Transforming Retail Customer Shopping Experiences Using Mobile Devices, Open Architectures, and Operational Business Intelligence

Chuck Densinger, Mohamed Dekhil, Riddhiman Ghosh, Jhilmil Jain, Meichun Hsu

HP Laboratories
HPL-2010-36

Keyword(s):
mobile, retail, operational, business intelligence, open architecture, customer experience, personalized, actionable insight

Abstract:
The past 2 decades have witnessed the rapid and widespread adoption of three transformational technologies which are profoundly changing how we work and live: the Internet and internetworked devices, mobile technologies, and advanced data analytics. But despite their ubiquity, none of these technologies has had a significant impact on the in-store retail shopping experience. Aside from the products they sell, the inside of a Target, WalMart or Best Buy store is much the same today as it was in the early '90's. In fact, since the introduction of UPC scanning and computer-generated product signs and shelf tags, there has been almost no significant change to the customer-facing technologies in most retail stores. A few are using wireless checkout (e.g., Apple), self-checkout (e.g., Home Depot), or interactive kiosks (Best Buy, Target), but none of these features significantly leverage the Internet, mobile devices or advanced data analytics. Further, retailers' systems are frequently siloed; data for e-commerce sites is often separate from that for store systems, and CRM systems are not fully integrated across channels. Recent advances in service-oriented architectures (SOA), master data management (MDM) techniques and message-based architectures which de-couple components are enabling evolutionary approaches to more effective integration. But these advances have not moved fast enough to enable a rich mobile shopping experience leveraging an integrated platform. In this white paper, we will explore how it is now possible for retailers to create programs which deliver transformational changes to customer experiences in the store, integrating those experiences across channels, and doing so in a way that is based on modest investments and evolution of IT assets - not wholesale replacement in a "big bang" approach.

External Posting Date: March 6, 2010 [Fulltext] Approved for External Publication
Internal Posting Date: March 6, 2010 [Fulltext]

Transforming Retail Customer Shopping Experiences Using Mobile Devices, Open Architectures, and Operational Business Intelligence

Chuck Densinger*, Mohamed Dekhil, Riddhiman Ghosh, Jhilmil Jain, Meichun Hsu

*Elicit, LLC
708 North First Street, Suite 244,
Minneapolis, MN 44501
chuck.densinger@elicitinsights.com

HP Labs
1501 Page Mill Road,
Palo Alto, CA 94304
{first.last} @ hp.com

Executive Summary

The past 2 decades have witnessed the rapid and widespread adoption of three transformational technologies which are profoundly changing how we work and live: the Internet and internetworked devices, mobile technologies, and advanced data analytics. But despite their ubiquity, none of these technologies has had a significant impact on the in-store retail shopping experience. Aside from the products they sell, the inside of a Target, WalMart or Best Buy store is much the same today as it was in the early '90's. In fact, since the introduction of UPC scanning and computer-generated product signs and shelf tags, there has been almost no significant change to the customer-facing technologies in most retail stores. A few are using wireless checkout (e.g., Apple), self-checkout (e.g., Home Depot), or interactive kiosks (Best Buy, Target), but none of these features significantly leverage the Internet, mobile devices or advanced data analytics.

Further, retailers' systems are frequently siloed; data for e-commerce sites is often separate from that for store systems, and CRM systems are not fully integrated across channels. Recent advances in service-oriented architectures (SOA), master data management (MDM) techniques and message-based architectures which de-couple components are enabling evolutionary approaches to more effective integration. But these advances have not moved fast enough to enable a rich mobile shopping experience leveraging an integrated platform.

In this white paper, we will explore how it is now possible for retailers to create programs which deliver transformational changes to customer experiences in the store, integrating those experiences across channels, and doing so in a way that is based on modest investments and evolution of IT assets – not wholesale replacement in a "big bang" approach.

Shopping Scenario

There are many possible customer interaction scenarios involving store, web, mobile and call center touch points. In this white paper, we focus on a scenario in which the customer is using a mobile device in a store shopping experience.

The scenario is this: a shopper enters the store and launches a mobile phone application provided by that retailer. The application automatically logs on, retrieves customer information (including loyalty program information), and, based on their preferences and loyalty program tier (or customer segment), may alert store employees to the customer's presence in the store.

