Written by Alan Rodger, March 2003 TA000185ADT

Technology Infrastructure

Butler Group Subscription Services

Application Development
TECHNOLOGY AUDIT

Lazy Software
Sentences 3.0, with LazyView, and Lazy Analytics

Abstract *Sentences is an Application Development Environment (ADE) tool that is based upon a radically different database model, allowing development of applications and their underlying data models to take place simultaneously. The database technology is powerful in a business sense, as it reflects the real world in its treatment of entities and their relationships, and allows applications an insightful view of the data. The LazyView tool adds the capability to link relational, or other non-Sentences databases, to applications, and to define queries.*

Lazy Software's departure from the relational database model can provide benefits of high added value to information, and rapid application deployment, at reasonable cost. These benefits are particularly accentuated when data structures are complex, or unstructured. The product also allows the costs of application maintenance to be reduced, as database definitions are not hard-coded into programs.

An evaluation is available via the company's Web site.

STRENGTHS	WEAKNESSES
• Removes constraints of the relational model on development. • Reduces cost of application delivery and maintenance. • Enables complex or unstructured data to be analysed.	• Lazy Analytics tool is less mature than the other Sentences products. • Pervasiveness of the relational model inhibits uptake of new paradigm.

FUTURE POTENTIAL

Some vertical solutions that could demonstrate the capabilities of Sentences might prove an incentive to prospective customers.

Butler Group

FUNCTIONALITY

Product Analysis

The Relational Model of data, the standard database architecture underlying most modern applications, has distinct disadvantages. Relational databases store each different type of data in a separate table, each of which has its own unique format. Application programs must be hard-coded to fit the tables, and any change to the structure of the data means that the programs must also be changed.

Lazy Software's product group is based on its Associative Model of Data (AMD), in which a single, generic storage structure contain all types of data, and the relationships between data elements. As application programs can obtain information about data structures and relationships from the database itself, they do not have to be amended when those structures and relationships change. Such programs are truly reusable, dramatically reducing the cost of application development and maintenance.

The Sentences product is so named because its storage of associations between entities matches the 'subject–verb-object' syntax of a simple English language sentence. The 'verb' is the association between the two entities, the subject and object. As complexity increases, one of the entities might itself be an association, e.g. ([Flight BA123 arrives at 16:15] on Saturday). Associations and entities are stored generically on the Sentences database, and are used to represent metadata definitions of associations ('person orders book'), as well as actual occurrences ('Simon orders Lord of the Rings'). Complex data behaviours, such as a person (who is an entity) having various roles as customer, employee, spouse, or shareholder, can be simply represented as the person's various possible associations with other entities.

The user interface is Sentences Explorer, which allows a non-technical user to easily apply real-world understanding to the task of defining data structures within Sentences. The same user interface allows entry of data values into the database, and at each stage the tool prompts the user helpfully. Many applications elements are automatically generated during database definition, but more may be added by use of the Sentences Java Application Programming Interface (API). LazyView, a query definition tool that also allows database aggregation, and Lazy Analytics, an On-line Analytical Processing (OLAP) tool for the Sentences database, complete the product set. Sentences also includes an API from which external applications can query its databases, via XML.

Some features mentioned during the remainder of this document are denoted as being available from v3.5, a major release of Sentences due in April 2003.

Product Operation

Sentences Explorer provides the user interface for database definition, and database population, and may be used to deploy an application (in which case, the facilities to amend the database definitions may be excluded from a user's view of Sentences Explorer, if required). The tool has a similar look and feel to Microsoft's Windows Explorer, with left- and right-hand panes within the main work area.

The user enters entity and association definitions in the left-hand pane, using browser buttons and natural language. These facilities contrast with relational database definition, which has to be conducted by an architect or administrator before a programmer may use the database. Entity definitions can incorporate standard data types, such as date, or monetary, in which case appropriate validation or selection can be invoked.

Entities may be defined as sub-types of other entities, in which case an entity's properties are inherited by its sub-type. Association definitions incorporate the source (an existing entity, or association), verb, and target (another existing entity, or association). The definition of the association's verb incorporates the specification of its inverse, in order to allow navigation in either direction.

