

**Universal Bank Business Context**

Universal Bank is a midsize, traditional bank that has acquired a complete range of financial services capabilities through a series of acquisitions. The bank currently has a limited online presence that is fragmented across its various divisions. As part of its expansion strategy, Universal Bank has decided to innovate in the online banking market by providing a host of value-added services on top of a fully integrated financial management capability. Some banks currently offer, or are developing the capability to offer, integrated online banking across all accounts (such as savings, checking, and credit cards). But no other bank offers a full range of value-added services such as financial advice, financial analysis and planning, and tax planning and filing. In addition, Universal Bank has not seen the synergies it anticipated from offering a complete line of products. This lack of synergy is caused by its inability to effectively cross-sell based upon existing relationships and customer knowledge.

One of the key initiatives involves strengthening the bank’s online presence. The online channel is an effective means of adding new services at relatively low cost. With many customers using online checking and savings today, it will be easy to move them to other value-added services if the bank can make the online services work better together. A key new service is the addition of an online bill payment feature.

The chief executive officer (CEO) decided the first step was to immediately add an electronic bill payment capability to the current online banking system. This would allow customers to schedule electronic payments online from their checking accounts — a high demand feature providing greater customer convenience. The CEO believed this added convenience would have an immediate impact upon customer satisfaction and loyalty, while demonstrating tangible progress to his board of directors. To initiate this effort, the CEO brought in his chief technical officer (CTO) and the vice president for consumer banking and asked them to deliver this capability before the end of the fiscal year. He expected rough-order-ofmagnitude (ROM) cost and schedule estimates within six weeks.

**Value Proposition for Online Services**

Universal Bank ’s value proposition for online services is to provide an integrated,

online financial portal for affluent clients that will enable them to view all of their financial assets at once. From this integrated view, clients will be able to eamlessly

execute all of their financial transactions including those associated with savings

and checking accounts, loans, stock trading, domestic and international bill payment and funds transfer. Clients will use this portal to save their most precious commodity: time. Through efficient operations, the bank will be able to offer this capability to clients without irritating service fees for each transaction

Providing this integrated service to customers will allow the bank to provide a

higher rate of return on assets than other banks. Focusing on high-net-worth individuals means that the bank can manage more assets with fewer transactions.

Fewer transactions result in lower total operations cost. Providing an integrated portal to customers enables effective cross selling from low margin service offerings (such as savings accounts) to high margin service offerings (such as loan and portfolio management services).

A compelling online user experience will drive adoption and attract new customers,

including high-net-worth individuals and lower income, mass market accounts.

Although the mass market accounts are less profitable, they will more than cover

costs; the higher margins, however, will come from the wealthy individual segment.

After the user experience is developed for high-end customers, it will be easy to

offer a reduced-service version for the mass market.

**Requirements**

The CTO immediately involved a senior program manager to create a project around this initiative. Unlike many projects, the CTO expected to not only gather requirements from the consumer banking division, but to also negotiate requirements with the consumer banking division based on the overall needs of the business.

As he reflected on the overall initiative, the CTO felt confident that the business

would continue to invest in additional financial services for its customer base and

that additional acquisitions were likely to follow. This was clearly not an isolated

initiative; rather, it reflected a longer-term strategy for the company. He realized it

was important to have a well-conceived technical architecture at the enterprise level that would smoothly support these corporate goals.

Beyond the functional requirements that would emerge, he wanted a solid technical

foundation that would allow him to meet operational requirements as well. He

pulled together an architecture team and asked them to create an architecture that would support this initiative and future initiatives. As a first approximation, he started with the following high-level requirements and constraints:

● Build a Web-based online banking portal that allows customers to pay bills online from their checking accounts.

● All account-related transactions will use the current system, which resides on an

IBM mainframe using Customer Information Control System (CICS) based transactions.

● The online bank system will reside in the corporate data center in Seattle, Washington. It will be connected to an acquired bank’s data center in Los Angeles,

California though a private leased line.

● Loan information will be pulled from the acquired bank’s loan systems, which

reside on systems that are based on J2EE.

● All customer profile information will use the current Customer Relationship

Management (CRM) system. The CRM system has encapsulated its functionality with a Web services. This system don’t have support.

● Domestic electronic payments will use the current payment system, and international electronic payments will use SWIFT-based transactions through an external payment gateway. Payees that cannot receive electronic payments will be paid using electronic transactions to a manual fulfillment center, which will then make the payments manually through the U.S. mail.

● The system’s overall transaction rates, concurrent users, and response time must meet the first year’s projected usage plus an engineering safety factor of 3x (or three times the first year’s projected usage) to handle burst load.