The customer can check the application for any messages, alerts or offers of importance. It may tell them that their favorite salesperson is on duty (or not), that they have a $20 rewards certificate available, and/or that purchases in a particular department are earning double rewards points today. If the customer has been browsing online, it would enable them to retrieve products they were interested in, and direct them to where in the store to find them.

When the customer is looking at a particular category or product, they are able to use the phone's camera to scan bar code information and retrieve product details, ratings & reviews, check compatibility with other items they've purchased in the past, or get more detailed product information than is available on the shelf tag. The application might allow them to assemble a bundle of items – for instance, an HDTV and its accessories, or a set of skis, boots and bindings – alerting them to product compatibility factors and bundle pricing that's available.

At any time, the customer could use a function on the application to call for sales support agent in the store, or call an at a call center. Many other features might be available, such as product return rates & information from product review services (to augment customer ratings and reviews), or the most popular accessories sold with a particular item, or the amount the customer must spend to get to the next tier or rewards level in the loyalty program.

Because mobile devices store our contacts, this application could take advantage of social networking behaviors. Shopping is a social activity. What if the application allowed the customer to share with selected friends what they're browsing, or what's in their wish list? What if a customer could text or chat with a friend from within the application, sharing images and other product info, to ask which color or style or brand they like?

And, of course, this becomes a powerful real time platform for presenting relevant offers and on-the-spot pricing deals to customers. Backend real time analytics can utilize factors such as customer lifetime value, customer segment tendencies, inventory stock levels, available manufacturer promotions, prior purchase behaviors, sales labor availability, the customer's self-reported preferences and interests, and even social media information, to formulate highly relevant offers in real time. Further, these messages and offers can be woven into a user-experience design in which they are relevant and timely in a way that traditional media and direct marketing – even banner ads and paid search – can never be.

As the customer completes their decision-making process, the mobile application can save their selected purchases under their mobile account or loyalty program account. At point-of-sale, all of the purchase details and customer offers are retrieved electronically and applied to the purchase. There is no need to present offers during the checkout process – which is too late and often annoying to customers.

Further, this type of application could create an expedited checkout, in which the customer has already scanned everything they're buying and approved the credit card transaction. The cashier only verifies that all items are correct, bags their purchases, removes or deactivates any security tagging, and the customer is ready to go.

This type of application could help employees and employers as well. Every customer using the application will be known to the retailer's systems and could be used to alert store employees – they would see who the customer is, what they're browsing, and a basic "dossier" summarizing the customer's past activity, preferences, and loyalty program status.

Further, retail organizations are under constant pressure to reduce labor costs. The availability of sales labor is shrinking as retailers cut labor costs. These mobile applications would enable customers to self-serve much more effectively, and to request service exactly where and when they need it (or for the retailer to anticipate when it is most needed). Further, utilization of call center personnel is enabled, providing retailers with a more efficient labor model. Access to call center employees may also provide customers with a better experience, as these agents can be more specialized, be more consistently trained, and have more tools at their disposal to answer questions and assist customers.

Finally, not only will these mobile shopping applications leverage customer data much more effectively than current marketing systems and tools, they will generate massive amounts of rich customer data as well. Imagine having both web browsing detail data and store browsing detail data, an order of magnitude increase in the number of offers presented and responses observed, and permission from the customer to join (appropriately and responsibly) their social networks.

Mobile and Internetworked Devices

As we can see from the potential shopping experience just described, the three enabling technologies of advanced data analytics, internetworked devices and applications, and mobile technologies are all required to bring it to life. However, the mobile device itself is the key enabler from the customer's perspective. Smartphones and other mobiles are radically changing how we interact with the world in ways the personal computer never could. Like personal computers, they can access the Internet and its vast resources (including data services like Google). But the similarities stop here. Mobiles are ultra-portable, location-aware, personal (more deeply so than "personal" computers ever were), nearly always with their owners, nearly always turned on and connected to a network, and increasingly, aware of (or having access to) details about its owner's life such as family, friends and contacts, schedule, and personal data including identification credentials and the financial ability to purchase. Further, nearly every man, woman and child of age to drive a car and make purchasing decisions owns a personal mobile phone – families may share a computer and an email address, but rarely a mobile phone.

