Important References

☒Important References

Risk Management and Financial Predictive Analytics (FPA)

On this page you will find the key references to a methodology and approach to developing the solution architecture for Financial Predictive Analytics in Banking and Insurance. You will find a brief summary of the proposition below but for a more detailed understanding we would recommend that you follow the links to the important reference material.

The Requirement

From 2003-07 a lengthy debate took place around the question of banks' adoption of Basel II, i.e. their reluctance, and issues around investing in internal models. Regardless of the Basel II approaches that banks adopt or supervisors require, there will always be an economic capital based argument to risk-align capital reserves.

Through the Basel II implementation period in Europe there remained resistance to the

implementation of robust Financial Predictive Analytic (FPA) toolsets in banking and insurance. As is now commonly understood, this process was not enforced by supervisors.

If the financial services industry is to exit the Credit Crunch period in the private sector in as short a time horizon as possible then commitment to FPA is fundamental. Banks are expected to increase spending on risk and compliance software to meet the lessons learnt from the credit crunch and the ongoing requirements of regulations such as Basel II.

The Banks are starting to cross a threshold where the manual, piecemeal, reactive approach to understanding how change affects IT service delivery is no longer sufficient or cost effective.

No-one is tighter on time right now than banking IT units working to respond to the ever more intensive requirements effectively sourced outside of their own organization in banking supervisors, regulators and accounting standards; in specific Basel II and IFRS7.

Predictive Analytics

Most banks know through the expert judgment of experienced personnel (the learned capital of the bank, after all) which macroeconomic factors are leading indicators (and thus probably drivers) of credit risk. Just go talk to the most experienced lenders you can find and ask them what new information they see as a good guide to changing credit conditions.

You will generally find when you test their ideas with a statistical model they have been doing it reasonably right for years! This type of view, of course was a key premise of the Rational Expectations (RE) hypothesis which underpinned the introduction of Monetary Theory as a guiding light to the Monetary Policy driven Economic Strategies of Margaret Thatcher and Ronald Regan.

The RE ideas developed from Quantitative approaches to macroeconomics discovered after the Second World War, in The Cowles Commission in the United States. These quantitative approaches have a long intellectual history, they are not a ?weirdo-science? at the margin of thinking about the banking, and they are crucial part of the mainstream of thinking about the business of banking today.

The development of Computing power to support quantitative analytics for economics and finance commenced at the RAND Corporation, at Santa Monica, California; initially a division of the Douglas Aircraft Company. RAND?s overall product was ?systems analysis? but RAND could put computing power at the disposal of early econometric analytics.

Not long after research began at RAND in 1946, the need arose for random numbers that could be used to solve problems of various kinds of experimental probability procedures. These applications, called Monte Carlo methods, required a large supply of random digits and normal deviates of high quality.

RAND's pioneering work in computing, led to the development of tables of random numbers which have become a standard reference in engineering and econometrics textbooks and have been widely used in gaming and simulations that employ Monte Carlo trials.
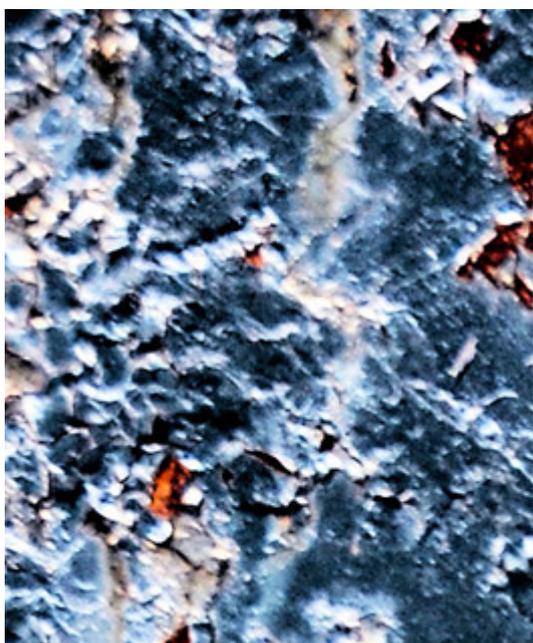
Over the last 50 years, econometric software has developed from complicated sets of computer-specific instructions into widespread easy-to-use software packages and programming languages. In the time of RAND, software was very computer-specific and labour-intensive and programmable computers required substantial capital input.

