Mobiles and the Mainframe – A Perfect Match

David Rhoderick, David.Rhoderick@us.ibm.com

IBM Competitive Project Office

October 2014

Contents

Abstract 3
Introduction 4
e-business 2.0 5
End-user expectations are changing 5
Use cases 7
Companies' digital strategies 8
Balancing new requirements, costs and strategy 9
Potential solutions 10
The incremental cost of supporting mobile devices 13
Potential business opportunities 15
Summary 16

Abstract

This white paper describes the impact of mobile computing on mainframe computing, both quantitatively and qualitatively. For this paper the term "mobile computing" refers to wireless consumer devices such as mobile phones, smartphones and tablets. The potential effect of these devices on Information Technology is enormous, causing a new paradigm to emerge – namely "e-business 2.0" – with significant differences (compared with the original "e-business" paradigm) in the provision of data from Systems of Record such as mainframes. Such differences include far more unpredictable demands for immediate service, 24 by 7 availability, much more stringent demands for timeliness, accuracy and security, and the servicing of a much larger population of consumers, who may have considerably less tolerance for poor service. We discuss these new demands and the related implications in platform choice that IT providers have in servicing e-business 2.0, including costs.

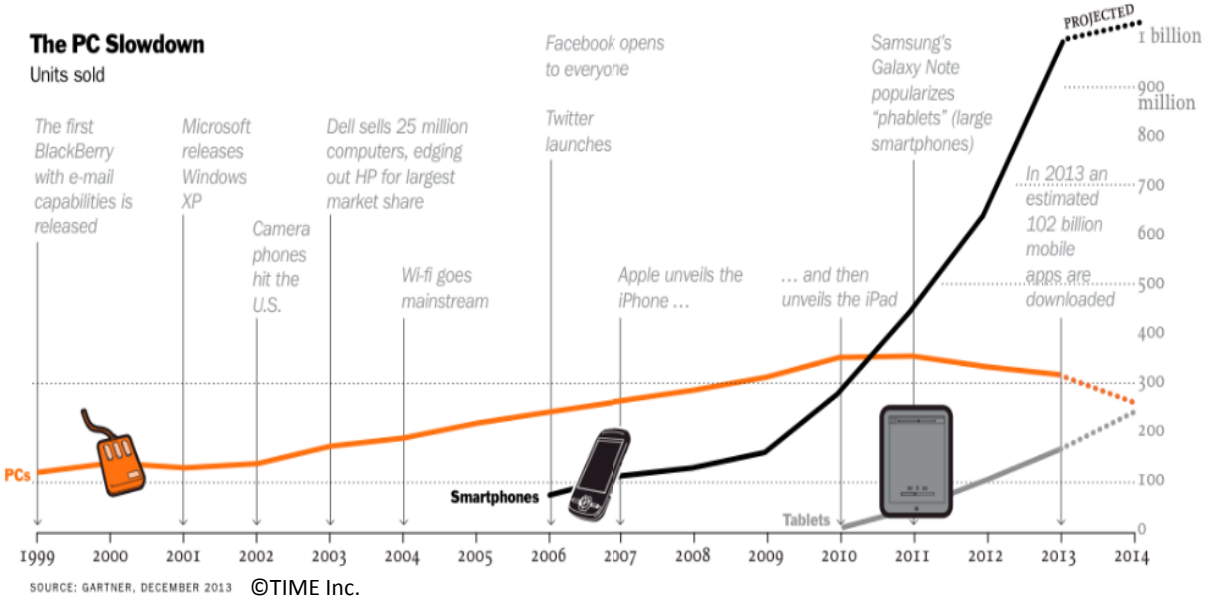
IBM's introduction of a new pricing approach for mainframe software may mark an economic inflexion point where serving data directly from the System of Record is **better** and **cheaper** than using an intermediate tier of stale data. We examine several traditional mid-tier data mart architectures and show that in comparison with modern mainframes, they are less effective (as they cannot keep up with fast-changing data), are more expensive to implement, and have security vulnerabilities.

Now is the time to re-examine the emerging demands of mobile users, to reconsider the capabilities and costs of different architectures, and to rationalize and lay down an extensible infrastructure that can handle the growth projected by "e-business 2.0."

The paper is intended for business technologists, IT solution architects and management.

Introduction

In January 2014, TIME magazine published a brief article "The Death of the PC – smaller devices will take over" along with this compelling chart based on data from Gartner, Inc. The data point



of 1 BILLION[1] smartphones shipped in 2013 is also reported by IDC[2] and, when added to the nearly 200 million tablets shipped, suggests that the mobile install-base is already bigger than the PC install-base at its height. This install-base is growing especially in developing countries where traditional Internet access is sparse. Ericsson, the Swedish mobile phone manufacturer, predicts 4.5 billion devices will be in use by 2016. In the United States it is forecast that 71% (160M) of adults will own smartphones, rising to 78% (179M) by 2016.[3] At the time of writing, new low-cost (US$50) Android smartphones (such as Tecno-Mobile in Africa, Xiaomi in China, Micromax in India) are taking the developing markets by storm, and at the high end, Apple's iPhone 6 pre-orders hit a record 4 million on the first day.[4] Another interesting trend is the proportion of mobile traffic on the Internet in relation to the total global traffic. In May 2014 it was reported[5] to be 25% with 50% compound annual growth rate.

