ACTA TECHNICA CORVINIENSIS – Bulletin of Engineering
Tome IV (Year 2011). ISSN 2067-3809

[1.] Zoltán BÁTORI, [2.] Tamás HARTVÁNYI

DEVELOPMENT OF FORECASTING SYSTEMS

ABSTRACT:
In this paper we demonstrate a detailed overview of the history of forecasting software applications over the past decades, concentrating especially on the interaction between hardware and software. Additionally we present a framework by describing important developments of forecasting techniques in terms of hardware and software environments. We then focus on the application areas of forecasting software modules in business and planning environments which are often partially automated due to the large number of time series involved. Finally we make some suggestions about in which direction forecasting software should be improved.

KEYWORDS:
forecasting software applications, detailed overview

INTRODUCTION - THE HISTORY OF COMPUTER DEVELOPMENT AND FORECASTING METHODS

The history of forecasting and time series methods began in the 17th century when numbers of sunspots and price indices were analyzed by scientists. However, the practical use of statistical techniques has been made possible by the invention of computers in the 1950s.

At the beginning, the use of computers for forecasting was limited by inadequate processor speed, random access memory and disk space. In the 1960s, forecasting was capable to analyze short and isolated series, collected in flat files, and processed by batch runs using Hollerith cards on mainframes. In these days programming was mainly done in Assembler and FORTRAN under a variety of different and largely incompatible operating systems.

The introduction of OS/360 in 1967 as a scalable operating system for IBM mainframes resulted rapid migration between hardware platforms, simplifying the movement of programs. It was developed by IBM (Brooks, 1974) with the intention of creating programs that could be run on IBM computers of different sizes. Prior to the development of OS/360, operating systems were only designed for individual computer architectures. With the arrival of the system, software could finally be moved from one computer to another.

The appearance of UNIX in 1969 resulted in the development of portable software for smaller systems. The introduction of personal computers such as IBM PCs and Apple Macintoshs in the first years of the 1980s allowed the use of computers at everywhere, independent of mainframes.

Many advances in computer science had an influence on forecasting. The continuous increase of processor performance, memory and disk space allowed scientists to deal with larger data sets and more complex algorithms. Later cathode-ray terminals supported the design and use of interactive applications with their character user interfaces (CUI), screen reports, and graphs. This development occurred in the 1970s, when a lot of mainframe management information systems and manufacturing resource-planning systems appeared on the market. The next step to graphical user interfaces (GUI) in the 1980s changed the software environment, as allowed a much larger community to use forecasting software.

Application software also changed in the 1960s, when forecasting methods were individually programmed using either Assembler or FORTRAN. This allowed later the selective use of new techniques such as smoothing or complex techniques such as Box–Jenkins models. Several statistical and econometric software systems were developed in the 1970s. In the meanwhile, material replenishment systems, which focused on inventories and production, were developed independently of the statistical forecasting tools. Later these became the roots of enterprise resource planning (ERP) and management information systems (MIS). This is important because this difference has never been closed entirely. Only in recent years have the data produced by ERP systems been used as input data into forecasting software.

Databases were developed simultaneously. At the beginning, data was collected manually and stored on punched cards, or transferred to text files on disks and tapes. While this remained the primary method of data input into statistical systems for a long years, transaction and planning data were migrated to database management systems (DBMS) quite quickly. The development process started from hierarchical and network databases in the 1970s to relational database systems in the 1980s, enhanced by object-oriented DBMS in the 1990s. The last two resulted in the object-relational database systems used these days. Within business planning, data processing was often replaced by PC-based spreadsheets, which were saves and modified on local PCs. This separation of local planning data and centralized transaction data still can be detected today, often resulting in problems of consistency and concurrency of the database. In a further step, the integration of database systems and transaction systems happened. With the constant increase of available data, databases moved on to data warehouses, offering also a wide range of tools for extraction (on-line analytical processing, OLAP), visualization, and analysis, including predictive data mining techniques.

