



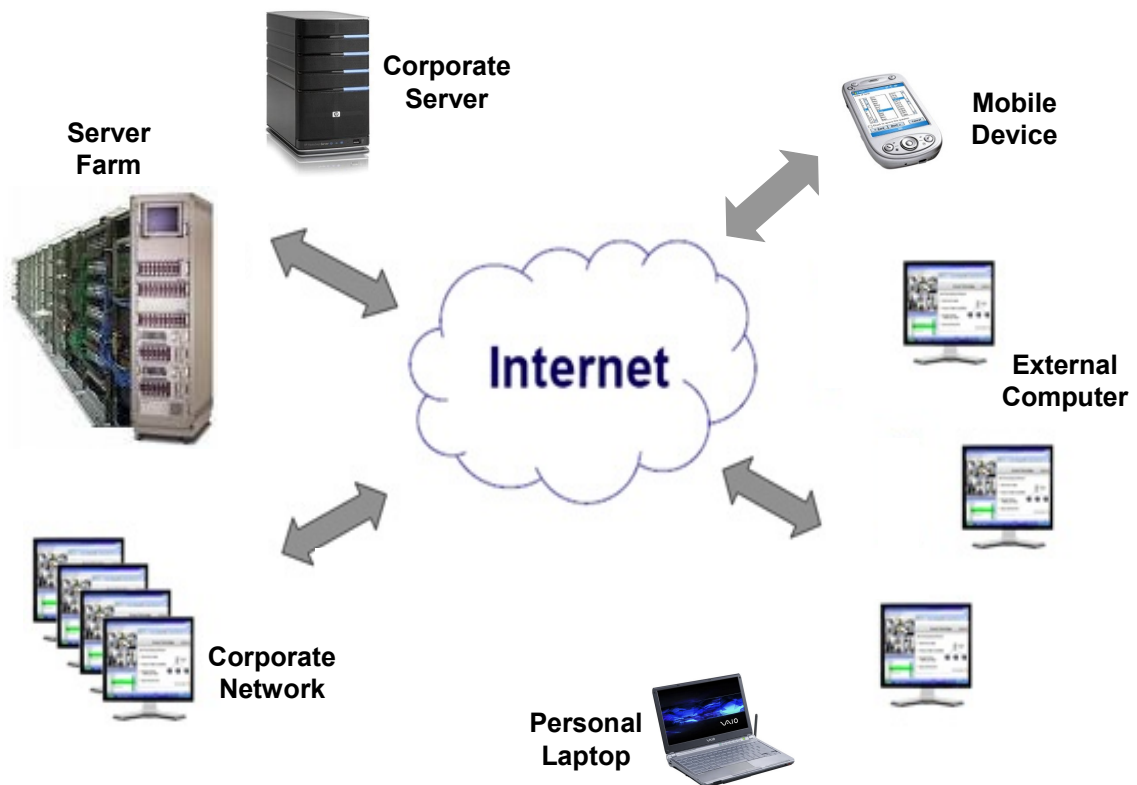
# FlitePlan

CLOUD COMPUTING SOFTWARE

**Executive Overview**  
**Business Architecture**  
**Technical Architecture**

## Executive Overview

The evolution of the internet, according to Wikipedia, has occurred in distinct phases, each with increased capacity and capability: Web 1.0 was read-only, Web 2.0 was read-write, and now Web 3.0 is read-write-execute – enabling fully fledged applications to exploit its power. Rich internet applications having all the features and functionality of traditional client-server programs can process the user interface on web browsers and perform mission critical functions and large scale data management on application servers. By being everywhere, the internet allows clients and servers located anywhere to interact, without constraint, offering immense business and technical advantages.



To fully exploit these advantages, however, needs a new type of software. On the internet, devices communicate without dedicated connections, with variable network delays, with potential security risks, and often with thousands of concurrent users. Many of the technical challenges have been addressed and solved by organizations such as Google, MySQL, IBM, Apache and others. FlitePlan is one of the first products to deliver robust, scalable, secure and flexible business solutions using these technologies.

FlitePlan directly represents business organizations, processes and resources in software, so is easily understood by business people. Organizations model how groups, companies, branches, divisions and even individuals are organized; purchase, service, manufacturing, finance, sales and other business processes describe how work is done among organizations; resources are the products, assets, money et cetera used and produced by these processes. Standard FlitePlan applications support most common business functions, and can easily be extended to satisfy specialised needs. Partial open source licensing enables business analysts and programmers to adapt these applications to particular business needs without compromising performance, security or integrity.