Which industries are these trends most likely to affect? As may be already evident, banks and financial services are already leveraging mobile devices, with the retail industry forecast to quickly become a hot growth area, especially with in-store engagement and mobile payments.

---

[1] The world population is 7.2 billion, at the time of writing.
[2] http://www.idc.com/getdoc.jsp?containerId=prUS24645514
[3] See Javelin Strategy & Research https://www.javelinstrategy.com/uploads/web_brochure/1408.MF_2014MobileBankingSmartphoneandTabletForecastBrochure.pdf.
[4] http://in.reuters.com/article/2014/09/15/apple-iphone-idINKBN0HA1AW20140915
[5] Mary Meeker, Kleiner Perkins Caufield Byers 5/28/2014 http: www.kpcb.com/InternetTrends

A recent report[6] from a major UK bank indicated that the average customer visits a physical branch twice a month and uses the mobile banking app 26 times a month, that they have had 5M downloads of the app since it went live in 2012 and 17.5M hits a week. £4B ($6.5B) is being transacted each month (£1,500/s or $2,500/s). The bank recently announced staff cuts of 1,700 citing the rise of new customer channels, particularly mobile banking, and is currently focused on communicating best practices relating to mobile security.

e-business 2.0

As indicated on the chart on page 1, the height of PC-based e-business ran from the mid-1990s to about 2010. During this period there was also a very healthy growth in mainframe usage – approximately 17% annually. Often a browser-based query or command from a PC is ultimately processed by a back-end mainframe transaction (read, write, update), typically processed by a business application running under CICS or IMS control. The ability of mainframes to scale easily and linearly (unlike other computing architectures), with very high availability, robust security and no loss of integrity to data (especially when using the z/OS operating system), has made them the obvious choice to hold System of Record[7] (SOR) data.

Now let's consider how much larger the potential growth of mainframe computing may be as consumers increasingly enjoy the convenience of their smartphones – their new "essential companion."[8] First, remember that there are now more mobile devices in use than personal computers – so the potential effect of mobile apps accessing the mainframe is huge. Given the fundamental differences – in magnitude of usage, in the change in the client-side technology (hardware and apps), in new consumer expectations (see below), and the requisite changes in infrastructure requirements – it makes sense to name this new phenomenon of mobile access to back-ends and the associated technologies. If we think of the PC-based era as "e-business 1.0", then I propose the term ***"e-business 2.0"*** for the new mobile era. As e-business 2.0 becomes the next phase of information technology, it seems natural and obvious for the mainframe to continue in its role as protector of a corporation's mission-critical SOR data – the availability, scalability, security and integrity of data are still best-of-breed and continually being improved.

End-user expectations are changing

As it is likely that most[9] of the new smartphone users will never have used a computerized interface to their back-end data (whereas most corporate IT-decision makers and influencers

[6] http://www.finextra.com/news/fullstory.aspx?newsitemid=26499

[7] The official, ultimate, authoritative (and often legal) source of the data. See http://en.wikipedia.org/wiki/System_of_record, especially "Where the integrity of the data is vital, a data element must either be linked to, or extracted directly from its system of record. The integrity and validity of any data set is open to question when there is no traceable connection with a known System of Record."

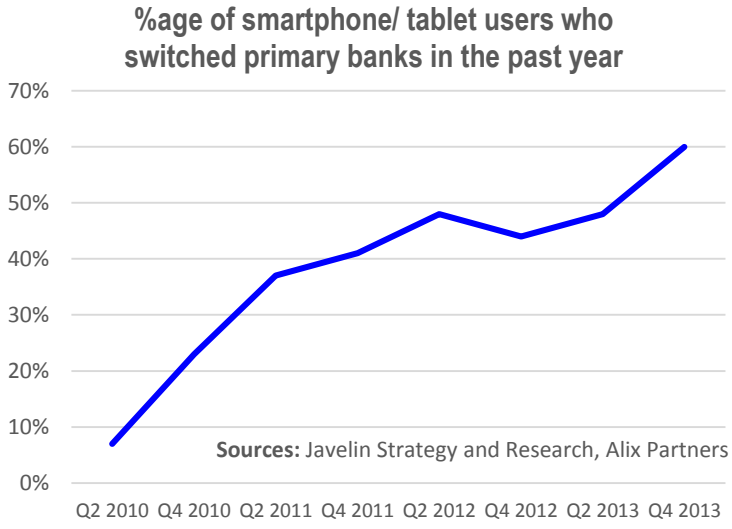
[8] 80% of U.S. adult smartphone users keep their phones with them 22 hours per day and 84% of smartphone users check an app as soon as they wake up.

