technical, user interface, reports and hardware requirements. This can be done by the company's own personnel (IT department) or a third party consultant can be appointed. The programming department or software house then develops the program to these exact needs and requirements. The result is a program that complies fully with the client's needs and is unique to the client's company. The disadvantages are cost and long development phases. Also, when company procedures and practices change, sections of the application may have to be rewritten. These types of application are suitable for large companies or when a unique requirement exists for which no off-the-shelf application is available.

The following is a concise list of questions to be asked and items to be considered during the investigation phase:

- How does the software fit in with existing hardware and network structure?
- What changes have to be made to existing hardware?
- Request a demonstration of the software.
- Speak to existing customers and users to find out about strengths and weaknesses.
- Investigate the quality of the operating and user manuals. Poor documentation often means poor software.
- What is the database structure and how does it fit in with existing structures? (Oracle, SQL, etc or proprietary)
- How does the software match up with your priority list?
- Integration of software into existing structures and other software currently used such as financial software.
- Report capabilities (systems reports as well as own report-writing tools)
- User interface and the ease of use.
- What technical and user support is available?
- Quality of training.
- Password and security features.
- Future development by software house.
- Calculate total cost of ownership (TCO)

The last item mentioned is a very important step and is sometimes neglected to the detriment of the end-user. The TCO enables the decision-maker to assess the total cost of selecting, implementing and maintaining a system.

In general, most TCO analyses are divided into one-time costs and on-going costs. One-time costs typically include software, hardware, outside services, internal project team resources, training, marketing and other project expenses required to select, implement and roll out the application. On-going costs typically include hardware and software maintenance fees, support costs, and on-going technical and end user training costs.

One of the more difficult aspects of calculating TCO is determining costs for items that have not historically been tracked by real estate companies. For example, external (i.e. maintenance) and internal (i.e. technical resources) costs are incurred for each piece of hardware associated with the application. There is also an incremental cost to support every

new server, and a cost per desktop that needs to be factored in as well. Lastly, there are also operational and technical costs related to supporting and maintaining a software package.

New information that was not captured previously may need to be entered by the properties and/or corporate offices. Nonetheless, TCO determination is a lengthy process that will need additional refining over time. However, this is an important process that must occur in order to fully understand the cost and value of the project." (Reummler [27])

Implementation

Implementation starts off with the final selection process. Reduce the initial list of possible suppliers to three or four. Evaluate the priority list again against the supplier's software capabilities and select the most favourable supplier in terms of the TCO considerations discussed above. Once the supplier is selected, the final implementation plan is put on the table and implementation carefully managed until completion.

6. PROPERTY MANAGEMENT SOFTWARE SELECTION

Like all spheres of management, effective property management requires actions at three different levels, namely basic administration (or supervisory level), general management (or functional level) and strategic management, as indicated in Figure 2 (Dixon [13]).

Database and system considerations

In practice, there often are separate databases for the property management software. Because of the duplication of data which inevitably occurs in such non-integrated systems, these systems tend to be wasteful and costly. Ideally, the database structure should be a relational database management system such as Microsoft SQL, Oracle or Sybase. Relational database systems enable additional modules to be added as required, while different departments can access and update data from a shared database. In addition, access to confidential data can be controlled through a password-controlled hierarchy of access types. Logging reports and audit trails will facilitate control.

The database structure should therefore make provision for the following minimum information:

Building or Property: building information, lease agreement administration, credit control, cash flow control, compatibility with accounting principles and practices and existing systems, accurate bookkeeping, owner reporting and identifying of key performance indicators.

Tenant: general tenant information (name and contact details), finding space for a tenant, accurate rent billing and collection, handling of payments, accounting and general ledger (GL) functions.

