“Our e-banking solution is an enterprise-banking system, architected using distributed-services, fabricated using components which minimize total-cost-of-ownership and delivers scalability, business-model-flexibility, robustness and ease-of-use.”
System Architecture & Components

Sys Architecture

Database Engine (DE)
- Implements Factory Method pattern to support multiple databases (Oracle, SQL Server, DB2 etc.)
- Database Connections Pooling for database connect speed
- Prepared Statements for execution speed
- Stored Procedures for execution speed
- GUIDs for inter-location uniqueness

Process Engine (PE)
- Pipelines of “code-lets” for re-use, plug-in modifiability, testability & logic extendability
- Components can be Java Classes, Servlets, Stateless Session Beans
- Can be extended to support TCL, PERL5
- CDF as data sharing & interchange format
- Pipeline expansions and collation as part of the “Build”
XML vs CDF

XML Advantages
- Many ready tools
- Global Standard

XML Disadvantages
- Parsing, search & tags-addition overheads

CDF Advantages
- Excellent Speed
- Prior experience for us

CDF Disadvantages
- Proprietary Format
- Need to write tools
- Even vendors such as Versata, TIBCO, ILOG use proprietary formats for rule-engines
- Speed is the most vital criterion for us
- XML is supported through an XML-Adapter, which can also support specific XML-flavors such as OFX, FIXML etc.

So Our Choice is CDF

Concepts
- Subject. Something Pelican stores info about. EG: Book, Cashflow, SB, Term Dep.
- Activity. Something that changes Subject-Info. EG: Open, Close, Post Credit, Maturity
- Source. Origin of an Activity. EG: Customer at Home Branch, ATM Switch, RBI Clearing House, VISA Electron Switch, Customer at Other Than Home Branch

Business-Logic = f (Subject, Activity, Source)

Workflow Characteristics
- Multiple Participants. EG: Each transaction has a Maker and a Checker (not same).
- Multiple Locations. EG: ABB transactions involve Branch-1, HO, Branch-2.
- Multiple Timings. EG: Outward Clearing instruments are released 2-3 days later
- Conditional Additions. EG: If amount > X, it could need one more authorization.
- Components. Workflow \rightarrow Process(es)
  Each Process is a “pipeline of components”.
Example.
Close Savings Account Workflow.
- Subject = SB (Savings Account)
  Activity = Close
  Source = Customer at Home Branch
- User-1: Validate-Account
  Validate-Balance
  Calculate-Last-Period-Interest
  Mark-for-Closure
- User-2: Authorize-Closure
- User-3: Print-Repayment-Instrument

Workflow Engine (WE)
- Implements Factory Method pattern to support multiple clients (HTML, XML, IVR, ISO8583 etc.) via Adapters
- Generates WFQ (Workflow Queue) based on Subject, Activity & Source
- Automatically sends JMS(CDF) to other Location’s Messaging Engine (if Subject’s Location is not same as Home Location)
- Manages individual’s Work-Queue & tracks Activities to Completion
- Implements MVC (Model-View-Controller) using “struts”
- Manages common functions: Audit-Trail, Replication, Numberings & IDs,
- Manages “In-Transaction” and “Delayable” Processes (such as “Statistics-Update”)"
- Manages Session-Variables (Singleton pattern)

Interchange Engine (IE)
- Implements Factory Method pattern to support multiple messaging middleware products (MSMQ, JbossMQ, MQSeries etc.) with JMS as the abstract layer
- Implements “near-synchronous” remote updates using messages
- Implements Secure & Guaranteed Delivery using Store & Forward Messaging
- Provides a method of implementation using Message Beans

Other Components
- Security Engine (SE). Implements Application Security as well as mechanisms such as PKCS, DES3
- Rule Engine (RE). Provides parameters & rules needed by various Pipelines; has a special module for “propagation”
- Timed-events Engine (TE). Generates timed-events for Contracts Maturity, SWIFT Messages, etc.
Unattended Processing

- Pelican has its own Batch Processing Environment
- Any Process can be (if sensible) run in “unattended mode” using Process Parameters
- Catalog of multiple instances of Process Parameters per Process
- Facility for Temporary Stop and Re-Start
- Automatic Error Reporting (mail, SMS)
- Framework for Problem Escalation
- Includes Automated Backups, Interest Calculations, Contract Maturities, etc.

Platform

<table>
<thead>
<tr>
<th></th>
<th>HO Database Server</th>
<th>Linux, Oracle 9i</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branch Database Server</td>
<td>Linux, MySQL or MSDE</td>
<td></td>
</tr>
<tr>
<td>Application Server</td>
<td>Jboss (includes TOMCAT)</td>
<td></td>
</tr>
<tr>
<td>Messaging Middleware</td>
<td>JMS, Jboss MQ</td>
<td></td>
</tr>
<tr>
<td>Language/Framework</td>
<td>Java J2EE</td>
<td></td>
</tr>
<tr>
<td>MVC Framework</td>
<td>Struts</td>
<td></td>
</tr>
<tr>
<td>Config. Mgmt. &amp; Build</td>
<td>ANT, CVS</td>
<td></td>
</tr>
<tr>
<td>Testing &amp; Bug Tracking</td>
<td>Junit, LoadRunner, bugtraq</td>
<td></td>
</tr>
<tr>
<td>Others</td>
<td>log4J, Swing (for Traders), Live HTTP Sockets, Java Web Start, taglibs, XML, Crystal Reports</td>
<td></td>
</tr>
</tbody>
</table>
Product Architecture