Business processes change the financial and operational value of resources. These changes are recorded by FlitePlan in a data warehouse having a dimension or axis for each aspect of their value. A powerful business-intelligence report generator enables management, financial

and operational information to be accurately and efficiently summarised in formats familiar to users. For example, income statement, balance sheet, accounts receivable and payable, inventory and work in process and other financial reports are exactly as they would be from a traditional accounting system. However, a data warehouse is not limited to financial data and allows reports having physical values for multiple organizations at many locations over long periods. Sales quantities, inventory balances, shipping tonnages, employment hours, et cetera are examples of non-financial values that are useful in business. An integrated charting tool can present this data in most popular graphical styles.

Existing databases are accessible to FlitePlan applications via an object-relational layer which supports most modern databases. Being sophisticated, it allows transactions across multiple databases of different types, so is able to integrate and eventually supplant legacy applications without the risks of a 'big bang' changeover. FlitePlan software can also access third party web services such as satellite imaging and mapping, product data, customer credit rating, foreign exchange, shipping and other data useful to business applications. Organizations that continue to spend money on monolithic ERP applications introduced in the Y2K era may benefit by evaluating this fast and adaptive Web 3.0 alternative.

Hosted on the internet, FlitePlan applications are truly global, and can help integrate the activities of customers, suppliers, employees and other partners wherever they are. No special action is required for a supplier in Singapore to deal with a customer in Seattle and with a banker in London. FlitePlan's workflow management ensures that documents are routed in correct sequence to the appropriate people in real time. This means, for example, that a request for quotation by a potential customer anywhere in the world is immediately presented to a salesperson for response. Process steps can also be delegated to customers, suppliers and other external bodies to improve service and accelerate workflow. Finally, internationalisation of a FlitePlan application is simply a matter of translating simple labels into the target language(s) – the web browser determines which language and date/number format to use.

These advantages are available to most organizations with no additional investment – almost every company and many individuals have access to the internet – and most use some form of corporate database. You probably already have the hardware and software required to run your FlitePlan applications. However you may choose to host the applications using a service provider offering 24x7 secure computing over the internet. Such services are offered at little or no cost by internet savvy companies such as Amazon, Google and IBM, enabling your company to entirely eliminate the cost and risk of its database, operating system, network, antivirus and other software, thus rendering your server rooms, power supplies, computers, backup libraries and support staff unnecessary.

The internet has revolutionized many aspects of today's business world and has created (and destroyed) massive enterprises. You now have the opportunity to benefit from the competitive advantages enjoyed by Google and Amazon.com without the substantial investment and risks that they have incurred. Before you spend another cent, penny or yen on your information systems we suggest you evaluate the Web 3.0 opportunities offered by FlitePlan.

Generous licensing options allow you to choose the FlitePlan that best suits your needs:

- Free for personal, small business, educational and non-profit use
- Time based charges for larger commercial organizations
- Transaction based charges for hosted applications of any size
- Developer license for source code, tools and methodology

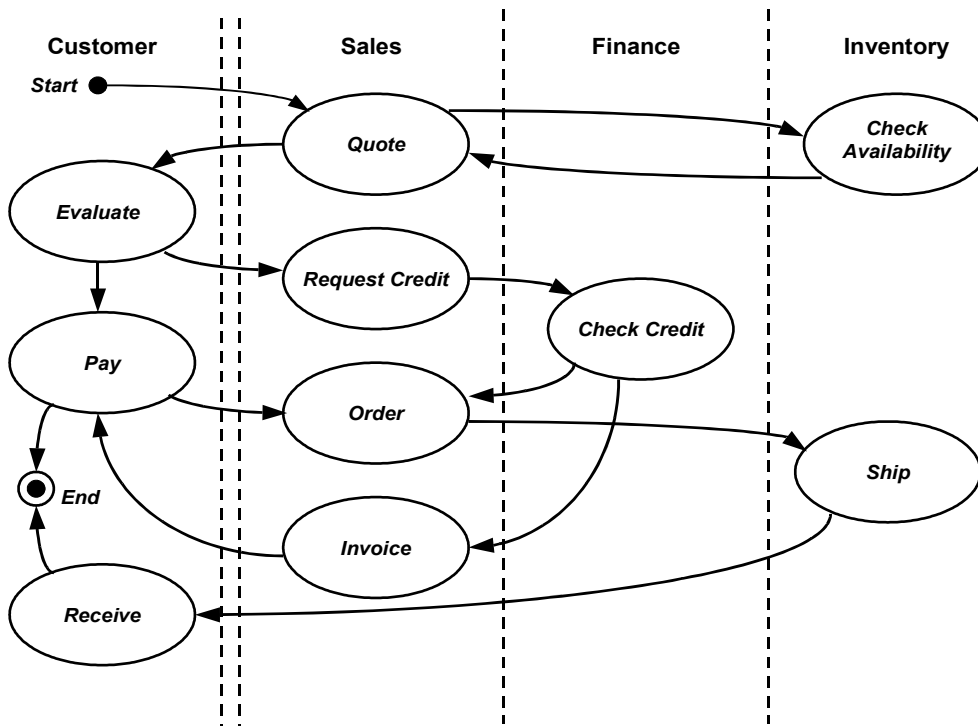
Please call CBT for information, demonstrations, licensing details and prices.