In other words, the mobile device is increasingly a prosthetic interface between the individual person and the Internet and all its resources. We are using our mobiles to mediate between the real world and the digital. Because of this, the time has come in which retailers can rely on the majority of their customers possessing a mobile device. Increasingly, those mobiles are smart devices with data plans. And customer acceptance – in fact, expectation – makes room for businesses to create experiences that utilize the customer's mobile device to enhance their shopping and service experiences.

As mobile devices – and the applications which run on them – evolve, the concept of the mobile as a phone will dissolve into the mobile as a communication interface, a universal translator and connector between the individual and all the people, businesses, institutions and digital services with which they interact. This will usher in payment processing via phone, storage of personal information such as medical records and financial data, and more personal details about individuals' social and business relationships. More and more information about the real world is becoming available, enabling "virtual reality" applications which overlay live images of the real world with data about it – restaurant reviews, driving instructions, the name of the person you just ran into (and their children's names), etc. These overlays increasingly will be verbal in addition to visual. The interfaces to the mobile will include not only earpieces (which exist already), but also glasses, and eventually, ocular implants. This may sound like science fiction, but these technologies almost all exist now, and only await application.

Consumer adoption of these increasingly personal capabilities will be driven by a combination of trust and value. Devices will implement security measures to ensure only their owners can access personal data. Service providers will have to earn the trust of consumers by acting on customer data responsibly and only with their permission. In fact, one scenario we've considered is that customers may retain ownership of their aggregated transaction and interaction data, and businesses will have to be granted permission to use it, rather than businesses housing customer data inside private data bases, inaccessible to the customers to whom it refers (and, perhaps, rightfully belongs).

While this may sound overwhelming, the good news is that the basic technologies required to deliver initial mobile shopping applications all exist today. All that is lacking is vision and careful attention to application architectures. The iPhone, Android phone and other similar products have taught us that experiences and applications can be adopted and become standard expectations extremely quickly if usability and value to the customer is high. As soon as one retailer introduces this kind of experience, consumer demand for it to be available everywhere will spread rapidly.

One gap does exist in bringing this experience to retail shopping: the ability to reliably identify physical products via a mobile device. Phonecams require macro lenses (for close-up focusing) in order to read conventional UPC codes on products – a feature increasingly commonplace. Phonecams without macro lenses can read optically-readable codes such as QR or Tag codes, but most likely improvements in optical processing will make these new standards unnecessary. Other technologies, such as RFID tagging, are also a possibility. Advances in optical processing and product recognition will eventually enable applications to use human-readable data such as product names, brand names and model numbers.

In any event, the applications and infrastructure architectures should be designed to expect and tolerate evolution of these technologies.

Advanced Data Analytics

Implementing these experiences will also require advanced data analytics applied in real time. These advanced data analytics capabilities range from voice recognition and natural language processing to contextual search to real time operational business intelligence. Google, Microsoft and others are developing technologies and approaches to language processing and search, but the field of operational business intelligence remains somewhat open.

Many companies are exploring and developing approaches to the analytics enabling product recommendation and "next best action" engines. These range from agent-based to predictive based on purchase cycles to enhanced collaborative filtering, and others. Each of these approaches has merits, but most are limited by a reliance on batch processing and a product-centric orientation. To have the best chance at making a highly relevant recommendation or delivering a meaningful message, the customer's immediate context must be taken into account. In the moment of interaction we have the best clues as to the problem the customer is trying to solve, or the "mission" they are on (see Figure 1).