While the foregoing two paragraphs describe the total complexity of database definition within Sentences, it should also be stressed that the tool has sufficient power and detail to deal with any extent of requirements, up to and including corporate scale. For anybody familiar with having to build database schemas and then write applications programs on top of those schemas, it seems an impossible exaggeration to describe the simplicity and ease of use that Sentences provides.

The right-hand pane of Sentences Explorer is used to allow the user to view, or populate, the database. For any entity, or association, Sentences uses one of a number of 'omnicompetent programs' to present a form to the user, which shows an appropriate view of the data type selected. For example, if the user has chosen to populate a date field, the form will allow selection of a value from a calendar. In other cases, where the user could require a new or existing value, the form would present the database's existing values, with an optional entry field for a new value. Associations are also presented via appropriate views, e.g. as tables. Data appropriate to sub-types of entities are viewed via tabs of the form for that entity.

The v3.5 release of Sentences will include a Diagram Editor, giving the capability to create, edit, save, and print diagrams of schema and data items.

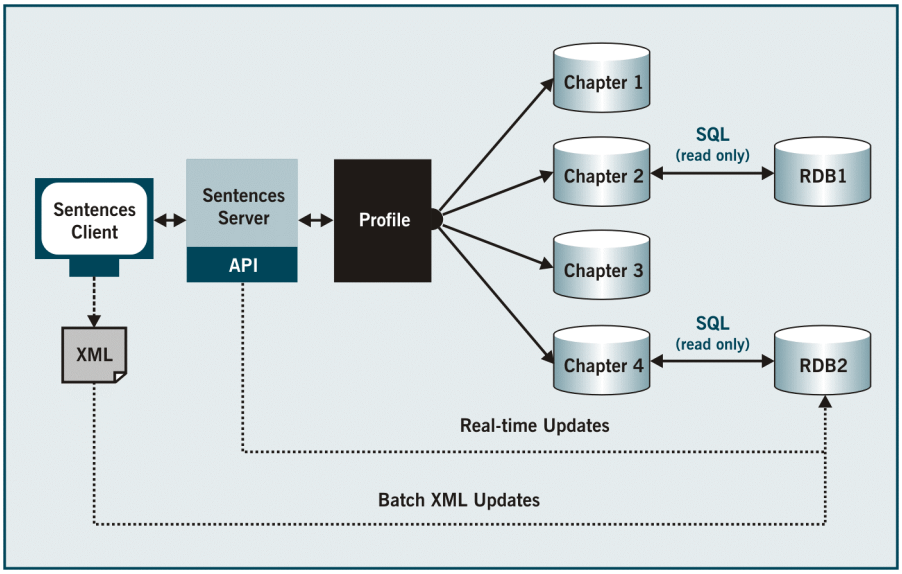
It is a crucial feature of Sentences that any unique value is held only once on the database, although this is transparent to the user. For example, if there existed a value 'John' of a Name entity, all persons with that name would refer to that stored value as an association, allowing Sentences to save storage of any duplicate values. If a user were to change the name for any person with that association, the database would add the new value and manage the change of associations without the user being aware. The benefit of this feature is the avoidance of storing any duplicate value, with a consequent cost saving. The associative database model also saves space by storing associations only where they occur for an entity.

Sentences databases are stored in Chapter files, these Chapter files being configurable to make up a Profile, which defines a specific database, or set of databases that are available to a user. The v3.5 release of Sentences will allow a password to be set on specific Chapter files, to restrict access. A Profile can define a merge of Sentences databases to take place, and specify generic rules options to control, for example, the treatment of duplicate records. Profiles are used in a product characteristic called Application Feature Partitioning, in which the Profiles available to individual users and applications grant different access rights to parts of a database. Storing separate chapters on different servers can facilitate clustering, and using URLs to access the various required chapters can assemble a whole distributed Profile.

LazyView, the query and reporting tool released in Sentences v3.0, allows connection to an external database via Java Database Connectivity (JDBC). Hence other data sources, such as relational, and other non-associative databases, can be accessed from within Sentences applications. During the query definition, the user defines the mapping from the external database to that within Sentences, which interrogates the external database at run time. LazyView offers a default mapping, based on the table definitions of the non-associative database – this can be refined, and conditional processing applied to record matching criteria. 'Equivalence' between tables, or rows within tables, can be specified.