DEVELOPMENT PERIODS OF FORECASTING SOFTWARE

While computer development was evolutionary, the development of forecasting software can be categorized into periods. These are:

- *Period I.: Mainframe forecasting software*
- *Period II.: PC and workstation forecasting software*
- *Period III.: Process-oriented and highly integrative forecasting software*

Period I.: Mainframe forecasting

Mainframe software, either in batch or timesharing mode, dominated forecasting software in Period I. However, some very popular programs took quite a while to offer even basic forecasting features. For example, SPSS surprisingly had no forecasting functionality until it added the „Trends" module in 1994. On the other hand, SAS/ETS was first released in 1980.

In industry, commercial firms devoted to forecasting for industrial clients did so with mainframe computing capabilities. The main packages offered were DAMSEL, TROLL, AUTOBJ. B34S, and TSP were available not only to industry but also to academicals use.

Prior to this time, microcomputers had been the domain of computer programmers, primarily because of the lack of application software. The first popular spreadsheet package, Visicalc, turned the microcomputer into an effective business application. Lotus 1-2-3, released in 1982, offered the combination of spreadsheet, presentation graphics, and simple database functionality to for the PC. However, there was no forecasting in this field in Period I. There were two main reasons for this. First, the lack of solid compilers. Second, mainframe packages could not be moved directly to the PC, as the PC was still not powerful enough. Only some parts of the mainframe software could be used on a PC version, and even then it should have to be rewritten in BASIC instead of the original FORTRAN.

Period II.a: IBM PC and forecasting packages

By 1985, the successful IBM PC and its clones had been around long enough that forecasting software was available. By 1989, more than one hundred software forecasting packages were available for the PC (Rycroft, 1994).

In this period, universities also began to move away from mainframes, setting up PC laboratories. For industrial companies, the situation was slightly different. Corporate IT departments, in order to controlling the mainframe, had long dictated computer use. Every single department could buy and use a PC, and they didn't need the IT department to approve the purchase or maintain the computer. For such departments, the PC had become effective enough to work in large scale forecasting, for example for production and inventory purposes.

However this decentralization generated new problems. Different organizational units might track the same data but maintain them differently, or use different numbers to represent the same facts. Therefore, the databases would also produce conflicting data and different forecasts too. The PC was at this stage just a batch engine, it was capable of only to produce forecasts for large numbers of items, and then write these forecasts to a file. It was still not good enough to do this work interactively.

The wide spread of standalone PCs produced its own urgent need. These PCs might solve the problems of individual organizational units, but they did nothing to improve the flow of information between these units. The persons making the forecasts for production had no idea how many parts were in inventory. Solving this incompleteness would be up to Enterprise Resource Planning (ERP) systems.

Period II.b: modern PCs and forecasting software

The Intel 486 processor appeared in the market 1989, and the class of forecasting problems for which mainframes were necessary becomes much smaller. The capabilities of forecasting software had reached a stage where even persons with no technical training could benefit from methods that previously had required technical training and support. After this, given just a unvaried time series, a program could determine which method best suited the data (for example exponential smoothing or ARIMA) and then optimize the parameters for the chosen model. The importance of this advance was that people could work out very good forecasts without consulting an expert. However for these forecasts to be good, they also needed reliable data. In this period, each department maintained its own database and each database had to be updated individually, so multiple databases could not be updated from the same source. Consequently, in these databases there were conflicting information and could not serve the entire organization. ERP vendors such as ORACLE, SAP and People- Soft realized the opportunity, connecting the disparate computers *and databases via a client/server architecture. However, it required many years before this task was completed.*

Period III.: PC and client/server architectures with forecast software

After ERP systems has been implemented, with all the data flowing back and forth, forecasts were completely missing from them and were often made with Excel 6, if at all (Sanders, Manrodt, 2002). Forecasting methodology has made great developments (Chatfield, 1996), but the pace at which these advances have been included into software were not fast enough. As far as industrial software is concerned, ERP vendors can't produce forecasting software. The solution was the integration of existing forecasting software with the ERP systems.