Commercial econometric software in the US started in Boston at the Massachusetts Institute of Technology (MIT), more specifically at the Center for Computational Research in Economics and Management Science. In the 1970s, code developed at MIT was not really copyright protected; it was built to be shared with the FED and other universities.

Through the 60s and 70s various statistical modeling packages for economics were built particularly at Wharton, the University of Michigan and the University of Chicago (where the Cowles Commission had been located). At Princeton the focus was on development of econometric models in FORTRAN. The use of FORTRAN is much declining now but Chris Sims, now at Princeton, who developed the VAR methodology in an applied manner and was at the forefront of RE in the 1970s now makes all his models freely available in R.

More and more econometricians are switching to the freely-available statistical system R. Free procedure libraries are available for R, http://www.r-project.org [1], an Open Source

statistical system which was initiated by statisticians Ross Ihaka and Robert Gentleman.

The Solution Architecture Challenge

From an architectural point of view, the concept of situation allows us to reason about the external factors that are driving the business. We can not only understand the fundamental sources of information system requirements, we can predict such requirements before the business itself has articulated them. We are looking for factors and trends that call for information system functionality. Is that not exactly where IT development and delivery operations are in regard to IFRS7 and Basel II right now? The requirement blueprint is not being developed by ?the business? (internally); it?s being developed in Basel or in the SEC, at the Fed or the White House or next year in the IMF!

Effectively we already have the requirements model for Financial Predictive Analytics in Banking and Insurance, well specified in the supervisory and regulatory requirements, comprehensively articulated by the Bank for International Settlements and related Supervisory bodies and Central Banks so that we can move directly to the Design model. In predicting requirements and a ?Solution Architecture? to meet banking supervision requirements post Credit Crunch (CC), one thing is for certain only Financial Predictive Analytic architectures are a real response to the requirement, standard BI, just will not do. We also are learning that mainstream statistical software like SPSS simply does not fit econometrics education and research. In academic research in econometrics, SAS has lost ground from its strong position at the end of the 1980s; few modern econometrics textbooks continue to use SAS examples.

Every time one envisions what it would take to have Basel II done ?properly? (and how many of us have done that) you are effectively re-designing a system for advanced risk management. Given, differences in specifics at the margin most respected commentators (referenced on the related pages here) have either the same or similar concepts of what a risk analytics unit will have to look like to completely implement the spirit of Basel II. The new global supervisory architecture can do nothing else than pursue a strategy of intensifying the requirements and powers of the supervisory infrastructure post the inauguration of Obama! Public pressure has motivated senior management to focus on holistic risk management and compliance programs.

Financial Predictive Analytics need to be fully integrated and to go beyond the piecemeal approaches that dominate Pillar 1 implementations throughout the world where risk is quantified at a business-unit or risk category level and aggregated ex post in a naïve manner,

wet thumb in the air, hoping the regulator has a hangover that day!. One has to relate all risks to a common set of fundamental risk drivers, holistically; no longer is a spreadsheet-based solution to risk quantification acceptable, it never was, it was just that the banks got away with it. Now they will not. Thus today you need an holistic approach to modeling risk which can only be supported in an enterprise solution architecture, which must be built up functionally.

Layered Architecture Patterns

The focus of the functional approach is on describing the function of the IT system and is primarily concerned with the structure and modularity of the software components (both application and technical), the interactions between them, their interfaces, and their dynamic behavior (expressed as collaborations between components). System that share a similar high-level structure are said to have a similar architecture style.

The objective is to reuse components to meet different needs. You capture, structure, enrich and distribute data with another end deliverable in mind, you want to try to identify common elements of multiple business requirements across different functions or different business lines and set up a semi-generic deliverable that satisfies multiple needs. This is reuse in action. Approaching reuse without understanding architecture style is like starting to construct a building without knowing if it will be a skyscraper or a garage.

Software of the same architecture style can be reused easier than those of different architecture styles. Software architecture defines the static organization of software into subsystems interconnected through interfaces and defines at a significant level how nodes (sites) executing those software subsystems interact with each other. A good architecture acts as a guide to which components should be reused during the development of application systems. A layered architecture is a software architecture that organizes software in layers.

Each layer is built on top of another more general layer. A layer can loosely be defined as a set of (sub)systems with the same degree of generality. Upper layers are more application specific and lower are more general. The layered architecture is favored for the design of

multi-tiered systems where one layer is built on the services of another. The OSI network architecture is one example of a layered architecture. Essentially both SAP and IBM are using Layered Architecture Patterns to present their respective architectures of SAP General Ledger and Bank Analyzer and IBM IFW and Info Sphere.