Space or Unit: Unit no, available date, responsible date, unit size, letting rate, remarks; description; key no, parking, VAT information finding a tenant for a space, space management, maintaining the spaces individually as well as collectively as well as the

common spaces. Managing cleaning contracts, security contracts, help desk facilities, asset register, creation of work orders and equipment maintenance to name but a few. This is all done to assist the tenant in focusing on his core business.
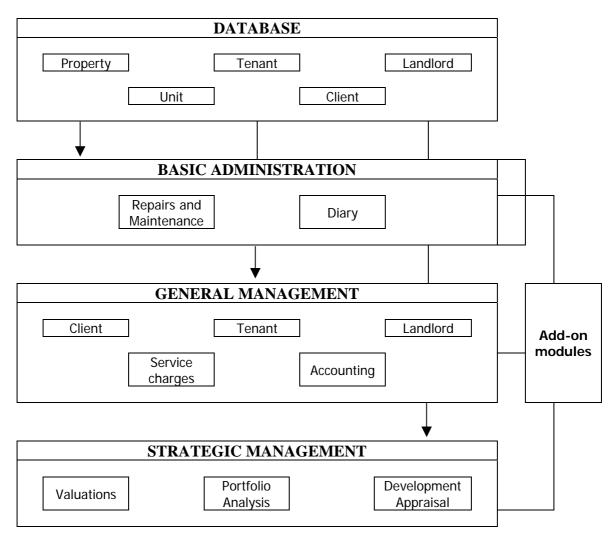
```
┌─────────────────────────────────────────────────────────────┐
│                        DATABASE                               │
│  ┌──────────┐        ┌──────────┐         ┌──────────┐        │
│  │ Property │        │  Tenant  │         │ Landlord │        │
│  └──────────┘        └──────────┘         └──────────┘        │
│        ┌──────────┐           ┌──────────┐                    │
│        │   Unit   │           │  Client  │                    │
│        └──────────┘           └──────────┘                    │
└─────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────┐
│                   BASIC ADMINISTRATION                         │
│   ┌───────────────┐              ┌──────────┐                  │
│   │  Repairs and  │              │  Diary   │                  │
│   │  Maintenance  │              └──────────┘                  │
│   └───────────────┘                                           │
└─────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────┐
│                   GENERAL MANAGEMENT                           │
│  ┌──────────┐        ┌──────────┐         ┌──────────┐        │
│  │  Client  │        │  Tenant  │         │ Landlord │        │
│  └──────────┘        └──────────┘         └──────────┘        │
│      ┌──────────┐              ┌────────────┐                  │
│      │ Service  │              │ Accounting │                  │
│      │ charges  │              └────────────┘                  │
│      └──────────┘                                             │
└─────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────┐
│                  STRATEGIC MANAGEMENT                          │
│  ┌────────────┐      ┌────────────┐       ┌──────────────┐    │
│  │ Valuations │      │ Portfolio  │       │ Development  │    │
│  └────────────┘      │  Analysis  │       │  Appraisal   │    │
│                      └────────────┘       └──────────────┘    │
└─────────────────────────────────────────────────────────────┘

                                          ┌──────────┐
                                          │ Add-on   │
                                          │ modules  │
                                          └──────────┘
```

Figure 2: Property Management Levels

Landlord, Client or Body Corporate: general details (name and contact details), shareholding if applicable.

Basic administration (supervisory level)

Repair and maintenance schedules are required by the property manager, as well as a diary to "flag" important dates for tenant's works, rent review and lease renewal dates. A good software program should also provide for a forward planning facility.

General management (functional level)

Dixon (13) aptly summarises the requirements at this level as follows:

"In terms of accounting procedures, the main property management related tasks will comprise

- *Rent invoicing and income connection*
- *Recovery of expenditure*
- *Disbursements/outgoings*
- *Service charge costing and apportionment*
- *Client and tenant accounts*
- *Report production.*

Any rent invoicing system should record amounts owing from tenants quickly and accurately and bill them accordingly. Receipts will need to be processed quickly and credit control systems maintained. Rent demands and accounts should be easily accessible as should rent apportioned over periods not concurrent with a normal rent review period. Interest on unpaid rent should be calculable and a stop on rent collection made if necessary. Full analysis of rents, classified by tenant, property or client should also be possible. Service charge accounting is often provided as a separate module. This will need to cater for multi tenanted buildings where perhaps some tenants do not contribute to some services. Separate schedules may well need to be set up in such cases. In addition, a full analysis of property expenditure, service suppliers, tenant expenditure, service charges, wages and salaries, and VAT on expenditure should be possible".

Strategic management

Lastly, a software programme should provide the information necessary to make strategic decisions. Such decisions include the performance and valuation of individual properties and property portfolios, as well as development appraisals. Features that are required comprise, among other things, tenancy and tenure details, the calculation of yields and profitability, discounted cash flows, cost, financing, tax implications and the valuation of both freeholds and leaseholds. All these should be supported by good menu-driven features and help facilities.