[9] I.e. the 900M-and-growing gap between smartphone and PC usage.

are very familiar with the browser paradigm), one should be careful not to automatically transfer conditioned attitudes and expectations from e-business 1.0 to the neophyte users of e-business 2.0. Many of these new users have developed a critical dependence on their smartphone device and expect ***dynamic***, up-to-date information (whether for real-time navigation directions, for communication, for news or alerts, for streaming music, for checking in to a flight …) so it should not be surprising that these users expect and demand immediate access to their current account balance or investment position at no charge,[10] anytime, anywhere in the world, and as often as they want.

The ability for the "millennial generation"[11] to check their bank account in real-time (and several times) before finalizing a transaction is real and growing. Response times are being compressed: the expectation among some bank customers to see **accurate account data immediately** (often and at no charge).[12]

Not only is the projected quantitative effect different from and larger than the e-business 1.0 era, but the qualitative effect is also different. Smartphone users expect their devices to be always available and always connected – such is the nature of a phone. Many people still close their browsers and/or power off their PCs at the end of the day. Getting people to do the same for smartphones is almost impossible – as any flight attendant or concert-goer will attest. Instantaneity is expected in this age where major news is now broken first on Twitter rather than traditional news networks and where stock trading response is now measured in milliseconds. And they are not likely to tolerate slow or poor service – studies have shown that 32% of consumers start abandoning slow sites between one and five seconds, and the number one reason why people in the UK switch banks is dissatisfaction with their mobile banking app. Related data was reported in the Wall Street Journal (April 2014) showing that 60% of mobile users switched banks in the past year. The



[10] Over and above the substantial fee for purchase and recurring monthly service charges
[11] The highest penetration (83%) of smart phone usage is in the age group 18-29, see http://www.pewinternet.org/fact-sheets/mobile-technology-fact-sheet/.
[12] Another indication of today's instantaneity is the email acknowledgement **within seconds** of the depositing of a check using a bank's mobile app.

conclusion is that smartphone users are impatient and intolerant of poor service – they will switch to a superior provider if they can.

Another differentiator of e-business 2.0 is the jettisoning of a browser-based user interface. While e-business 1.0 consumers used PCs to access remote data, primarily using browsers, e-business 2.0 with its focus on evolving smartphone technology requires a less rigid interface. Anyone who has tried to browse using a small smartphone will understand why a traditional browser is not viable for regular usage – the ergonomics are quite different (form-factor, aspect ratio, visual angle, input device, sensors…) – and hence we see the resulting phenomenon of rapid growth in highly customized "mobile apps" for smartphone access to SOR data.

In a well-separated scalable architecture, the back-end applications should be unconcerned with the vagaries of the user interface, but easily connected through a stateless middle tier that brokers and assembles information in real-time, without storing any of the SOR data. This new middle tier, sometimes called a Mobile Enterprise Application Platform (or MEAP), such as the IBM MobileFirst Platform, manages the smartphone user interface with separate and secure access to existing business processes and data sources. On today's mainframes, this intermediate platform can even be co-located with the SOR for maximum performance, security and economic efficiency. Access to the back-end is very fast and secure. With the processing speed and data capacities of today's mainframes, consumers can get confirmation of their private data in seconds (perhaps enhanced by an enticing personalized offer for another business opportunity).

Casualties of e-business 2.0 might include *browsers* and the old concept of a *middle-tier data repository*. Both are no longer essential and may join *PC-based client-server* as evolutionary dead-ends.

Use cases

There seem to be three main ways that consumers use their smartphones to access back-end data:

1. *Habitual checking:* regular querying (e.g. daily) of the value of an account balance or portfolio
2. *Transaction-specific checking:* conducting a query prior to, or after, making a specific transaction (e.g. depositing check using the smartphone camera and scanning software) – the frequency may be sporadic but the information delivered must be real-time
3. *Anticipatory checking:* continual real-time querying in expectation of an important asynchronous account update. Examples are the wiring of funds in/out of the account, the clearing of a check or expected regular payment (income), the cancelling of a check, the reversal of a transaction, the recalculation of a portfolio's valuation, etc.