DEVELOPMENT OF FORECASTING SYSTEM FOR BUSINESS AND OPERATIONAL PLANNERS.

BUSINESS AND OPERATIONAL PLANNING

Business planning repeated on a regular basis, often with the creation of a monthly sales plan. Accordingly, the sales plan is the basis for marketing plans, purchase and production decisions or investment planning. A sales plan differs from an operational plan in that it addresses a higher level of aggregation in terms of both time and product, and is expressed in revenues rather than volume. Business planning forecasts are usually worked out on a monthly, quarterly or annual basis for product groups (instead of products), brands, and different business units such as sales regions. The average forecasting horizon usually ranges from 1 to 5 years.

Production and logistic decisions are supported by an operational plan. The main aspects of these plans come from the company's supply chain:

- *demand plan,*
- *inventory plan,*
- *transport or distribution plan,*
- *replenishment plan,*
- *production plan,*
- *maintenance plan, and*
- *collaborative plans.*

For operational planning, forecasting programs are used to calculate future demand per stock keeping-unit (SKU) on a daily, weekly or monthly basis. For example in energy planning, forecasts are required by the hour and at 15 minute intervals. In general, forecasts at the SKU-level needed for a large number of items, very often in the thousands, and these items are usually grouped into a product hierarchy, by distribution channels and by sales regions. When numerous items must be forecasted on a frequent regular basis, the uses of pre-defined or automatic forecasting techniques are critical. Forecasting systems must not only meet organizational requirements for accuracy but also for processing speed and robustness (Hartványi, Nagy, 2009).

The difference between business planning and operational planning depends on the way a firm organizes its planning processes. Ideally, a forecasting system should integrate both elements into a consistent set of plans. This is not easy for many companies today, where we find separation more often than integration. In operational planning, the forecast time interval is short and often not more than 6 weeks. These forecasts heavily affect decisions on the production levels per line, lot sizes, transportation schedules, and the purchasing of materials for particular time periods. In business planning forecasting the number of time series is usually low, allowing individual inspection and modification by the planner. On the other hand, in operational planning the number of time series is huge, which severely limits the possibility of individual inspection and modification. So, automatic procedures for forecast calculation are necessary.

Additionally, business planners are usually less experienced in forecasting techniques than in the functional areas of the business, such as marketing, finance and accounting. For operational planners it is the same, because they are often engineers or business administrators with a detailed knowledge of the logistical and technical processes, but with limited knowledge with forecasting methodologies.

FORECASTING SYSTEM FOR BUSINESS AND OPERATIONAL PLANNERS

Although forecasting libraries in FORTRAN and Assembler were used from the 1960s, but their use was very limited. Practically this meant that, sales plans were still set up on paper. Very slowly, larger companies began to implement these routines for business planning. However, these routines only provided forecasts, without any integration to other systems. To calculate forecasts, batch runs had to be programmed and intervention in the forecasting process itself (for example alteration of parameters) was not possible. While an analyst concentrating on forecasting a few series has enough time to try out different forecasting models to improve forecast accuracy, the time needed for an operational planner with a much larger number of series was too much. Therefore, they used only basic, standard models within the business forecasting process. As a result, forecasting accuracy was usually poor. It took a long period of time for companies to implement forecasting for business planning and even longer for operational planning.

In simple batch processing systems, the user could not interact with the software as it was described. This deficiency was remedied by the introduction of line oriented terminals, which allowed the software to ask the user for his input during the different processing steps. For example, a seasonal decomposition could be calculated before deciding whether to apply seasonal or no seasonal forecasting models.

With the appearance of character-based user interfaces the user could move the cursor all over the monitor, and enter instructions without following a prescribed sequence. This was the first time the planner could make technique selections by setting all parameters simultaneously on the same screen before starting the forecast calculation.