## Business Architecture

FlitePlan has a model-driven architecture (**MDA**)\* in which software components reflect business concepts such as resources, processes, events, organizations, roles, et cetera. It uses the resource transform and *resource, event, agent (REA)* concepts, pioneered by Chris Marshall and William McCarthy in the 1980s, which have gained wide acceptance and have been successfully applied in many enterprises. The REA approach treats an information system as a virtual representation of the actual business, in which software objects directly model real-world *business objects* such as parties, resources and processes: It describes how corporate and individual parties (agents) acquire and use goods, services and money (resources) - to create value via purchase, sales, manufacture, service and other *business processes* (events).

## Process, Organization and Role

Business processes guide the flow of work among organizations and individuals to ensure their accuracy, efficiency and security. A purchase of the wrong item from an inferior supplier with incorrect delivery date and inadequate cost control is unfortunately all too common. Ad hoc controls introduced to improve the situation add overhead costs and often fail over time. Workflow management (**WfM**) and role based access control (**RBAC**) features help to ensure that steps in the business process are done when required by properly authorised users. To achieve this, each task in the process is identified, allocated to an organizational role, and joined to other tasks by business rules. For example, a process might be started by a request for a product by a customer – a sales clerk checks if stock is available with the inventory controller, responding to the customer with a quotation or an apology. The customer then evaluates the proposal, perhaps requesting payment terms - which are checked by the finance department - alternatively paying upfront. When credit is allowed or payment is made, the product can be ordered and shipped. The business process diagram separates these tasks into the organizations (customer) and roles (sales, finance and inventory) authorised to do them. FlitePlan uses this information to route tasks in sequence only to users that have been assigned the appropriate role(s).

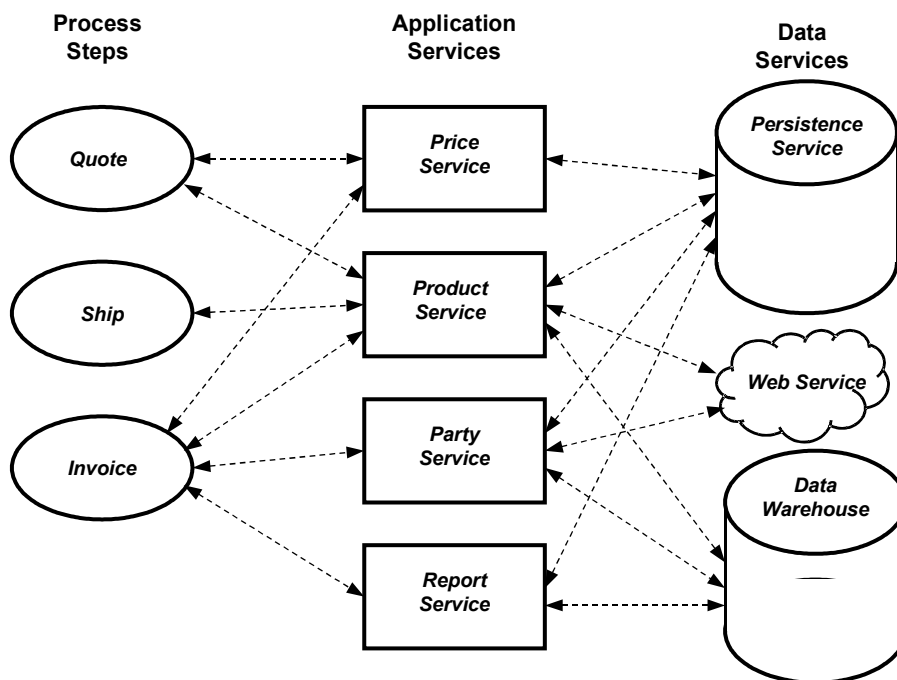


\* Refer to Wikipedia and Google for more information on the acronyms and phrases highlighted in bold and italic fonts