The user can specify partial merger of results in presentation form, with details of one being shown as a tab of the other record's data. Alternatively, 'supersedence' can be specified, creating one remaining record from two original matching records. Column equivalence offers powerful choices to merge fields. If database merge by program is required, the equivalence and supersedence functions can be invoked by an API.

While direct access by Sentences to external target databases is read-only, program interfaces may be defined from applications to update such databases, or an XML document sent to a database for update. The following diagram shows the possible operation of Sentences in conjunction with relational databases.



LazyView also incorporates a Query Editor, which allows assembly of a query from schema elements, specifying conditions or parameters to be satisfied. Results of a query can be viewed as a list, with default means of presentation used according to the data items included. Alternatively, XML, XSL, or a reporting tool can be used for presentation of results.

Lazy Analytics is a tool written by an OLAP specialist partner company, Digital Aspects, which used Sentences APIs to write an OLAP engine suitable for the Sentences database. A user working within Sentences Explorer may define the dimensions, levels, and measures that make up an OLAP cube. In order to invoke the OLAP analysis capability, analytics is incorporated within the user's profile. None of the information from source databases is copied into a Sentences database for OLAP purposes, allowing cost savings over some OLAP functionality in the market place. Presentation of results is via Microsoft's OLE DB for OLAP interface (for Windows clients), or the 'XML for Analysis' interface for browser-based and thin clients. Microsoft Excel is used as the client interface to support other common analytical functions.

Product Emphasis

The AMD, as the basis for the Sentences product, allows database and applications definition to be undertaken much more easily and rapidly than using the relational model. Consequently, using Sentences should significantly reduce the cost of application development, and subsequent maintenance requirements would continue to see these benefits achieved.

Where an application's data is particularly complex, or is unstructured, the relative simplicity of database definition and query with Sentences will add greatest value. The product has no market competitors, as no other enterprise-strength associative database is available commercially. Lazy Software does not consider relational database vendors to be selling the same type of product.

▶ DEPLOYMENT

Sentences is written in Java. Its client is browser-based, running from a Java applet – however, applications can be deployed with HTML pages via an API. The Sentences Server runs as a Java servlet installed as part of a Web server, and runs automatically to process requests received from the Sentences Client. The only installation required on the client is for the Java 2 Runtime Environment (J2RE) plug-in for the client Web browser.

The supported platforms for Sentences are

- Microsoft Windows NT4.
- Microsoft Windows 2000.
- Microsoft Windows XP.
- Linux.
- Sun Solaris.
- IBM xSeries.
- IBM iSeries.

Supported Web Servers are:

- Microsoft IIS 4.
- Apache.
- Tomcat (supplied with the product).
- iPlanet.
- IBM WebSphere v4.

In terms of the product's database storage, each entity is stored as an 'item' on a Sentences database, with a unique, meaningless identifier. An association is stored as a 'link' (also referenced by a unique identifier), with references to the source, verb, and target items that form the association. Identifiers are never viewed by the user, but are used by Sentences to refer to the item or link.

The following is an example of the use of the identifiers to store items and links on the database, representing the associations of some entities:

The information:

Flight BA1234 arrived at Heathrow Airport….on 1-Jan-2003…..at 10:05am.

Database elements:

Items	
Identifier	*Name*
77	Flight BA1234
08	Heathrow Airport
32	1-Jan-2003
48	10:05am
12	arrived at
67	on
09	at

Links			
Identifier	Source	Verb	Target
74	77	12	08
03	74	67	32
64	03	09	48

Lazy Software provides a four-day foundation course to train users, covering the building of database schemas, population of the database, use of forms, and installation and administration of the product. On-line and telephone-based help is also available. While use of Sentences requires a different set of skills, those are not based on the ability to learn some syntactical language, but mainly on understanding what is required from a business perspective, what entities will be involved and the associations that need to be created. Depending on requirements, XML or XSL knowledge might be required to produce customised reports, or Java skills required in order to manipulate the database via its API for specific application needs – specialised training is available for these areas.

Custom datatypes can be created by anyone who is proficient in Java, by writing Java classes that implement the Sentences Datatype interface. Both custom and supplied datatypes control the behaviour of an entity type. Custom datatypes appear to the user alongside the supplied Java datatypes; datatypes cannot be changed by a user via Sentences Explorer.