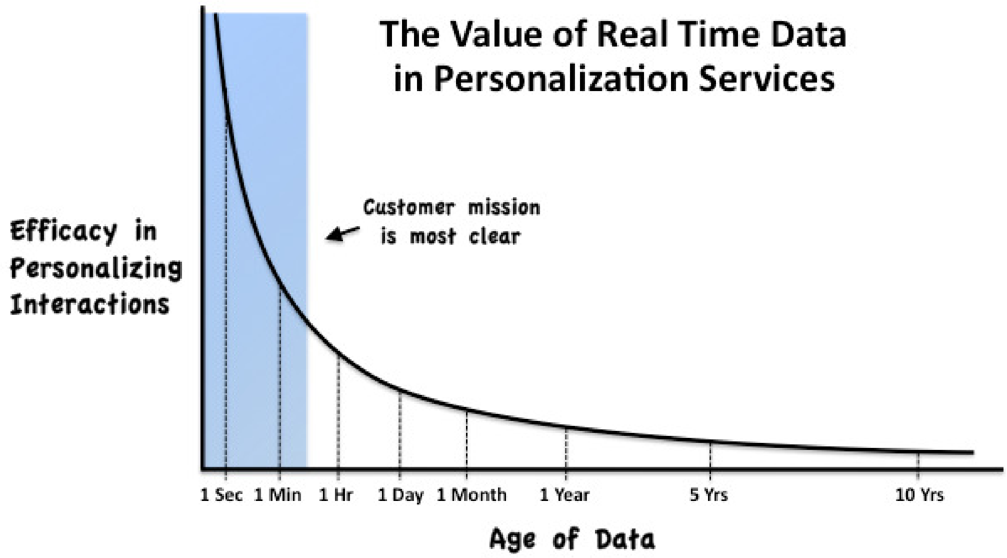

Figure 1. Value of real time data in personalized services

For this reason, real time decision-making will lead to the most relevant possible interactions. The key architectural requirements for real time shopping experiences to work are (1) keeping the analytics close to the master data, and (2) sufficiently decomposing the system into independent components that can be extended or replaced easily.

What is meant by “keep the analytics close to the master data?” Traditional BI systems are data repositories separate from operational or transactional systems, with data and processing architectures optimized for large batch query processing. These BI systems are typically fed by massive ETL feeds in overnight batch processes. Such data warehouses are not up to the task of supporting real time customer interactions. Their deficiencies include:

- Tremendous complexity and latency is introduced when massive data stores must be replicated. These systems are best suited to business performance management, not real time analytics.

- The use cases for real time analytics (such as those discussed above) typically require extensive use of customer master data – facts such as purchase history, stored preferences, web search data, loyalty program information, etc. If the master data and analytics infrastructures are separate, they are unlikely to perform well in real time.

- Typically, performance management systems operate on summary data – aggregations of lower-level atomic data. But personalized messages and offers require access to an individual customer’s detailed data, and that data must be available for analytical processing.

- *Operational* BI systems must be tuned for real time processing, not large batch queries. Sub-second response times are needed.

Operational BI systems must be architected to enable real time processing of customer facts (master data), and algorithmic operations against those facts.

This leads to the second requirement: decomposition. In a real time shopping application, there are various layers of tasks to be completed in the analytic processing which results in a recommended set of actions and messages. These tasks must be isolated as responsibilities to be performed by separate elements in the application system which are coordinated, but which operate independently. Independent operation facilitates evolution of the system, as components are upgraded or replaced. But it also facilitates performance, by enabling parallelization of tasks that do not have to be done serially.

To illustrate, the process of making a recommendation involves taking some educated guesses – much as a human would. Since the data about any individual customer is limited – even the most sophisticated customer intelligence systems have a limited number of customer behavior observations – we must fill in the gaps by applying patterns and inheriting characteristics from aggregates. The customer belongs to a cluster or segment containing thousands or millions of customers. Those customers exhibit statistically predictable behaviors. The customer is also shopping in a specific channel and product category, and there are aggregate behaviors around these dimensions as well which can help fill in the blanks as we are making our educated guesses.

Thus, in the application system, there will be processes responsible for keeping track of behavior patterns of various aggregates – customer segments, categories, geographies, channels, etc. There will be another set of processes responsible for monitoring store resources – inventory and labor, for example. Another set could monitor data around competitor pricing. Another could monitor supplier promotional programs and discounts. Yet another will be responsible for gathering customer facts – preferences and purchase history. And finally, there will be a class of processes (or services) responsible for gathering together the findings of all of these components and resolving them into a single decision.

This sounds complex, and the behavior such a system exhibits will be complex, but each of these components has a relatively simple, clear role, and its behavior can be modified, extended and improved over time without having to re-test and re-implement the entire system.

This approach necessitates new data base architectures. The traditional distinctions between transactional systems and analytical systems no longer suffice. Further, the meaning of the term "data warehouse" may be shifting. Do we still need data warehouses? What function do they serve? Consider the following design "virtues":

- Transactional services (components) need local data
- Decompose large applications into decoupled components, with emphasis on interface stability
- Analytics should be kept close to the primary data
- Summary views of whole entities (customer, employee, store, etc.) are often needed

Historically, analytics have been mostly kept separate from master (or transactional) data, due to their very different workload characteristics. But as analytics (and their resulting insights about master data) become essential to how businesses operate, it will be increasingly necessary to weave transactional and analytical processes together. Further, the results of analytics will increasingly become extensions of master data. More on this below.