Technical aspects and user interface requirements are discussed below.

7. WHAT PROPERTY MANAGEMENT SOFTWARE SHOULD DO

The above four paragraphs describe in detail what a property management software application should be able to do. It can all be summarised in the following: maintain, sort, summarise and display data, automate and control daily tasks, maintain historical data, improve planning and labour productivity.

8. WHAT PROPERTY MANAGEMENT SOFTWARE WILL NOT DO

Although a helpful aid to the property management environment, a property management software application cannot do everything. Some of the things that a property management software application cannot do are: replace a property manager, replace common sense planning, assign work and do site inspections and bring order to chaos (garbage in, garbage out).

9. THE EVALUATION OF SOFTWARE

The software evaluation can be divided into eight main categories, namely general, technical, functional, reporting, help functions and user/operator manuals, user interface, other and pricing structure.

General

General gives an overview of the software and its uses and some background on the company that is developing it. It also mentions other software packages that might relate to the property management software.

Technical

This section discusses the minimum hardware requirements as well as hardware configurations in which the program will operate. The programming language in which the user interface was designed, as well as the database standard used. Other hardware or software required is also mentioned.

Other aspects that should be considered are: Hardware and software requirements and the interfacing with existing hardware/software and network structures. The user should consider his existing corporate network infrastructure, network protocol and existing database format (SQL, Oracle etc.) and include the IT department in his decision-making process if applicable.

Security aspects provided to aid in preventing unauthorised access to the system and data are also discussed in this section.

Functional

In this section, the potential user should critically evaluate the software to determine if it complies with his/her requirements. The available functions should be able to satisfy his specific needs. If for instance accounting and general ledger modules are required, this should specifically be demonstrated and assessed.

Reporting

A database that just contains data means nothing if that data cannot be extracted efficiently and combined with other information to form effective management decision tools. Reports are tools that can be used to manage more effectively, reduce and cut costs and in doing so, enable the manager to gain a competitive edge on the opposition.

The basic reports that are included in the system, as well as the possibility of writing one's own reports, are discussed in the paragraph on reporting. The methods and ease of specifying additional criteria are also discussed.

The potential user should determine if the built-in reports are adequate to fulfil his needs and if they can be customised to his requirements. If additional reports are required, how easy is it

to create these reports? Some software packages feature built-in report-writing tools while others rely on third-party software to achieve this.

Help functions and user/operator manuals

Users need assistance at different levels. Because the user cannot ask someone all of the time, he is reliant on the help and manuals offered by the system.

Well-designed help files correspond to the five basic questions that users usually ask (Brenda [9]):

a. *Goal-oriented:*   *"What kinds of things can I do with this program?"*
b. *Descriptive:*   *"What is this? What does this do?"*
c. *Procedural:*   *"How do I do this?"*
d. *Interpretive:*   *"Why did this happen?"*
e. *Navigational:*   *"Where am I?"*

For example, "about boxes" are one way of addressing the needs of questions of type (a). Questions of type (b) can be answered with a standard "help browser", "tool tips" or other kinds of context-sensitive help. A help browser can also be useful in responding to questions of type (c), but these can sometimes be more efficiently addressed using "cue cards", interactive "guides", or "wizards" which guide the user through the process step-by-step. The fourth type (d) has not been well addressed in current applications, although well-written error messages can help. Type (e) questions can be answered by proper overall user interface design, or by creating an application "roadmap."

The effective use of these help guidelines are discussed as well as the availability of operator and user manuals.

User Interface

Because the menus and screens are the first and foremost thing that a user sees and experiences when using software, the user interface is just as important as the proper functioning and the features offered by the program.

Users often evaluate software on what they see, the feel and experience the software creates and how easy it is to navigate around and access the functions they mostly use, and not so much on optimum database design and other technical functions with which they never come into contact.

The next two paragraphs are a summary of tips to user interface designers from Ambler [2]. Potential buyers can also use these tips to evaluation the user interface:

General Guidelines

- Be consistent in a user interface. Set the standards from the start and stick to them.
- Choose industry standards so as to increase the chance that your applications will look and feel like other applications developed externally to your organisation. This does not mean

that individuality and creativity needs to be ignored but if the program does not function or react in a way the user is used to, it is often perceived as too difficult to use.