## Resource and Value

A resource is anything of value to an organization, and FlitePlan measures its value for planning, operational and accounting purposes. A business process consumes, produces and/or modifies one or more resources, intending to increase their overall value. Financial value is measured by accounting systems, which typically fail to provide the non-financial planning, operational and intelligence needs of an enterprise. FlitePlan data services enable comprehensive, consistent, flexible and extensible models of each resource including its financial and physical values. Furthermore, a resource may have user-defined attributes. For example, a clothing company may wish to record the style, material, colour, size and designer of its products. The attributes are added without any change to the database structure, may be used in posting data to the warehouse, and for other purposes such as online searches. The FlitePlan search facility allows both exact matches and *fuzzy* search to find close matches.

Application services present and update data in response to requests by a business process. For example, the quotation step of a sale process might use one service to calculate the price of each item given its quantity, due date, location, pack size and currency; another service to find out if and when inventory is available for shipping; yet another to calculate the shipment cost; and finally use a report service to format the quote to be sent back to the customer. The services are accessible to clients via the internet and directly to each other on a server. Complex requests can be factored into a set of simple operations, and advanced transaction management ensures that a multi-stage request is either completed or aborted in its entirety.



The *persistence* service stores the state of a business in one or more databases with functions to create, read, update and delete (**CRUD**) data that describes business objects such as customers, products and finance. The service transforms business objects used by applications to and from database records, and maps between their relational and object-oriented forms. This is completely transparent to the end user and application programmer, but allows for customised mapping when needed. For example, an object representing a product may reference and update data in many tables in several databases at different locations, but to the user it appears to be a single entity.

The service also harmonizes data from all sources to eliminate differences in their names, representation, precision et cetera. Database dates and numeric values, which may be in

many formats in the databases, are converted to and from ISO formats for processing by applications. To further facilitate international usage and trade, FlitePlan is preloaded with data for each country, currency, language, location and unit of measure in the UN/CEFACT, ISO and SI standards. Flexible tax tables allow sales, value added, pay as you earn and other income tax schemes to be defined and used for any jurisdiction. Product price lists are defined by date, quantity, unit of measure, location and currency for all types of contract, and each contract can have multiple suppliers and customers. These features eliminate much of the complexity of legacy applications while retaining the very high performance available from modern relational databases.

*Web services* also allow third party services to be used – services offered by Google include satellite images, maps and search, for example, which may be accessed by the application. Other companies offer product data, customer credit ratings, foreign exchange rates, shipping options and other services that can be used by the process. This information is available free of charge and by subscription in many different formats, but it is presented by FlitePlan in a consistent form to applications.

### **Business Intelligence**

Business is a collection of activities carried on for a purpose, be it science, technology, commerce, industry, law, government, defence, et cetera. The communication facility serving the conduct of a business (in the broad sense) may be referred to as an intelligence system. The notion of intelligence is also defined here, in a more general sense, as the ability to apprehend the interrelationships of presented facts in such a way as to guide action towards a desired goal. Business intelligence (**BI**) refers to technologies, applications and practices for the collection, integration, analysis, and presentation of business information and sometimes to the information itself. The purpose of business intelligence is to support better decision-making, so also described as a decision support system (**DSS**). [Wikipedia]