This may lead to a "virtualization" of the data warehouse, in which some analytical tasks become "transactional" – operational – and others remain "after-the-fact reporting" – business performance oriented. Definition and orchestration of these tasks into roles will augment traditional data modeling, and will lead to new data warehousing architectures.

SOA, MDM and Message-Based Architectures

Application logic and data go hand-in-hand, and so this topic has already been touched upon in the previous section. Further, much has been written about services-oriented architectures, their reliance upon sound master data management practices, and the use of message-based architectures to facilitate de-coupling. We will focus here on the application of these concepts to the kind of systems that will enable the store shopping experiences described above, and some of the novel requirements it places on such architectures.

Further, we will explore some ways in which this style of architecture can accelerate the introduction of these experiences without wholesale replacement of existing application infrastructures, including POS and website systems, CRM, pricing and inventory, etc.

Three characteristics of this class of applications (mobile shopping and real time personalization) drive some unique features:

- Intolerance for latency at the "edge"
- The bewildering onslaught of heterogeneous micro-events
- The rate of change in customer expectations exceeding traditional IT delivery rates

At the user interface, whether it's a website, mobile application or CRM system, it is unacceptable for response times to grow significantly due to the insertion of real time analytics. Customers and employees are on a mission, and personalization – at least for now – is an enhancer, not the enabler of the core mission. This forces the contemplation of architectures that enable personalization processing to happen asynchronously, and for its results to be optional. That is, applications at the edge must be able to present default experiences when personalization is not enabled, not available, or does not arrive fast enough.

Add to this the fact that there is a growing quantity of information – call them "micro-events" – about customers and their interactions with a retailer (or relevant to it). Between web browsing, mobile activity, social networking, sales and support contacts, purchase activity, and 3rd party data sources ranging from applications for auto loans to home purchases to high school graduations, the quantity and variety of data points about customers is exploding in availability, complexity and heterogeneity. The most successful marketers and customer relationship managers will learn to ingest this data in ways that make it intelligible and actionable. This demands event-based architectures with extensible event detectors and processors.

Finally, the rate of change in customer expectations, driven by product and service innovation, has clearly exceeded traditional corporate IT delivery rates. It typically takes 18-24 months for corporate IT departments to deliver a major new capability. Yet in that timeframe, a phenomenon like Twitter can be born, explode into relevance, and be overtaken by yet another innovation. Traditional IT approaches and legacy systems architectures simply cannot respond fast enough to changing consumer expectations. Yet with customer data security concerns, payment card industry standards, and similar demands for security and reliability by shareholders, large companies cannot cut corners or relax standards. Further, they are hamstrung by the complexities of their legacy applications. This condition demands a new approach to delivering systems capabilities.

The promise of SOA is to address at least part of this challenge. De-coupling and "componentizing" creates networks of micro-applications that work together to accomplish larger tasks. If new functionality is needed, a new component is added. If business requirements change, or new technologies become available, only affected components need to be replaced. In fact, a well-architected, well-orchestrated SOA system can, over time, tolerate replacement of every single element of hardware and software while preserving the overall architecture and system functionality.

The further requirement introduced by the need to keep response times fast at the edge, is careful design of a message-based architecture that is largely asynchronous. Said another way, the application system should be activated and fueled by the constant flurry of event messages, with processes responsible for monitoring those event streams, and recognizing when action is called for. When events of interest occur, real-time analytics can be invoked, and personalized messages or offers generated and published, with the appropriate edge applications subscribing to those messages and acting on them.

This dual publish-subscribe pattern enables a number of things. First, edge applications can publish micro-events and receive recommended actions independently. The stimuli can come from any source – the edge application itself, a 3rd-party data source, social network feeds, etc. – and be sent to any source. It also enables the introduction of new use cases into the system without every sub-system having to know about them. For instance, if the UI has been written to handle offer messages in a specific structure, and a new type of offer is created based on a new type of data feed becoming available, the edge application may require no new coding for the use case to be realized. Careful role definition and separation enables the introduction of new roles and components, and the clever application of existing capabilities to new use cases.