The first software products that allowed business planners to interact closely with the forecasting process appeared in the 1970's. These offered simple planning methods such as administration of time series, aggregation and disaggregation of series, planning screens, report generators, and functions to modify data and produce simple graphs.

Simultaneous progress was being available in data base programs (for example dBase) and spreadsheet systems. These, along with rapidly increasing hardware capabilities, offered major advances in forecasting software including parameter optimization (optimizing smoothing constants in exponential smoothing), multi-level forecasting for product and geographic hierarchies, data and forecast overrides, and so on.

Graphical elements, interfaces to databases, spreadsheets, external data sources, numerically and statistically robust methods, and simple automatic algorithms for the selection and specification of forecasting models were now common tools of business forecasting software. Not surprisingly, awareness of forecasting program tools grew rapidly, although the majority of companies at the end of the 20th century still used spreadsheets to develop sales plans.

Most recently, with the emergence of computing networks and intranets, participants in the forecasting process who were located at different sites could more readily collaborate with each other, particularly on sales plans. Collaborative forecasting capabilities were implemented into systems such as Demand Solutions, Futurmaster and Futurcast.

The difference between business planning and operational planning was now disappearing. Software's such as Peer Planner and Logol could be used to calculate forecasts at the product level for operational planning, as well as at the product-group level for business planning. The emphasis in such software's was not on the planning process but on the forecasting engine. However, the use of methodologically sophisticated forecasting software's strongly linked to production scheduling, transport planning, inventory and purchasing was unknown in the past. After a long period of time the main obstacles were the missing interfaces between the forecasting and the production planning components.

The first commercial forecasting software's, like IMPACT, were simply operational forecasting and replenishment systems, providing SKU forecasts. However, product-level forecasts were needed to support production scheduling and material replenishment. As a result, simple forecasting models were included in production planning systems, including BAAN, i2, Peoplesoft and SAP/ R3 (Fandel et. al., 1998). In comparison to the business forecasting software's, these operational systems (SAP/R3, mySAP) incorporated only simple methods such as trend curves, elements of exponential smoothing and tracking signals. Standard techniques, such as probability-based prediction intervals and out-of-sample evaluations, were not implemented. This difference was partially closed by the end of the 1990s. SAP for example developed an application called APO, where forecasting methods and sophisticated optimization routines augment the simpler functions included in SAP/R3. Still, sophisticated modeling such as the automatic Box–Jenkins systems as implemented in Autobox and SCA-Expert, as well as rules based forecasting (Collopy, Armstrong, 1993), have not developed into operational planning.

THE FUTURE OF FORECASTING SYSTEMS FOR BUSINESS AND OPERATIONAL PLANNERS

The most business and operational planners focusing on similar data, mainly sales figures. Sales effects at both the product and group levels have common origins, such as seasonality, trading days, and promotions, so it would seem that the same forecasting methods could be applied. On the other hand, there are significant differences. While a business planner focuses on forecasting a small number of aggregated series and makes effort to provide detailed explanations and reporting, operational planners have to keep their attention across a large number of series and frequent forecast rounds. So the operational planner can intend much less time to the specific features of the data and the forecast models, and seeks to automate the forecast process as much as possible.

Future challenge is to integrate the business and operational planning components in one application. First of all DBMS interfaces are required, just as they are made by analysts. However, for rolling planning systems it is not so important to have many interfaces. Instead, a stable, solid and fast interface to the transaction database or online data warehouse needed. Obviously this technique is simplified (and cheaper) if little or no interface programming is required. The reliability and online synchronization of the forecasting database with the actual enterprise database are the most important factors here.