- Support both novice and expert users. Make it easy for new users to use the program but provide shortcuts for experienced users.
- Word text consistently, positively and in full English. Explain errors made by the user in enough detail for him to correct it.

Screen Design

- Navigation both between screens and within screens is important. Make use of tabs to move between screens of the same subject. Toolbars should be consistent with the same functions in the same location. Dynamic toolbars, toolbars that grey out functions not available or add functions depending on the active screen, should be used with some consistency and provide excellent feedback to the user.

- Use colour appropriately. Colour can be used to enhance and provide additional functionality to your database application if used cleverly and a few basic ground rules are followed. Colours could be used effectively to indicate to the user primary key fields, compulsory fields, read only fields, select from list fields, enabled fields and disabled fields. This immediately indicates to the user what fields must be completed, what is optional, what cannot be changed and what is selectable from a list or dropdown box.

  The problem with using colour is that the user may be colour blind. If colour is used to highlight something on the screen the colour blind user might not see it. In this case a secondary indicator in the form of a flashing icon or symbol must be used to attract the user's attention. A second problem is that colour does not look the same on different computers and/or monitors. It is therefore advisable to allow the user the option to enable or disable the colour settings and to custom set the colours used.

- Follow the contrast rule – put dark text on light backgrounds and light text on dark backgrounds.
- Use fonts sparingly and consistently. This includes the use of size, style (bold, italics, underlining etc), typeface or colour.
- When items are unavailable grey them out, don't remove them if you want the users to form accurate mental models.
- Use non-destructive default buttons. The default button shouldn't do something that is potentially destructive; such as delete a field or record.
- Left justify edit fields and right justify their labels.
- Right justify integers, decimal align floating-point numbers, and left justify strings.
- Don't create busy/crowded screens.
- Use group boxes and white space to group logically related items on the screen.
- Open windows in the centre of the action. When a user double-clicks on an object to display its edit/detail screen then his or her attention is on that spot. Therefore it makes sense to open the window in that spot, not somewhere else.

Other

Other items discussed include features such as Web enablement, CAD interface, client base, new developments and other special features not discussed elsewhere.

10. CONCLUSION AND RECOMMENDATIONS ON SELECTION PROCEDURE

Companies that have not embraced technology are finding it gradually more difficult to increase efficiencies and achieve a competitive advantage in the marketplace. Even more companies have embraced technology blindly and without proper planning and have paid the price for it. "Both long-term strategic planning and short-term tactical planning are key to successful leveraging technology within today's real estate companies." (Ruemmler & Morey [27])

Before any investigation of property management software applications can commence, a company must develop and implement a strategic information systems plan (SISP). The SISP must be given the same consideration and level of importance as the company's strategic business plan. Only once the SISP is in place can the feasibility, investigation and implementation processes start.

When used strategically, technology will be the important difference that separates a company from the competition. "The overriding consideration in selecting a software package is finding one compatible with a company's long-term business goals and the need of tenants and clients." (Morey [24])

11. EVALUATION OF PROPERTY MANAGEMENT SOFTWARE PACKAGES

The property management software packages currently (March 2003) available in South Africa are:

- MRI (Management Reports International)
- MDA Property Management by MDA
- Manhatten by Raindrop SA
- PropPro and PropPro Lite by Nicor
- Propac by Propsys Solutions
- Premis 2000i by the CSIR
- Genes1s Facilities Management by Genes1s Concepts

The main characteristics of these software packages investigated in this study are summarised in table 1 below, to assist the potential user of any of the packages in making a selection.

Price Structure

All companies viewed pricing as a sensitive issue and some were not willing to provide detailed pricing structures. Taking the pricing structure into consideration at the time of the survey (March 2003), the various software packages have been classified according to their suitability for three main user categories: the small user, the medium user and the large user.