Their presentations look proprietary but they are not, they are simply specific presentations of the Layered Architecture Patterns (LAP) generic approach, which is remarkably consistent across the different organizations using it. Microsoft uses the same technique to present .NET compliant architectures, Ericsson uses LAP for development of GSM software etc etc. Where the LAP technique seems to get dropped is when these vendors present SOA pictures.

The Enterprise Data Management Layer

Let?s take the example of IBM (referenced in detail on the related pages here). The tight integration between the IBM Industry Data Models, Rational Data Architect and IBM InfoSphere Information Server allows organizations to exploit industry-specific business and technical metadata to accelerate data integration projects such as master data management initiatives or data warehouse development.

That two layer model is then fused with the COGNOS BI layered architecture to support an overall data management platform or environment which will support Financial Predictive Analytics. This is a world beating if complex proposition, it can get a financial institution off to a flying start in having a quick win because the industry data models arrive out of the box with 25 years of data analysis and deep thinking in the data model; assets farmed from almost

every single one of the great banking names of the past

The questions of the appropriate table structure implementation of ?involved party? or the product type to transaction hierarchy are answered for you and no matter how many planet brains you may have in your data modeling tutorials on semantics, you are unlikely to come up with a data model encompassing semantic logic which will top the IBM banking / financial markets data models.

Assuring that Enterprise Data is available through a service interface is as important as developing the higher level software components. But SOA-ready Enterprise Data Management services aren?t enough for composing enterprise risk management platforms.

The Predictive Analytic Layer
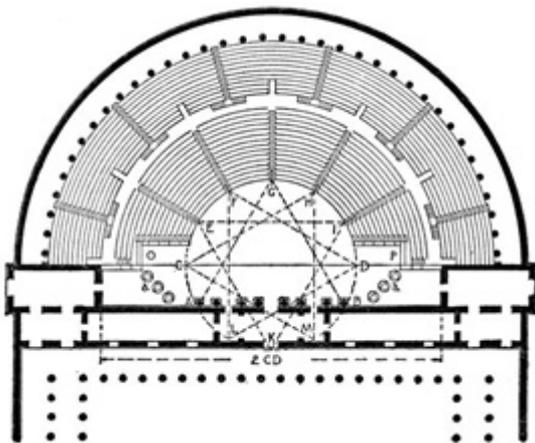
 You need to get into predictive modeling of risk. Risk is conceptually an attribute of our current view of a future state.

Predictive modeling answers questions about what?s likely to happen next. Using various statistical models, these tools attempt to predict the likelihood of attaining certain metrics in the future, given various possible existing and future conditions.

To do that, as quickly as IBM can assist you with enterprise data management you need to leverage a different set of assets, that of the econometric community.

The key business accelerator in Open Source is Community, particularly in Financial Predictive Analytics; since it is via the community in an open source framework that one's initial intellectual capital is gathered or to put it another way that those predictive assets are farmed.

Open Source Predictive Analytics

The commercial open source business model is now relatively mature; the commercial open source vendor addresses issues of open source software which constrain adoption in production enterprise application development by providing support of the code base as a product, making it fit for enterprise production deployment. The commercial open source vendor provides a documented, supported build together with recourse in the event of issues with the software, just like any proprietary software vendor. REvolution Computing is the same in its relationship to R in all these respects. Enterprise Mission Critical needs have very real concerns behind them and REvolution Enterprise-level customer support makes REvolution products suitable for professional, commercial and regulated environments. REvolution provides technical support from statisticians and computer scientists highly versed in the R language and the specifics of each REvolution R build; http://www.revolutionanalytics.com [2]

That is what REvolution Computing provides for R, the Open Source development community environment in which all of the intellectual capital in econometrics since 1946 is embedded. The open source community of worldwide statisticians and econometricians is at the bleeding edge of analytics, but not always creating software that can be set to work in a scale-able fashion, from an IT or production software perspective (it is not designed with that in mind). REvolution is aimed at ensuring that it can ? i.e. ensuring that the best from the research world can be used in production.

A consideration for the CTO today is "If you don't have an Open Source strategy, you better get one" predicated on the 'advantage' to the development process of the forge or community. RedHat's great intellectual achievement is 'Controlled Distribution' of Community developed Objects; it's this which took Linux into the enterprise. It's this concept which the CTO understands, today. We can traction this concept in Financial Predictive Analytics, where its deliverable advantages are most appropriate and relevant, today.