This last use case becomes pathological when a *mass-event* occurs. If the user population is large (e.g. consumers, citizens, benefit recipients), and the anticipated payment time is the

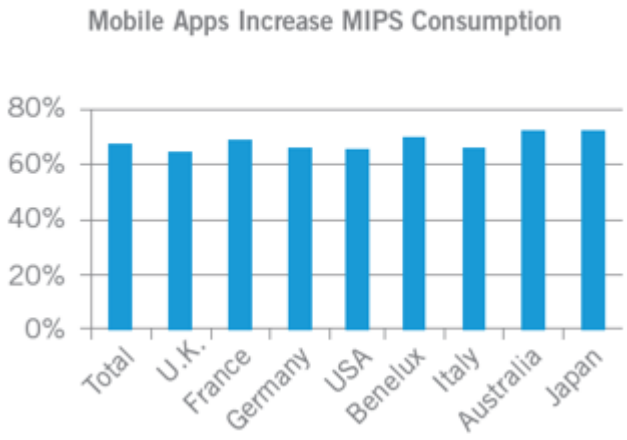
same for everyone (e.g. midnight on Thursdays), there is a serious likelihood for a sudden SOR demand spike due to mobile queries originating at the same time (e.g. 12:01 am). See the BBC News article "Mobile banking apps recover from glitches"[13] where it was reported that *"the problems with the apps were due to too many people trying to check their accounts to see how much they had been paid by their employers"* and "*We are currently experiencing record usage of our mobile banking app. Over 5,500 customers are logging on every minute*."

The problem is compounded when the SOR data is not up-to-date – in some cases it has been observed that users will keep issuing the same request until they see the data change. We label this use case the "sticky finger" effect.

Companies' digital strategies

Business are quickly exploiting the new opportunities that e-business 2.0 offers – 68% of companies surveyed are expecting more social/digital interaction in the next 3-5 years and 54% expect to focus on customers more as individuals.[14] Furthermore "in line with CMOs,[15] four-fifths of CIOs aim to digitize their front office to sync with customers more effectively" and 84% of CIOs explicitly mentioned mobility solutions as top of their plans. This is particularly true in financial services companies which are looking for new ways to differentiate themselves from their competition, and as mobile services become essential for a growing number of banking customers. The data and transactions on the mainframe often represent the competitive advantage for businesses – their challenge is how best to leverage access to these critical assets from new mobile channels.

Many CxOs are rebalancing their priorities to focus on customer experience. In light of this increased interest in customer experience, and the competition to recruit millennials as lifelong loyal customers,[16] banks are investing in mobile apps as a free service, and also considering whether the data shown should be instantaneous and accurate or historical.



This potential effect of e-business 2.0 is not lost on many large enterprises

[13] http://www.bbc.com/news/business-26387754
[14] IBM Institute for Business Value 2013 Global C-Suite Survey – www.ibm.com/csuitestudy
[15] Chief Marketing Officers
[16] The opportunity to snag new customers through a superior mobile experience has strong appeal – millennials may not be a very lucrative revenue stream initially but as their careers grow into their peak earning years (and generate more revenue for the bank), the initial attraction to a better online experience may generate loyalty when they are the mostly likely to be shopping around and most vulnerable to switching banks.

– 68% of mainframe users[17] believe that the increase in mobile applications is driving higher MIPS usage. Many mainframe customers have already started experiencing such growth.

Another example of a company's urgency to open their systems to mobile devices comes from the following statement prescribing a new system: *"smartphone ownership now exceeds 50% of adults. Without offering the functionality desired by our customers there is a significant risk of falling behind in the market where competitors are already offering these services … we will fail to attract sufficient custom to achieve our target growth in market-share. The smartphone channels offer a true alternative in many instances to the internet self-service channels."*[18]

Balancing new requirements, costs and strategy

If mainframe organizations want to make SOR data and the associated transaction processing available to their customers through the new ubiquitous smartphone channel, a critical question is "what's the best technology to do this?" Before answering, let's summarize the requirements:

- Fast response with up-to-date data – "how current (or stale) should the data be?"
- 24 by 7 availability
- Ultimate privacy and security
- Ability to develop, connect, and provision through multiple service channels (traditional face-to-face, phone, VRU, web, mobile) using the same data source
- Elastic scalability to quickly respond to unpredictable usage demands
- Ease of development and extensibility
- All at minimal incremental cost

A short-term problem that some companies have expressed is the risk that initial introduction of smartphone mobile applications will not yield an immediate payback. For example, providing a balance look-up service may be expected to be free, but may not always lead immediately to a transaction (that can levy a commission charge).[19] Much of the SOR data is still stored on mainframe systems, because this is where the respective core-banking processing takes place, and the data is very securely protected and backed-up. Because each "read" of this data requested by a mobile app requires mainframe processing, there ***may***[20] be an increase in the peak demand on the system.

[17] In September 2012, the independent research firm Vanson Bourne (on behalf of Compuware) interviewed 520 CIOs from large enterprises across a range of industries in the U.S., Europe and Asia.

[18] From an anonymous customer document. Also note the differentiation between e-business 1.0 and e-business 2.0.

[19] Not a new phenomenon – one of the new business models in e-business 1.0 was in the travel and insurance industries: new services offered automatic comparison of fares, driving up the so-called "look-to-book" ratio.