Underlying all of this is a network of data which must be carefully managed. Master data is managed first and foremost by governance rules and processes. That is, business and IT stewards of key data must decide which applications, components and subsystems have primary ownership of various key elements of data, and orchestrate how they are coordinated to keep the whole integrated and representative of the true state of the business and its relationships. In a SOA environment, this includes data definitions, business rules, interface specifications, and process orchestrations. [Much exists in the literature on MDM governance and practices. We won't say more on this topic here.]

Real time analytics adds a dimension to traditional MDM. When analytics is integrated with operations, two types of customer master data emerge: natural and synthetic. Natural facts are just that, facts about customers such as name, address, income, age, etc. – facts which naturally and simply inherit from the world. Synthetic facts are calculated, synthesized or derived, based on business rules and defined procedures, and include things like cluster or segment assignments, lifetime value, historical shopping frequency, etc.

To support real time personalization, the concept of master data management must be expanded to include management of synthetic facts – where they are stored, which processes calculate them, how frequently the calculations are refreshed, and the business rules defining the calculations. The requirement to keep analytics close to the data arises from this new type of master data.

Again, this can sound dauntingly complex. Yet the simple idea inside this approach is that events of interest in the environment call for specific responses: (1) decide which scenarios (use cases) are to be handled; (2) ensure that micro-events signaling the occurrence of these use cases are published; (3) set up a complex event processor which subscribes to the event streams, and signals the occurrence of events of interest; (4) build web services which respond to messages that an event of interest has occurred, and which calculate the best response; (5) publish the desired response; and (6) instrument the UI applications to listen for messages telling them a personalization action is needed.

Figure 2 depicts a conceptual view of such an architecture.

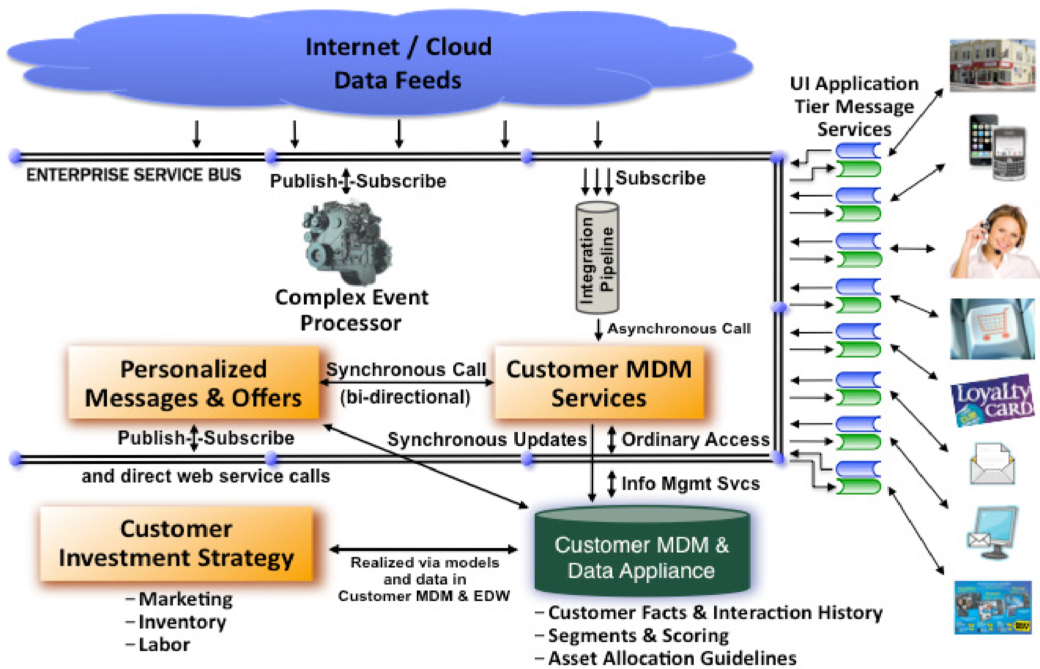

Figure 2. Conceptual view of the architecture

Designing a Program to Transform the Retail Shopping Experience

Discuss implications for the ecosystem of IT providers – what does corporate IT do, and what kinds of partners are needed? How do we recruit the community to do reviews, generate UI concepts, etc.? Who opens what data to whom? What role do the telcos play?