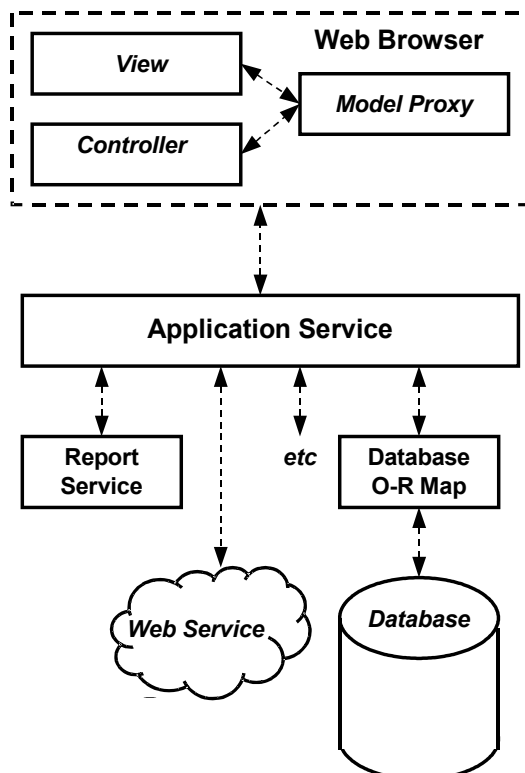
In changing the state of a resource a process changes its financial and operational value. For example, receipt of an item into a store increases the quantity on hand and inventory asset value. FlitePlan uses a multidimensional *data warehouse* to provide historical, current, and predictive business information directly from operational data in real time. Data is partitioned in the data warehouse into facts, which are generally numeric values, and dimensions, which are properties that give context to the facts. For example, a sale is measured by quantity and price values (facts) and unit of measure, currency, date, location, and product, customer and salesperson identities (dimensions). Other dimensions may be used for actual, projected, and budgeted values, et cetera. Benefits of the dimensional approach are that it is easy for managers to understand and to use, and data retrieval is very fast so reporting and charting tools can rapidly extract, analyse and present information from the data warehouse.

Posting rules specify how values are to be recorded in the data warehouse. They are defined in the application services and are applied to all processes that use the services. For example, inventory used in production or maintenance, shipped to customers, or issued as marketing samples are subject to the same posting rules – thus avoiding many of the problems associated with traditional accounting systems – where each type of transaction may have different logic. While most rules are standardised to conform to *generally accepted accounting principles*, other rules may be adapted or added for specific information needs. For example, commission might be accrued to a salesperson for each sale without affecting the financial ledger. Such information can optionally be included in or excluded from reports at the user's choice. This innovative approach to data warehousing removes the need for separate general ledger, accounts receivable and payable, inventory and work in process control, and other systems.

## Technical Architecture

The FlitePlan architecture separates an application into distinct layers, each having a specific purpose, which are typically the responsibility of different people. It uses a model-view-controller (**MVC**) framework to manage the relationships between the data on the server (the model), its presentation in a web browser (the view), and the buttons and other widgets by which the application is used (the controller). A finite state machine (**FSM**) governs the behaviour of each business object and controls the visibility and availability of widgets according to the current state of the object. The FSM has *guards* to enforce integrity and business rules when moving between states. The model is partially replicated on the client as a proxy created in response to requests sent to the server, typically triggered by controller events. Careful design ensures that the partial model is compact and relevant to the task at hand so as to minimise network traffic of these requests and responses.

Client devices deliver fast graphics and rapid response to the user when there is no interaction with a remote server, but can be slowed significantly by network and database delays when such interaction is required. An elegant solution to this problem is for the client to interact with the server asynchronously. This means that the client does not wait for a response to its request, but continues with other work, and handles the response from the server if and when it is received. Although simple in concept, asynchronous programming introduces complexity that is difficult for the average application programmer to control. FlitePlan hides this complexity entirely within its MVC framework. Business applications are further simplified by the FSM which eliminates most of the conditional if..then..else branching found in procedural code. The integrated RBAC furthermore removes the need for security concerns in application programming. This is broadly illustrated by the following diagram:



*Client application created in standard HTML and JavaScript runs on web browsers such as Firefox, Opera, IE6-8 and Chrome. An MVC framework manages user interface widgets and model proxies to deliver the look and feel of desktop software. Clients request data from the server via asynchronous remote procedure calls.*

*Application services respond to client requests with serialised objects representing the current structure and state of the business model. The objects are created from databases, web services and server-side computations. Services can also execute CRUD operations, send email, create reports, et cetera.*

*A database configuration file maps standard data manipulation operations into database specific SQL statements. Query customisation and tuning may be done here.*

*Almost any database or web service can be used, including legacy application data. Rich data such as provided by Google Maps, online news feeds, financial data services and other information sources are simple to present via application services to FlitePlan clients.*

## Presentation Layer

FlitePlan applications execute in standard web browsers without plug-ins or any other software on the client device. This eliminates the logistical, management, licensing, version control and other technical problems associated with traditional client-server software. A new version of a FlitePlan application is downloaded automatically if and when the web browser detects that it has changed, otherwise the browser continues to use the previously loaded version. Applications can take advantage of the formatting abilities of the browser, including cascading style sheets (**CSS**), to gain maximum visual impact. An application is formatted simply by modifying its css file with the desired page *markup*. This work is typically the domain of creative marketing people and may be delegated accordingly with no risk to the functionality, integrity and security of the application.