Implementations in various character languages are facilitated by the product's UNICODE base, taking the character set of the language via the user choice defined in the client operating system environment.

Due to the radical difference in database technology model, performance cannot be directly compared to competitor relational databases – Lazy Software is engaging with the Transaction Processing Performance Council (TPC) in order to produce metrics for Sentences.

▶ STRATEGY

For Sentences to become more successful, Lazy Software has to overcome barriers of perception and imagination. The first is that as a pure ADE it competes with well-established products that have a large installed base. As a new data model it competes with the corporate reliance on relational technology. Therefore, central to the success of Sentences will be tying these two elements together to demonstrate that existing ADEs and relational databases are not the only way that business problems can be solved.

It is attempting to get this message across by having a direct channel in the UK and forming distribution channels in the rest of Europe and the US. The company is an IBM Business Partner (Advanced Developer Member), and a Sun iForce Partner.

A typical customer engagement incurs initial costs of £45,000 – £100,000. Of this, the initial licence cost is £37,500 permitting unlimited use on a single server, and the remainder is the cost of training and services. An annual £7,500 maintenance charge is payable, which includes unlimited telephone support, and all new release and version upgrades.

Lazy Software plans one new version of Sentences per year, and two or three intermediate releases. The v3.5 release, due in April 2003, includes a diagram editor, componentised installation and licensing and password protection for chapters.

October 2003 is the target for v4.0 release, to include a fully omnicompetent HMTL user interface to complement the omnicompetent Java user interface, and automated mapping from relational tables. V5.0 is planned for October 2004, in which declarative specification, and automatic implementation and enforcement, of business rules is planned, comprising both constraints and actions.

The benefits of Sentences are relevant to businesses of all sizes, but currently Lazy Software is targeting the enterprise market (both vertical and horizontal) to try and gain acceptance for what could be a revolutionary product. Past customer deployments of Sentences have been in tactical systems. With value delivery as a strategic focus of many organisations, Sentences may be well placed to feature as a more strategic solution.

▶ COMPANY PROFILE

The company's founders first came together in 1984 to form Synon Corporation, which went on to become the dominant application development tool vendor for IBM's AS/400 platform (now iSeries), gaining a 70% market share. One of those founders, Simon Williams (now Lazy Software's CEO), was Synon's Chairman and CEO until 1990, and CTO thereafter. Synon was sold in 1998, whereupon the same founders established Lazy Software with £1.5m of their own cash. Venture capitalists invested an additional £6m in 2001, and the company remains privately owned.

21 employees work for Lazy Software, of whom 19 are in the UK and 2 in the US. 40% work in R&D, 30% in sales and marketing, and 25% in support and services. 80% of revenues are earned in the UK, the remainder arising from the US. Currently Lazy Software has over 40 customers, whose use of Sentences is mainly on a departmental basis i.e. tactical, rather than strategic, applications. Key clients' names include Alcan, Argos, John Crane, Johnson Matthey, Lloyds TSB, Sterling Commerce, and Antalis.

▶ SUMMARY

Butler Group believes that Sentences from Lazy Software is a solution that the market has been waiting for, even though the market may not have realised the fact. Information integration, and the ability to add value to data, are critical issues for modern organisations – both are addressed impressively by this product. What might have been an academic exercise in database advancement has been brought into the business arena by a clear understanding of the shortcomings of both the relational model and the object model.

This business focus has been extended by the creation of the ADE, which will ensure that users can start to benefit from implementing this solution without a steep learning curve. While Butler Group believes that few organisations will move away completely from the relational model to embrace this new paradigm in the immediate term, it also believes that organisations that are open-minded enough to consider radical solutions will be delighted by the ease of use of Sentences, and the business benefits that it will bring.

▶ CONTACT DETAILS

UK Office
Lazy Software
Mercury Park
Wycombe Lane
Wooburn Green
Buckinghamshire
HP10 0HH, UK

Tel: +44 (0)1628 642300
Fax: +44 (0)1628 642301

E-mail: info@lazysoft.com

www.lazysoft.com

US Office
Lazy Software, Inc.
28 State Street
Suite 1100
Boston
MA 02108
US

Tel: +1 617 988 2731
Fax: +1 617 457 5700

E-mail: US-Sales@lazysoft.com