[20] Note that additional costs are only incurred when the Sub Capacity Reporting Tool (SCRT) peak for the month is established. Otherwise no additional costs are incurred (the utilization is in the "shadow" of the peak.)

One of the strengths of the mainframe is that it can easily and quickly scale to accommodate increased demand, but there may be an associated additional cost if the system gets more heavily used. This is typical of the nature of infrastructure build-out – when capacity is exceeded, more resources must be added. Typically, incremental unit cost is much less than initial unit cost and pays back over time. So now might be a good time for a business to revisit its overall IT infrastructure strategy with respect to e-business 2.0. If the business has decided that the growth of mobile technology is a fundamental, long-lasting change which is critical in their engagement with consumers, then shouldn't the IT infrastructure be adapted to provide for future needs? Infrastructure investment is intrinsically future-looking, so a long term financial justification is in order. Investment in incremental mainframe capacity to enable the initial phase of a read-only query app can be reused later for the deployment of more sophisticated transactions – the eventual return will be much more than just the read-access to SOR data.

Potential solutions

Use case #1 (page 7) is the occasional, individual transaction per consumer and randomly distributed across the consumer population throughout the day. Use case #2 is less frequent, specific to an individual, and involves an account update that implies bank charges can be associated with it. This is probably benign and easily implemented using direct access to the mainframe. The problematic, pathological case is #3, the "sticky finger" situation, where many individuals are doing queries frequently for account balances. If this use case can be cost-effectively solved, the other less significant ones should be covered.

The ETL Problem

Several IBM 'Eagle' Studies have encountered serious inefficiencies when companies built distributed warehouses with the expectation of saving mainframe costs – but they ended up paying significantly more in terms of additional software, hardware, labor, networking, and storage. Even more surprising, their mainframe MIPS usage went UP due to the export processing required (in one case a whole mainframe was deployed JUST to manage the ETL/distribution of data to a fleet of distributed data warehouses; in other cases 16%-40% of the peak MIPS were used for ETL transfers). IBM's Competitive Project Office also constructed an ETL total cost case that demonstrated annualized cost impact of more than $2M.

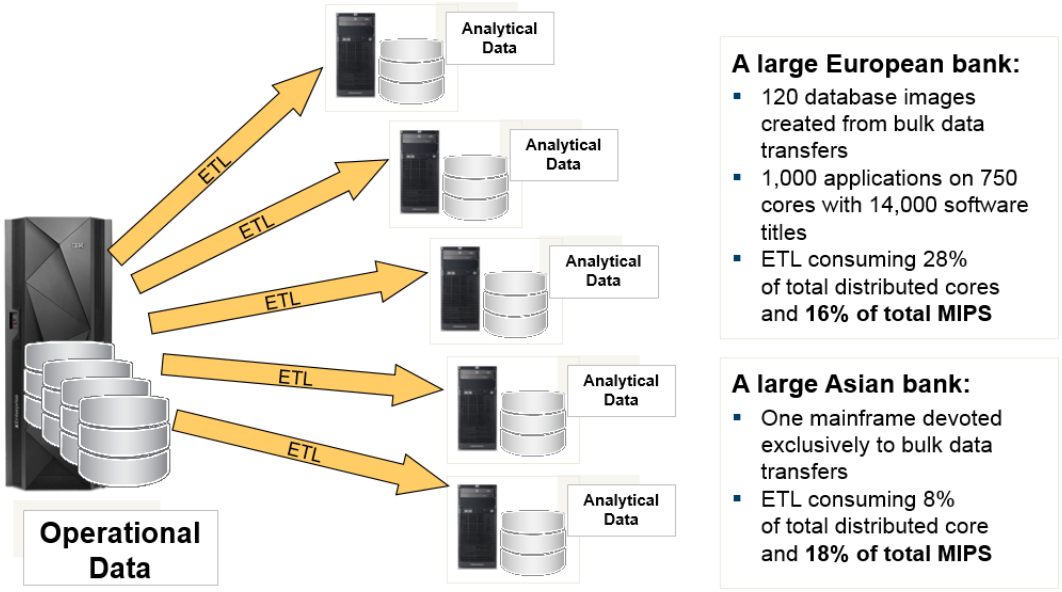
Ultimately ETL is a suboptimal technique as it takes a snapshot of data only at a given timestamp. For dynamically changing data, any decisions that are made based on such a snapshot are not based on the latest information. A Rule of Thumb for data warehouse usage predicts that only 20% of the data being warehoused actually gets used: this implies that 80% of the ETL work is wasted.