Additionally, in supply chains of consumer products bullwhip effects often occur, which can be described as an increase in variability as fluctuations move up the supply chain. This means that retailers directly detect the customer demand without much variation while inventory and reorder levels fluctuate considerably across their supply chain. An possible method to handling this problem is the introduction of collaborative planning and forecasting replenishment (CPFR) and vendor managed inventory (VMI) applications. Because these forecasting processes involve several organizations of the supply chain, the software must come with a standard interface by which data can be exchanged. Some companies have developed standards regulating the data exchange processes, as well as the data structures and contents. For example, a standard for exchanging information within the German consumer goods industry has been worked out by the Centrale für Coorganisation (CCG, 2002). Standardization of supply chain management *processes is also begun, according to the Supply Chain Council (2004), which developed the „Supply Chain Operations Reference". More and more SCM vendors follow this process architecture, so forecasting system vendors will also have to pay attention to it.*

There is a special problem with truncated supply chains which quite common in practice. In the case of a surplus demand, most systems usually do not archive the actual demand but only the actual sales, so that only sales data can be used for forecasting. Consequently all forecasting techniques, with the exception of subjective approaches, generate biased forecasts which lag behind real demand. Another frequent problem which sometimes happens in practice is that sales data are achieved on the day of invoicing which often does not fall on the day of shipment. Shipment being relevant for production and logistics scheduling. Such problems cannot be solved by methodological inventions but only by correct database structures. Nevertheless, application and database vendors should set up solutions to save this information jointly with the time series to be forecasted to allow more detailed analysis and to make the application suitable for future methodological enhancements.

Additionally, when implementing special effects like advertising and calendar events, modeling is still often on a case-specific base requiring user interaction, for example by setting up a distributed lag structure for the advertising effect. With the huge number of time series in planning, some of the techniques indicated above (pre-defined effect profiles and lag specifications) should be run automatically and over hierarchies. Manual modification must be limited to a small number, requiring the use of some kind of effect prorating or automatic modeling.

Error prone procedures can only be used if exceptions will be caught by trap mechanisms. Unfortunately there is still much software that is not able to detect and handle numerical errors (such as overflow and insufficient data) suitably. Furthermore there are still well-known and widely sold software's where the forecasting methodology is limited to a small number of trend curves and exponential smoothing methods.

While many methods usually work satisfactorily for some longer series, especially on a monthly base for short time intervals, the increased application of high frequency data needs the incorporation of causal effects. Unfortunately, current software's do not offer a well established but simple methodology. As a result, most planners are forced to limit their forecasting repertoire to techniques which do not take causal effects into account.

Forecasting systems has been designed as standalone applications focusing on model selection for obtaining accurate forecasts. The forecast software vendors invest little in the processing of the forecasts for important decisions such as those involved in inventory replenishment and production scheduling. Moreover, many of them do not offer interfaces to other information systems.

An important deficiency of planning systems is the lack of attention paid to the theoretical basis of modeling, and therefore to the measurement of uncertainty in the forecasts. Without measures of uncertainty, the forecasts are not directly useable to replenishment and scheduling decisions. If, for example, forecast error distributions were computed and passed on to ERP software, forecast uncertainty could be involved into the computing of lot sizes and replenishment levels and intervals.

It is expected that forecast method selection should not be based simply on forecast error metrics but also on the costs of forecast errors in terms of replenishment decisions (Gardner, 2004). For example, the frequency of out-of-stock occurrences resulting from a certain method should suggest the need to change to a different method, as should excessive inventory costs. The required feedback between the forecast and the decision is not resolved in planning software's. The problem is aggravated by the concentration on point forecasts in optimization routines for production scheduling. Consequently they fail to provide the capabilities of modern forecasting methodologies to measure uncertainty.

In production systems, the number of out-of-stock situations is often tracked by key-performance indicators (KPI). Most KPI-systems do not include reliable indicators of forecast errors, the difference of forecasts from actual demand. While recording signals have been around since the 1960s, these metrics are more often found in planning system than in forecasting system. Sometimes statistical metrics can be found in systems with a primary focus on planning, but the majority of advanced planning systems only give non-statistical alerts. From a business point of view, out-of-stock percentages and excess stock are useful, but these statistics are hardly used for reporting forecast accuracy. For the last static and dynamic forecast simulations are suggested, but their availability in integrated planning and forecasting software's is not common.