An application can also be displayed in any language, including those with non-Roman alphabets, by simply translating the text in a properties file to the target language(s). The correct language version is displayed according to the locale setting in the user's browser. Date and number formats are also locale sensitive. Perhaps most importantly, what the user can see and do is controlled by the current state of the business object, and by the roles he or she has been allocated by the system administrator. Users see only the forms and fields that are required for them to do their work, and may execute only tasks for which they have permission. Finally, FlitePlan makes extensive use of *lazy loading* to ensure that only objects that are actually required are loaded.

This advanced framework relieves the application programmer of many difficult technical tasks, allowing the focus to be on the application logic itself. FlitePlan applications are assembled from a set of high-level components, which encapsulate access control, state management, service request and other functions behind simple programmer interfaces. Typically a single line of code is all that is required to add a new widget to a form. The many business objects supplied (including source code) are sufficient to illustrate how to develop advanced applications in FlitePlan. Unlike tools that rely on code generation, component based applications may easily be adapted to suit particular needs by persons with minimal web programming experience. Someone with a good understanding of the business domain and process requirements can typically develop and adapt client applications.

The finite state machine – which does not rely on code generation - radically simplifies code in any non-trivial application. A state transition is defined by a line of declarative code to specify its event, action, guard and states – and is easily derived directly from a *state diagram*. This both simplifies code and encourages software engineering discipline. User interface components may be controlled by the FSM to change their appearance and function according to the current process or entity state. In addition, declarative role based access control enforces the user's read, write and execute permissions at form, field, command and code levels. Finally, business objects requiring work are *automatically* routed to users with the appropriate role. Custom workflow rules may be added to the application if required.

## Service Layer

The server architecture required to support FlitePlan is also radically less complex than that required for traditional client-server and web applications, which typically need session state to be maintained on the server to 'remember' what the user is seeing and doing between requests. Web applications using Active Server Page (**ASP**), JavaServer Page (**JSP**) or JavaServer Face (**JSF**) technologies reconstruct entire HTML pages on the server and send them to refresh the browser every time a change is made. Asynchronous JavaScript and XML (**AJAX**) techniques help to improve the user experience but are difficult to program and generally have nowhere near the interactive quality of desktop applications. FlitePlan emulates the desktop experience by delegating all user interface processing to the client browser and obtaining and updating data as required from the server via *stateless* services.

Stateless services know nothing about their clients but merely respond to requests by doing one or more tasks and (optionally) returning data to the client. For example, a request for data might return information while a request to save a record or create a report might not.

Because they do not maintain session state for each client, stateless services scale to serve very large numbers of concurrent users without significant increase in response times. Furthermore, a service can perform a series of operations, including computationally intensive tasks, in response to a single request. A pricing service, for example, might have to obtain inventory availability, currency exchange rates, shipping costs and other data to determine and return a simple number - the price - to the client.

Services are typically used by different applications and are designed accordingly. Standard information is obtained and maintained via services that directly support business objects – account, asset, finance, product, party, sale, purchase, receipt, payment, price – are examples common to most organizations. Specialised business domains such as retail trade, distribution, manufacturing, maintenance, project management, travel and accommodation, scheduling, shipping et cetera typically require customized services. Standard UN/CEFACT, ISO and SI services provide country, currency, language, location, taxation and unit data needed for international trade. System administration is facilitated by authentication, organization, role, user and session services. The computing infrastructure is exposed via reporting, email, messaging and other technical services, and so on.

FlitePlan services are typically implemented by plain old Java objects (**POJO**) without any need for heavyweight enterprise frameworks. Complex objects can be assembled from multiple and distributed data sources such as relational databases and web services. Standard functions enable CRUD operations on entities and relationships in the database with minimal Java and without any structured query language (**SQL**) code. Simple high-level methods post values to the data warehouse and ensure that transactions having financial significance are correctly balanced. Very sophisticated accounting needs can be satisfied using such methods. These features also allow attribute, relationship and transaction types to be added and deleted without change to the database schema. Finally, advanced transaction management ensures the integrity of service requests under fault conditions.