The foregoing describes a comprehensive vision for the underlying capability required to deploy a highly-aware and continuously sensing customer intelligence system to enable highly relevant and content-rich customer-facing applications, such as the mobile shopping app we've been discussing. We recognize that this idealized view is far from reality in today's businesses – along dimensions of business practice, information technology and organizational culture. However, we believe it's possible for organizations to migrate in this direction through evolutionary steps which are affordable and which deliver value at every stage – to customers and shareholders – making such a migration viable.

The first step is one of vision and commitment. Leaders must examine market position and assess the potential value of introducing a customer relevancy capability such as described above. Particularly for retailers, consideration must be given to new competitors in the marketplace, especially online and mobile commerce, plus hybrid business models. They must also consider current customers and their likelihood of adopting technologies such as mobile shopping, as well as the ability of their brand to extend to that type of experience. Finally, a clear and publicly stated commitment to this direction is necessary, along with an ongoing change management effort to inform and involve employees, and to shepherd the necessary business changes.

Secondly, the organization must think through and design the desired changes to the customer experience. This should be a living evolving process that becomes part of the business's ongoing planning and operations. This customer experience blueprint should be richly informed by direct customer feedback, and should be publicly available inside the organization – and perhaps shared with customers as well. Why not create an online forum in which customers and employees collaborate on the evolving customer experience blueprint? This blueprint should describe both the facts and the emotions desired in each customer interaction – that is, it should say what will happen, as well as how the organization wants the customer to feel. It should realistically consider cost-benefit trade-offs. For example, customers may want a dedicated sales agent to focus completely on them in the store, but that may be cost-prohibitive. Finally, it should clearly articulate differentiated experiences for different customers and their varying needs, as well as their greater or lesser value to the enterprise.

The third step involves detailed operational and capability planning. The organization must lay out the roadmap of changes to its operating model, which will include setting detailed business objectives, developing new measurement and performance metrics, and developing the phase plan for implementing changes to capabilities, including people, process and technology. This latter step must be guided by the customer experience blueprint and associated expected benefits. That is, the organization should prioritize and sequence the implementation of new capabilities in order to evolve the customer experience in meaningful steps towards the articulated blueprint.

In order to create accountability and establish leadership with respect to such a program, it is recommended that businesses create a "customer business unit" – a new organizational function that is responsible for managing the customer dimension of the business, analogous to that of product or channel management functions, such as Merchandising and Field Operations. This new business unit will guide both planning and operations around the customer.

With respect to the evolution of technology, it is critical to have IT architects and senior leaders dedicated to this effort. The customer experience blueprint will provide guidance on the key capabilities required as the business evolves. The key underlying infrastructures will include information management services and customer data integration (CDI) to enable the real-time services and applications, and of course the services infrastructure, including an enterprise service bus and services registry

Another key layer is the message taxonomy and message interaction model. This is an extension of the data model. It lays out the types and structures of messages being published by various processes in the environment. Goals of this aspect of the system include maximizing independence between components, ensuring clarity of roles between components in the system, and isolating behaviors so that components can evolve and new functionality deployed without requiring changes throughout the system.

In the midst of all of these layers of planning and design, there are some practical, simple steps that can be taken. Experimenting with a mobile shopping application will enable both the customer and the firm to learn what adds value and what doesn't. It will force the practical solving of theoretical problems. And it will ensure that value delivery begins quickly. It is entirely reasonable to take shortcuts in technical and business architectures while getting these early applications online, as long as those shortcuts are taken consciously and with a plan to remediate later when more robust infrastructures become available.

In short, a moderated blend of careful long-term planning interleaved with rapid experimentation and learning is better than too much of one or the other. Service-oriented architectures on top of robust information management infrastructures provide a foundation for flexible, evolving application environments. These technologies can co-exist effectively with legacy applications. Engineering both ETL (batch) and services interfaces to legacy applications and data stores is becoming relatively straightforward, and permit the co-existence of contemporary and legacy application architectures.

The key is to get started, and to ensure there are both business and IT leaders dedicated to the work. Significant customer and business benefits can be realized from fairly modest efforts, which will bolster the resolve of management to continue the journey of transformation.