In surveying various different architectures to solve the "sticky finger" problem we have found the following common approaches:

1. Access data using the current configuration. This leverages the strength of System z and continues to retrieve the data in real-time from the SOR – in other words, "Business As Usual." Assuming appropriate tuning is done, capacity is available, and workload

management prioritization settings are in place, the system will most likely be able to contain the additional load and deliver very responsive service without impacting other workloads. In many cases, the mobile workloads do **not** drive the mainframe systems to a new peak CPU utilization, thus there is no effect on additional charges (i.e. the 4-hour rolling average that is used for z/OS software charging does not change).

2. Use approach #1, but segregate the read-only data access to a separate LPAR[21] that shares the SOR data, and use techniques such as Active-Query and asynchronous PPRC[22] to contain and manage the cost dynamic of this service. Although the separation of mobile workloads into a different LPAR may appear to be a way to reduce cost, it increases complexity and overhead. Distributing read-only mobile transactions to a separate LPAR means that capacity can be dedicated to that workload, potentially reducing software charges, but this tactic may reduce long-term flexibility when the business introduces new consumer offerings.
3. Offload the data in advance using an ETL[23] approach similar to a classic data warehouse.

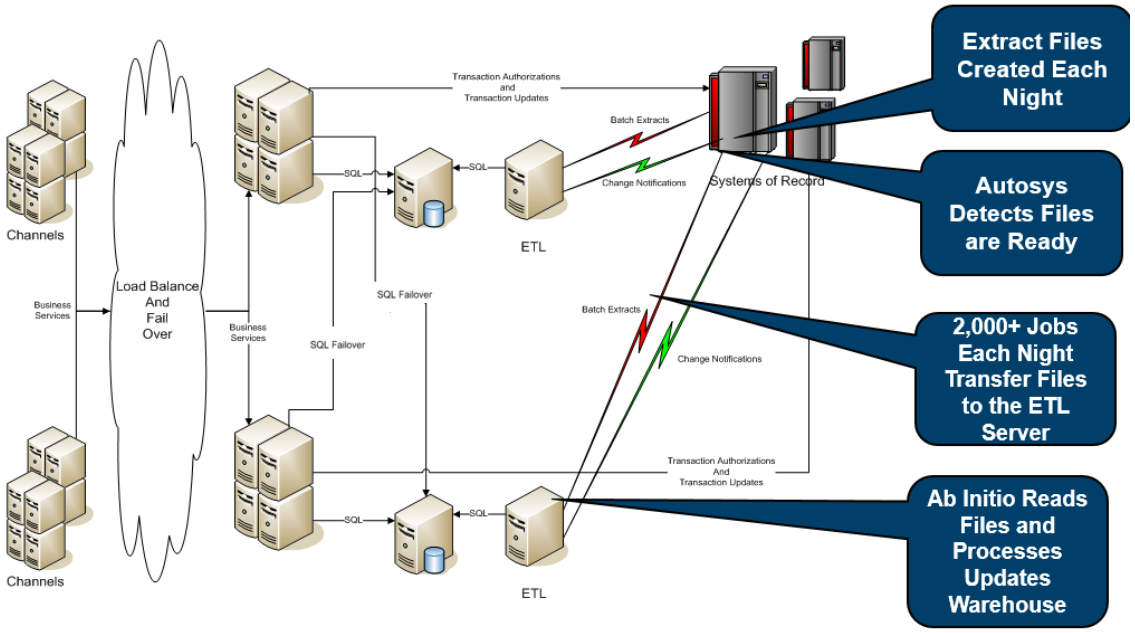


The ETL process is typically run nightly during the "batch-window" involving thousands of batches jobs to be initiated which may drive up mainframe cost by adversely affecting the "4 hour rolling average"[24] and thus increase software charges. Note that ETL is not "on-demand" and, as there is no way of predicting future access to the data, much of the copied

[21] Logical Partition – i.e. a separate system image.
[22] Peer to Peer Remote Copy – i.e. replication of SOR data to another system.
[23] Extract, Transform and Load.
[24] z/OS system software monthly charges are based on the peak smoothed average utilization of the system commonly known as the 4 hour rolling average – or 4HRA. The process to collect and report this measurement uses the SCRT.

data is never accessed, thus making the process inefficient (see *"The ETL Problem"* side-bar). Security of the distributed warehouse must be carefully considered.

4. Add a "trickle-feed" capability to ETL (#3) i.e. do intra-day updates, possibly at the time of update (driven by database logs). This will provide more accurate customer data but adds to the cost and generates additional work for the SOR, which must now send the trickle-feed data throughout the day. Inefficiency increases as, depending on the frequency of access, updates will be in vain if they are not read before the next update (i.e. data gets updated many times but never gets read). A typical customer architecture is shown here – note the complexity of infrastructure – networking, load-balancing, failover, multiple versions of applications and data, that all need to be secured (consider all the potential attack points), monitored and managed, multiple software packages/multiple ISVs each requiring expertise and labor, many separate servers – all of which drive up cost, rather than reducing it.