However, there are commercial challenges to managing an Open Source project in-house. Some Banks can do it but it is "new stuff". All too often we ignore the real effects of ?new stuff? on our project timescales. The impacts of technology change on IT projects in general and on the planning process in particular is often underestimated. You need to have the right knowledge and experience about the new stuff; methodology and development techniques; to plan and execute on its implementation. UL has the experience and expertise to help you climb that learning curve and approach the solution architecture with the appropriate agility, by understanding and recognising the challenges.

The commercial challenges of managing high performance computing (HPC) capacity is alien territory to most financial institutions but REvolution know how to do that! Everything about Risk Management requirements in financial services today speaks the combination of the

challenges of three difficult aspects

1. HPC
2. Prediction / Forecasting
3. Community-based Open Source Software.

WEB 2.0 (Click on image below to enlarge!)
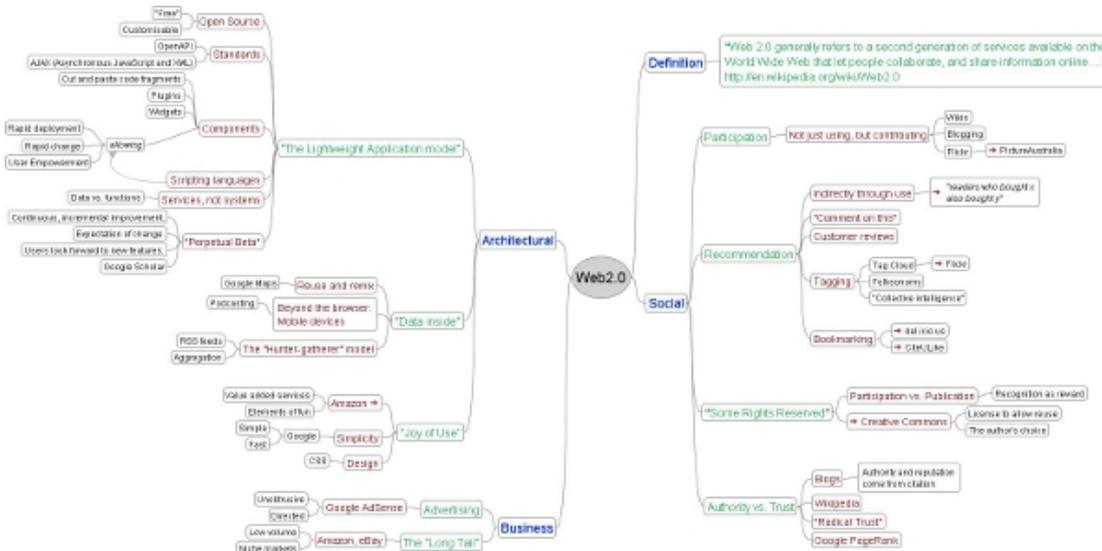
[3]

Computational web services can be viewed as wrappers around the actual Predictive Analytic program Web services provide one approach to model deployment.

The promise of global integration is lost if employees cannot team effectively with colleagues and respond rapidly to global opportunities. Established relationships work well within enterprises, it is the agility to create new relationships and discover unknown capabilities that will differentiate global integrated enterprises.

The Participatory internet hosts interactive commerce sites, blogs, and email services, which capture the wisdom of users by capturing and re-using their interactions.

This is the repository of the intellectual capital of the people of the world; some good, some bad, a lot good! As applications and data are moved to network delivery and storage, the user?s computing platform can be anything that supports a web browser.

The role of simplicity is increasing as the world gets complex. It requires design and discipline, but has growing market value as businesses and consumers seek simplicity and confidence in their support services.

Employees need to be part of the solution and their critical insights and participation need to be actively captured; too often the employee has no opportunity to correct or supplement data, although almost every aspect of their usage provides business value.

Cooperate, Don?t Control!

Source URL (retrieved on *2019-11-14 17:51*): http://www.asymptotix.eu/node/23

Links:
[1] http://www.r-project.org
[2] http://www.revolutionanalytics.com
[3] http://www.asymptotix.eu/sites/default/files/img-john/somerights.jpg
[4] http://www.asymptotix.eu/content/john-morrison
[5] mailto:john@asymptotix.eu?subject=important%20references%20%2F%20asymptotix