	MRI	MDA	PREMIS 2000i	PROP-PRO	PROP-PRO LITE	MAN-HATTEN	PROPAC	GENES1S
TECHNICAL								
Security levels: Login / profiles	x	x	x	x	x	x	x	x
Security levels: Field Level	x					x		
Database								
Microsoft SQL	x	x	x			x	x	x
Oracle						x		
Microsoft Access					x			
Other				progress		sybase etc	sybase	
Operating Systems								
Windows	x	x	x	x	x	x	x	x
Linux				x				
Unix				x		x		
Other								
FUNCTIONAL								
Property Info Screen	x	x	x	x	x	x	x	x
Tenant Info Screen	x	x	x	x	x	x	x	x
Unit Info Screen	x	x	x	x	x	x	x	x
Landlord/Client Info Screen	x	x	x	x	x	x	x	x
General Ledger Module	x	x		x		x	x	
Diary Function	x					x	x	
FM/Maintenance Management Module	x	x	x	x		x	x	
Key Performance Indicators		x				x		
REPORTING								
Built-in	x	x	x	x	x	x	x	x
Own Reports	x	x	x	external s/w		external s/w	external s/w	
HELP & MANUALS								
On-line help	x	x	x	x	x	x	x	x
Field sensitive help	x	x	x	x	x	x		
User manuals	x	x	x	x	x	x	x	x
On-site training	x	x	x	x	x	x	x	x
Telephone support	x	x	x	x	x	x	x	x
Web based support		x						
OTHER								
Web based modules	x			x		x		
USER INTERFACE								
Consistent throughout	x	x	x	x	x		x	x
Use of sound								
Use of colour	x		x	x	x		x	
USER CATEGORIES								
Small (< 1000 accounts)					x		x	
Med (1000 to 20 000 accounts)	x	x		x			x	x
Large (>20 000 accounts)	x	x	x	x		x		

**Table 1: Main characteristics of property management software
available in South Africa (March 2003)**

The **small user** will typically manage not more that 1 000 tenant accounts. This type of user is usually small leasing or rental agencies or estate agents. The types of property are usually residential houses, flats, townhouses or simplex or duplex units. The **medium user** can manage up to 20 000 leasing accounts. This type of user manages typically blocks of flats or townhouse developments or even small neighbourhood or strip type shopping centres or commercial properties. The **large user** usually manages large shopping centres and commercial properties. The properties are either owned by them or managed on behalf of major property owners and can amount to several billion rands' worth. The number of tenant leases is more than 20 000 and can as high as 200 000.

12. CONCLUSIONS AND RECOMMENDATIONS ON SOFTWARE PACKAGES

Given the different features of the various software packages and the varying requirements of the user, a number of recommendations can be made.

If the user is looking for scalability and full customisation, the MRI and Manhattan software will suit his needs. Both packages feature all the required property management modules, as well as various financial modules.

More affordable, but with all the required property management features as well as a fully integrated accounting and general ledger module, are the MDA Property Manager by MDA, PropPro by Nicor and Propac by Propsys Solutions.

Entry level software is provided by Nicor with its PropPro Lite version. However, this package offers no financial module. The ProPac software offers entry level pricing, but with a fully integrated financial module especially suitable for residential letting agents.

For a maintenance management package, Premis 2000i by the CSIR offers unrivalled capabilities.

If the user requires integration with CAD software coupled with property and asset management the products from Genes1s and Manhattan should be considered.

This **small category user** is usually a small leasing or rental agency or estate agent. The types of lease are residential houses, flats, townhouses or simplex or duplex units and there are no more than 1 000 accounts. For this type of user, the PropPro Lite and Propac packages are suitable.

Propac, MRI, MDA and PropPro software will suit the **medium category user**. This type of user manages blocks of flats, townhouse developments or even small neighbourhood or strip type shopping centres or commercial properties with up to 20 000 accounts.

The **large user** with more than 20 000 accounts and as many as 200 000 accounts should look at the software provided by MRI, MDA, PropPro and Manhatten, as well as Premis, for maintenance purposes.

While these recommendations are believed to be valid at the time of writing, it must be borne in mind that most of the software packages are continuously being upgraded and that pricing

structures may change on short notice. Nevertheless, application of the above guidelines should be useful for a user confronted by a sometimes bewildering number of alternative solutions to his property management needs.