Also, consider that although the intent is to reduce cost, the various batch jobs are most likely to increase the peak utilization especially if they are done at night in the "batch window". On the other hand, mobile workloads that access the SOR data directly in real-time, getting the actual true value of the data, may incur no increase in cost, depending on the monthly 4HRA peak.

Although the customer declined to disclose their financial data, they did state that anticipated savings did not materialize, even though the original reason for the project was to offload "expensive MIPS". Another failed objective was to improve the currency of SOR data available to the "channels" (on the left of the drawing) – clearly for much of the time the data being disseminated is out-of-date.

Several IBM Eagle studies of mainframe installations have shown that the intermediate caching of data costs more than direct access to the SOR. A significant hidden cost is the cost of the mainframe to manage and move the data to the intermediate tier – the MIPS being consumed range between 15% and 40% of total capacity.

Trickle-feed could also be triggered by technology such as Change Data Capture,[25] but again, this is likely to increase overhead and over-provision fresh and unrequired data which never gets used.

5. Use a distributed cache that holds a copy of the relevant data, which is fetched **on-demand** from the back-end data store. Practical implementations may consider an in-memory distributed data grid technology such as IBM WebSphere eXtreme Scale.[26] The cache is accessed in **read-only** mode by the smartphone, not written to, because although caching technology allows for updating of the grid, any such updating would compromise the integrity of the SOR data. Security of the distributed cache must be carefully considered especially because of its distributed nature and the possibility of network intrusion. IBM's Competitive Project Office investigated the performance and cost characteristics of an on-demand cache and concluded that in most cases it is more expensive. The only case that on-demand caching ***may*** save money is when there is a lot of reading the same data (as in the "sticky finger" effect) – i.e. consumers repeatedly looking at information they have already seen. It is unwise to design an architecture based on one extreme use case and ignore all other use cases and requirements. A better approach to avoid the "sticky finger" situation is to "push" update messages to smartphones in real-time.[27] This approach involves the transmission of a notification message to the smartphone device or intermediate temporary cache (which could be cloud-based) when the SOR data has been updated – referred to as "push" rather than "pull" and may be the most efficient. End-users will quickly get used to relying on the system to alert them instantly,[28] and desist from "sticky finger" behavior.

All the solutions listed have elements of sub-optimization because consumers have differing access patterns and frequencies of access (some users seldom or never, some several times per day), so each account has a different frequency of update (some bank accounts are very active per day, others very inactive over the same time period). Without careful consideration, it is quite likely that a solution can be devised that costs more to implement than it saves.[29]

The incremental cost of supporting mobile devices

The addition of a mobile channel to a mainframe infrastructure may, or may not, have an immediate effect on mainframe costs. As already discussed, it all depends on when the monthly

[25] http://en.wikipedia.org/wiki/Change_data_capture
[26] Or similar technology from other vendors
[27] See "Mobile Design Patterns: Push, Don't Pull" http://www.redbooks.ibm.com/abstracts/redp5072.html?Open
[28] Recent experimentation with a major US bank shows that the latency of notification can be 10 seconds or less
[29] A large mainframe customer deployed an ETL solution only to discover, too late, that it cost $8M (and growing) but only saved $2M annually

peak utilization occurs and which workload(s) drive that peak – for many businesses the cost of new mobile-initiated transactions may be zero as the resulting workload is contained within the sub-capacity cap defined by the monthly peak of the 4HRA (which is often set during a nightly batch window).

Mainframe computing is an infrastructure-based concept – its true payoff is derived from growing volumes of transactions, and its main justification lies in the anticipation of a business that will grow over time where the cost per individual transaction diminishes as more transactions are processed. Companies that plan massive expansion (in consumers, in transactional volume, and variety of increasingly sophisticated applications) should consider investing in a mainframe infrastructure to deliver on their business plan. Doing so may allow them:

1. A consistent level of service, regardless of system load (to maintain superior customer satisfaction while supporting business growth)
2. The provisioning of accurate customer data with maximum safety, security and privacy (enhancing customer confidence and loyalty)
3. To quickly and easily extend their existing business model to deliver new customer applications (retaining existing customers, attracting new ones). Such application development can now be done using "DevOps" processes.

Over time as mobile workloads grow in the way the first graph (page 4) suggests, there is a legitimate concern that software and hardware charges may grow faster than the anticipated financial return – from read-only queries that do not generate any additional revenue for the business. IBM's new Mobile Workload Pricing (MWP) approach may mitigate the effect by significantly reducing the charges associated with mobile queries if and when the mobile workload increases the monthly peak.[30]

IBM's Competitive Project Office recently studied the potential savings that MWP might create for a large bank that introduces a mobile channel. We compared the distributed cache architecture (Potential Solution number 5 above) with direct access to the mainframe SOR, and found that the cost of the **distributed cache was 83% more** than going directly to the mainframe SOR (for a typical blend of users).