CONCLUSIONS

Only a few forecasting systems offer state-of the- art functionality. Too many software's rely on outdated techniques. Examples are non-optimized smoothing parameters, poor initialization in exponential smoothing, erroneous formulas for computing safety stocks, graphs with inefficient time scales, lack of capability for forecast adjustments, aggregation of individual item forecasts and prorating of aggregate forecasts, as well as erroneous prediction intervals. Forecasting vendors need to upgrade to incorporate more recent methods.

Many new techniques in forecasting are not included into forecasting software within a reasonable time. Software vendors usually have to wait to see which new methods stand the test of time, but these methods are tending toward the simpler ones. If a scientist wishes to apply a new method, he must duplicate the effort of the inventor, as he must rewrite code that already has been written once. This is not the right way for science to progress. By given the choice between programming a difficult method and a simple method, many researchers will choose the simple method, because it is easier. So these are the easily programmable methods that get used in applied journals.

Software vendors concentrate to new techniques face an additional obstacle. They discover that a new technique has become popular in the applied literature, and then try to write their own code for the proposed forecasting model. Often they must make educated guidance about the details of the algorithm that were published in the article. Moreover, testing new techniques is even more difficult than it should be because inventors of the methods were not required to create an archive of the data used. Sometimes it is impossible for the developer to decide that his version of the program gives the same answer as the inventor's published results.

So while hardware processes are much faster, software advances are lagging behind. In order to test a new method, the developers and researchers have to program the code from the start. In the age of the internet, there is no reason not to record the data and the code used as the basis for an article. Imagine how easy it would be for a researcher to compare two or three different techniques if he did not have to program each one from the beginning.

We believe that development in forecasting needs forecasting methodology to be closely linked to the available data and to the environment in which business decisions are made. We would require much closer integration between the information offered by forecasting and the use of this information in optimization and decision-making.

REFERENCES

[1.] Brooks, P. (1974). The mythical man-month. Reading, Massachusetts Addison-Wesley

[2.] CCG (2002). Der Weg zum erfolgreichen supply chain management. Köln Centrale für Coorganisation

[3.] Chatfield, C. (1996). Forecasting in the 1990s. The Statistician

[4.] Collopy, F., Armstrong, S. (1993). Rule-based forecasting: Development of an expert system approach to combining time series extrapolations. Management Science

[5.] Fandel, G., Francois, P.(1998). PPS- und integrierte betriebliche Softwaresysteme. Grundlagen, Methoden, marktanalyse. Berlin, Springer

[6.] Gardner, S. (2004). Exponential smoothing: The state of the art: Part II. Paper presented at the ISF 2004, San Antonio

[7.] Hartványi, T., Nagy, V. (2009): In-sourcing Model for Food Storage and Forwarding. Acta Technica Jaurinensis, Series Logistica. Vol. 2. No. 3.

[8.] Rycroft, S. (1994). Microcomputer software of interest to forecasters in comparative review, International Journal of Forecasting

[9.] Sanders, R., Manrodt, B. (2002). Forecasting software in practice: Use, satisfaction, and performance. Interfaces

[10.] Supply Chain Council (2004). Supply chain operations reference model (SCOR) 6.1. www.supply-chain.org

AUTHORS & AFFILIATION

1. Zoltán BÁTORI,
2. Tamás HARTVÁNYI

1-2. ÓBUDA UNIVERSITY,
KELETI KÁROLY FACULTY OF BUSINESS AND MANAGEMENT,
INSTITUTE OF ENTERPRISE MANAGEMENT, HUNGARY

ACTA TECHNICA CORVINIENSIS
– BULLETIN of ENGINEERING
ISSN: 2067-3809 [CD-Rom, online]