Services are best developed by experienced software engineers. While simple changes can be made by relatively inexperienced programmers, the key challenge is to analyse and design the most suitable services and their operations. This requires an understanding of how a service might be used, often without a clear specification. Standard FlitePlan services are guidelines for new and adapted services, but other approaches may suit the needs and constraints of particular environments. For example, services can be written in C# for the .NET framework should this be preferred. Such decisions should be made by senior people who know both the business objectives and technology constraints of the organization.

## **Data Layer**

The data layer accesses database and web services and maps their data into formats suitable for the service layer described above. Any modern relational database including MySQL, DB2, Derby, PostgreSQL, SAP, Oracle, SQL Server, Sybase and Informix may be used for object persistence and data warehousing. A configuration file defines network location, login parameters, the dialect of SQL, and other database idiosyncrasies. The file uses an extensible mark-up language (**XML**) dialect to map database tables and columns to object classes and fields. Object and table data types, formats and default values are defined by simple XML declarations, and CRUD and other operations by parameterised SQL statements. Queries may also be formulated in the service layer and passed as parameters to the data layer, although this approach is discouraged for all but the most complex functions.

The standard data schema is normalised to eliminate redundant data and carefully indexed to accelerate performance, and the data warehouse is further enhanced for fast response even with very large datasets. However, many organizations have existing applications and databases, which cannot be abandoned overnight in favour of a new system. This legacy data can be used by FlitePlan applications with appropriate XML configuration files and services. For example, a service may draw information from several existing databases and tables, possibly residing on different computers in different locations. No changes are required to any of the applications using the service because its interface (ie: the operations

that it supports) does not change. Similarly, migration of data between databases requires only minor changes to the configuration files, thus allowing cost effective, high performance, options to be used.

This flexibility also enables organizations to benefit from *cloud computing* resources provided by Amazon.com, Google, IBM and other major corporations. Because customers do not own the infrastructure - they are merely accessing or renting - they forego capital expenditure and consume network, computer and database resources as a service, paying instead for what they use. Many cloud computing offerings have adopted the utility computing model which is analogous to how traditional utilities like electricity are consumed, while others are billed on a subscription basis. By sharing computing power between multiple tenants, utilization rates can be improved (as servers are not left idle) which can reduce costs significantly while increasing the speed of applications. A side effect of this approach is that computer capacity rises dramatically as customers do not have to engineer for peak loads. Adoption has been made even more attractive by increased network bandwidth, which enables excellent response times from centralized infrastructure at remote sites. [Wikipedia]

The data layer is the foundation on which services are built so requires people with in-depth knowledge of information technology. Object-relational mapping (**ORM**) requires some skill in SQL programming and optimisation, use of web services is often non-trivial, and providing reliable internet access and capacity can be challenging in some situations. While many of the issues can be delegated to a cloud computing service provider, technical skills should be available in-house or through consultants to develop, extend, port and tune the data layer. This expertise can also evaluate the best browser, operating system, network, database and reporting options, as FlitePlan does not prescribe any particular mix of platform technology.

## Conclusion

Applications using the FlitePlan framework offer significant benefits to organizations of any size, particularly to those wanting global reach, and especially when hosted in the cloud:

- Standard applications exist for many business requirements;
- Open application source code allows customisation for individual needs;
- No custom software is needed on the client – just a web browser;
- The Internet enables users to access applications regardless of time or location;
- Client applications run anywhere on any PC, laptop, net book, tablet or PDA;
- Role based security and work flow allow access only to authorised users;
- Application services run on all modern operating systems, databases and networks;
- Cloud computing providers can host applications of unlimited size, with benefits like:
- No capital expenditure because infrastructure is owned by the provider, not the user;
- Multi-tenancy to share resources and operating costs among many organizations;
- Centralised infrastructure in areas with lower real estate, power and labour costs;
- Increased peak-load capacity without the need to engineer for individual load levels;
- Improved efficiency over in-house systems that are often only 10-20% utilised;
- Performance monitoring to ensure users are not affected by bandwidth or computer load;
- Reliability and disaster recovery is ensured by way of multiple redundant sites;
- Services can be scaled up to meet ever increasing demands of very large user bases;
- Secure and available because of centralized administration, audit and backup.