Furthermore we studied how "push" technology could mitigate the "sticky finger" situation: we assumed that anxious "sticky finger" users would stop overloading the mainframe with repeated "pull" requests if they were informed that a transaction had just occurred. Upon simulating the 2 scenarios, we found that a "push" approach using CICS Event Publishing

[30] Specifically, if customers can precisely track transactions originating from mobile devices and submit CPU seconds to IBM, then IBM will use a tool that removes 60% of mobile MSUs each hour. If the mobile workload is a peak workload (it increases that month's highest 4HRA), the customer will see a benefit. If not the customer still gets benefit as the mobile workload is not contributing to peak costs.

technology reduced mainframe costs by 51% (because there were no longer any sudden spikes).

In another related study, IBM compared the throughput characteristics of the MobileFirst Platform deployed on a mainframe (using Linux on System z) and on an Intel x86 cluster (identical numbers of cores and amount of RAM). We found that running MobileFirst on System z resulted in 61% higher throughput and 36% lower response time than using x86 technology for the same workload on the same number of cores. Over 3 years the total cost of acquisition (TCA) per TPS[31] of MobileFirst on the mainframe was 29% lower than the TCA of IBM MobileFirst on the X86 cluster. This led to a 16% lower TCA for the entire solution. Reasons for the mainframe superiority include a more efficient hypervisor, higher system utilization, co-location of the MobileFirst Platform server with the backend data server, and a better caching architecture in System z processors.

Potential business opportunities

Engaging with new, younger customers using mobile technology is a very effective approach to grow business and build loyalty and trust.[32] Rather than considering mobile read-only queries as an unrecovered cost, companies might consider possible strategies and innovations to monetize the mobile channel and invest in extending the mainframe to become a mobile infrastructure. Here are some examples:

1. Cross-sell or up-sell using customized offers based on real-time business analytics on demand. Because the mainframe has the capability to perform analytics in real-time using an IBM DB2 Analytics Accelerator on the real data, it is now possible to make a suggestion based on the initial mobile query to generate a "next best action" (such as a suggestion to move money to or from another account). This could give the business a huge benefit in terms of immediate responsiveness to customers that use mobile devices.
2. Provide different qualities of service based on customer account status – i.e. have Platinum, Gold, Silver, Bronze etc. tiers where a specific customer's status is determined by their account balances and/or fees already paid. Service levels could be adjusted and managed using Workload Management capabilities.
3. Use advertising to recover any additional fees. Other free mobile apps do this (Spotify, Pandora, Weather Channel, etc.) – the model is well-accepted. Customers might pay to disable the advertising or gain additional functionality.
4. Make certain APIs available for other companies to leverage for fee, or as part of a larger business strategy.

[31] Transactions per Second

[32] The reader may want to consider the success of Starbucks, the US-based coffee chain – see "Why Is Starbucks the U.S. Leader in Mobile Payments and Prepaid Cards?" http://blog.unibulmerchantservices.com/why-is-starbucks-the-u-s-leader-in-mobile-payments-and-prepaid-cards/

Summary

For modern customer-oriented mobile applications that access System of Record data at the heart of existing business processes, it is essential that the data is accurate at the time of access for the best customer experience. Middle-tier data storage systems don't satisfy this condition because the middle-tier must access the back-end data at least once per mobile request to guarantee accuracy: this activity costs more than a direct mainframe access that bypasses the middle tier. IBM's newly announced Mobile Workload Pricing makes the cost difference even more beneficial to the mainframe and some companies are already enjoying software savings. It is worth re-visiting existing deployments, with a focus on cost and data freshness, to see if accessing the mainframe directly can improve consumers' experience and enhance their experience and loyalty to a mainframe company's products and brand.

Whilst sourcing the data DIRECTLY from the SOR, it is likely that an intermediate integration platform will be needed such as a Mobile Enterprise Application Platform (MEAP). The IBM MobileFirst Platform[33] is a new capability that manages smartphones in a secure fashion and has adapters to connect to data sources. MobileFirst can run on an IBM mainframe (under Linux on System z) and can then enjoy the benefits that co-location bring (such as coordinated security and low latency access to the System of Record data). Also, MobileFirst can help deliver the benefit of IBM's Mobile Workload Pricing by providing the necessary tagging and tracking while still reducing overall cost.

Mobiles and the mainframe are a perfect match because when companies use IBM mainframes to serve their customers they create the simplest, lowest-cost solution that provides the best customer solution (consistent, fast responses and actual data) with the least risk (best scalability, privacy, security and availability).[34]

[33] Formerly known as "IBM Worklight"

[34] See also "System z in a Mobile World" http://www.redbooks.ibm.com/abstracts/redp5088.html?Open

IBM Corporation
Route 100
Somers, NY10589
USA

Produced in the United States
October 2014