13. REFERENCES

[1]   **Alport, R.**, November 2002. "Interview and product demonstration", Gens1s Concepts.
[2]   **Ambler, SW.**, 2000. "User Interface Design: Tips and Techniques", http://www.ambysoft.com/userInterfaceDesign.pdf
[3]   **Berkun, S.**, July/August 1999. "The Importance of Simplicity – Create Ease of Use Without Losing Power", http://msdn.microsoft.com/ui/.
[4]   **Berkun, S.**, May/June 2000. "Fitts's UI Law Applied to the Web", http://msdn.microsoft.com/ui/.
[5]   **Berkun, S.**, September/October 1999. "How to Avoid Foolish Consistency", http://msdn.microsoft.com/ui/.
[6]   **Berkun, S.**, March/April 2000. "Why Good Design Comes from Bad Design", http://msdn.microsoft.com/ui/.
[7]   **Berkun, S.**, January/February 1999. "Why Are Good User Interfaces So Hard to Make? Three Insights into Good Design", http://msdn.microsoft.com/ui/.
[8]   **Berkun, S.**, March/April 1999. "Making Usable Products: An Informal Process for Good User Interfaces", http://msdn.microsoft.com/ui/.
[9]   **Brenda, L.**, 1991. "The Art of Human Computer Interface Design", Addison-Wesley.
[10] **Cato, W.W. and Mobley, R.K.**, 2002. "Computer-managed maintenance systems: a step-by-step guide to effective management, labor, and inventory". Second edition, Butterworth-Heinemann.
[11] **Cloete, C.E.**, 1994. "Property Management Part 1 Volume 1". SA Property Education Trust, Sandton, p. 133.
[12] **Cloete, C.E.**, 1994. "Property Management Part 2 Volume 2". SA Property Education Trust, Sandton, p. 322-325.
[13] **Dixon, T.J.**, 1998. "Selecting property management software: problems and solutions", Property Management, Vol. 11, No. 2, p. 155-167.
[14] **Fricke, F.**, 1992. "Facilities Management", Architectural Science Review, September 1992, No. 3, p. 81-82.
[15] "From zero to income & expenditure statement in 30 minutes", The Property Professional, April/May 2003, p. 52.
[16] "Handbook of systems engineering and management". Edited by AP Sage and WB Rouse, 1999, Chapter 17, John Wiley & Sons, INC.
[17] **Keet, D.**, 2002. "The $999.99 Property Management System", Article forwarded via e-mail.
[18] **Liebenberg, P.**, Interview and product demonstration, CSIR Boutek, August 2002.
[19] Little, D., Jan/Feb 2000. "Software, Choosing Systems for the Next Century', Journal of Property Management, p. 60-65.
[20] **Marshall, B.**, Interview and product demonstration, Manhattan Africa, November 2002.
[21] Microsoft Corporation, October 2000. "UI Guidelines vs. Usability Testing", http://msdn.microsoft.com/ui/.

[22]   **Morey, S & Giudice, D.**, May/June 1995. "Survey of Property Management and Accounting Software Part I", Journal of Property Management, Vol. 60, No. 3,   p. 58-59.

[23]   **Morey, S & Giudice, D.**, May/June 1995. "Survey of Property Management and Accounting Software Part II", Journal of Property Management. Vol. 60, No 4,   p. 56.

[24]   **Morey, S.**, 1999. "Survey of Property Management and Accounting Software 99', Journal of Property Management, p. 80,90,91.

[25]   **Pedersen, H.**, March/April 1995. "An Organized Approach to Software Selection", Journal of Property Management, p. 62-64.

[26]   **Potgieter, W.**, Interview and product demonstration, MDA Property Systems, September 2002.

[27]   **Ruemmler, C & Morey, S.**, 2001. "Survey of Property Management and Accounting Software 2001", Journal of Property Management, p. 49-67.

[28]   **Silverman, P and Dario.**, Interview and product demonstration, Nicor, September 2002.

[29]   **Steincke, S.**, Interview and product demonstration, Management Reports Inc. (MRI), August 2002.

[30]   **Talin**, 1998. "A Summary of Principles for User-Interface Design", www.sylvantech.com.

[31]   **Tryhou, RLH.**, 1998. "The selection of property management software in South Africa". Unpublished MSc assignment, University of Pretoria.

[32]   **Van Niekerk, D. & Vermeulen, M.**, Interview and product demonstration, Propsys Solutions (Propac), September 2002.