● The system must meet or exceed the service level agreement (SLA) for our current online system.

● The system authenticates the customer by using information retrieved from

the directory server .This process allows the identification, authentication, and authorization of the customer in the Web application

Use Cases Principal

As the team dissected the high-level requirements of the chief technology officer

(CTO), the members of the team arrived at the following use cases:

1. View Scheduled Payments
2. Execute Scheduled Payment
3. Receive Payment Response
4. Schedule Payments
5. View Scheduled Payments Use Case

The first use case they discussed was the View Scheduled Payments use case.

This use case involved a portal that allowed users to see their account information,

including their current account balance and a list of scheduled payments. To build

this portal, the team would need to connect to multiple back-end systems and to

aggregate the results in a single view.

To implement View Scheduled Payments, the portal would have to display the

following information:

● Account information from the mainframe

● Profile information such as name and address from the Customer Relationship

Management (CRM) system

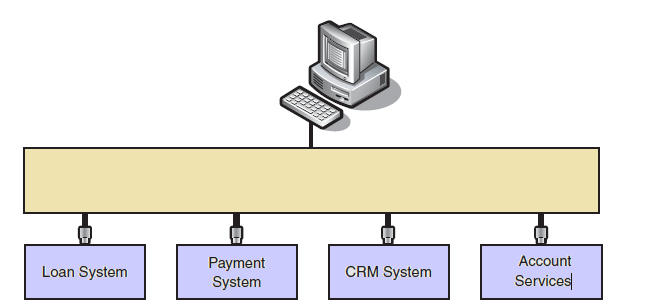
● Scheduled payment information from a payment system.

Optionally, the portal would have to display any other loans the customer might

have with newly acquired banks so that the customer could submit electronic

payments toward these loans. These systems are located in a remote data center.

.



The following is the flow of the use case

1. A customer browses to the online bill payment application.

2. The system prompts the customer for a user name and password.

3. The system authenticates the customer by using information retrieved from

the directory server.

4. The system sends an request to the related loans systems.

5. The system retrieves customer profile information from the CRM server.

6. The system retrieves the customer’s mainframe account number from the

payment server.

7. The system retrieves account balance information from the mainframe.

8. The system retrieves a list of scheduled payments from the payment system.

9. The system builds the presentation, which displays account balance, scheduled

payments, and customer profile information.

11. If loan information is available, it appends this optional information to the

presentation.

.

1. Execute Scheduled Payment Use Case

The next use case they considered was the Execute Scheduled Payment use case. To implement this use case, the system would:

● Start up at a system-defined interval.

● Retrieve the set of payments to be made on or before the current date.

● For each payment, the system would verify that there were sufficient funds in the

payment account and then debit the account for the payment amount.

● Send the payment to an appropriate payment channel.

There were four kinds of payment channels currently in scope: domestic payments

through a clearing house, electronic payment gateways using Society for Worldwide Interbank Financial Telecommunication (SWIFT) transactions, electronic payments to a manual fulfillment house, and account-to-account transfers within the bank’s internal system.

The following is the flow of the use case:

1. A system scheduler initiates this use case and begins to execute the payment.

2. The system requests the list of payments to make from the payment

system.

3. For each payment, the system checks the account balance in the

mainframe. The system debits the account if sufficient funds exist.

4. The system retrieves the appropriate security credentials for the

message exchange.

5. The system sets the routing information, transforms the message to the format understood by the recipient, and then sends the message.

1. Receive Payment Response Use Case

The Receive Payment Response use case described the behavior of the payment

Gateway and the manual fulfillment house after they processed the payment request.

In this use case, these payment channels returned the result of their processing to

Universal Bank ’s system. If the payment was successful, the payment status and transaction ID were updated in the payment system. If the payment failed, a compensating transaction to credit the account was first issued to the mainframe and then status and ID fields were updated accordingly in the payment system.

The following is the flow of the use case:

1. The precondition for this use case is that a payment message has been sent to one of the payment recipients: the SWIFT payment gateway, the manual fulfillment

partner, or an acquired bank.

2. After processing the payment request, the payment recipient sends a payment

response to the system.

3. The system correlates the response to the originating request.

4. If the payment failed, the system credits the customer account on the mainframe.

5. The system updates the payment record in the payment system with status and transaction ID.

1. Schedule Payments Use Case

This use case allows customers schedule yours payments associated with various commercial services. For this, the client can select the type of service to pay, the company and associated billing code and selected the date on which he want to perform the automatic debit.  
  
These payments may be either nationals or internationals