- Product = Service-1, [ Service-2 […]]
- Savings Account
  - Transactions Service
  - Cheques Payment Service
  - Cheques Collection Service
  - Credit (Deposit) Service
  - Surplus Balance Sweep Service, etc.
- User selects which Services are to be included for an Account
- A Service-charge can be associated with a Service
- Services included in an Account determine the Activities possible in the Account
- For example, if an account has “Cheques Payment Service” not included, Cheque Book Issue will not be allowed.
- Default Services-Bundle is defined for each Product; these can be over-ridden in individual Accounts.
Contracts Accounting
- Common back-end for any “pre-determined exchange of Cashflows”; future transactions “Scheduled” accordingly
- EG: All types of Term Deposits, Term Loans, Bills Discounted etc.
- Flexibility in respect of associating one or more Actual Transactions with one or more Scheduled Transactions (many-to-many)
- Flexibility for the bank to define its own Products

Payments Engine
- Definition of “Instruments” parameterizing Drawees, Creation Method, Presentment Method, Settlement Method etc.; can replicate any real-life instrument (such as Cheque, DD, MT, TT, Gift Cheque, Warrant etc.)
- Tracking of Settlement Cycle for each Payment
- Support for Multiple Clearing Houses
- Settlement can be done through Nostro/Vostro Accounts

Retail Solutions
- Equation - iSeries (IBM)
- Bankmaster
  - Open Systems (Unix, NT)
Money Market
- Deal Input (Fixed DP / Loans)
- Amendment / Verification
- Maturity Options
- Automatic Rollover
- Reports

Foreign Exchange
- Deal Input
- Position Account Keeping
- Amendment / Verification
- Profit & Loss
- Reports

Automated Settlements
- S.W.I.F.T./TELEX Message Production
- Incoming / Outgoing Messages
- S.W.I.F.T. Statements
- Default Settlement Instructions
- Meridian Link
- S.W.I.F.T. Alliance
Reconciliation
- Allows reconciliation between “our” and “their” ledger via SWIFT 940/950 messages
- Meridian Match creates matches
- Allows discrepancies to be cleared
- Delivers a true balance on currency A/c
- Is generally operated on a daily basis

Standard Reports
- Management Reports e.g.
  - Cash Flow, Dealers Ladder, Status List,
  - Today’s Maturities, Loans/Deposits Analysis

- Operational Reports e.g.
  - Audit Trail report
  - Interest Application report
  - SWIFT report

- Log Reports e.g.
  - Interest Rate Change Log file
  - Easidata Log file, Security Log
  - Limits Maintenance Log file

- Correspondence e.g.
  - Statements, Confirmations, Advices,
  - Demand Notes, SWIFT
Financial Returns

- Production of P&L, B/S, Trial Balance
- Accruals
- Variety of Types
- User definable coding structure
- Collection Program

Operational Data
- Retail
- Corporate
- Treasury
- Transactions
- Fees
- Statistics
- Commissions
- Lending
- Accounting

Business Data

Your Business

Business Information
- Bank-wide View
- Regulatory Reports
- Forecast What-if
Business Performance
Forecast What-if?

Historic, Present and Future Comparisons

1998
1999
2000
2005
% % INTEREST RATES % %

INDIVIDUAL RATES
\( \text{eg} \ 5\% \ CR \ on \ Deposit \ A/C \)

BASE RATES
\( \text{eg} \ Code \ 01 = 5\% \)

NORMAL

TIERED

\textbf{Apportioned}
- 5\% = First 10k
- 6\% = Next 20k
- 7\% = Next 70k
- 8\% = Anything over 100k

\textbf{Non-Apportioned}
- 5\% = 0 - 10k
- 6\% = 10k - 30k
- 7\% = 30k - 100k
- 8\% = 100k plus

Security
- ID & Passwords
- User Definitions
- User Definable Menus
- Time Out / Lock Out
- Log Files / Audit Trail
Teller Facilities
- Cash Management
- Teller Totals
- Cash Analysis
- Teller Limits
- Teller Journals
- Teller Transfers
- Money Market
- Passbook Printing

Enquiries
- Alpha Search
- Customer Search
- Account List
- Account Details
- Transaction List
- Transaction Details
- Account Notes
Posting & Transaction Definition

- Flexible Transaction Definition
- Transaction Types (direct / batch)
- Commissions
- Exchange Rates
- Receipt Printing
- Passbook Printing
- MICR Supported

Trade Innovation

The fastest growing & most comprehensive Trade Finance system on the market

- Import and export documentary of credits
- Inward, outward and direct bills for collection
- Cash letters
- Reimbursement credits
- Clean banker’s acceptances
- Standby letters of credit and guarantees
- Financing
- Participation of LCs and guarantees

- Flexible accounting and charges
- Workflow management
- Document production

Internet Banking

- Solution for all Markets
  - B2B, B2C
- Rich Functionality
  - Payments, Accounts, Ordering Service, Notifications, Administration, General, Service Administration
- Proven Security
  - Supports standard firewalls, Multi layered approach, Authentication (digital certificates), Integrity, Confidentiality (128, RC4), Non-repudiation, Public Key Infrastructure
- Fully Customisable

ATM /POS

- ATM online to host system database
- TCP/IP connection to host
  - scaleability
  - larger volumes of transactions supported
- POS
  - interconnectivity of systems - shared network
• growth in traffic volumes
• wide range of POS transactions supported

**Investment Management**
• Designed to handle:
  • Custody and dealing operations.
  • Corporate actions.
  • Portfolio management and analysis.
  • Client valuations and reporting.

**Support Objectives**
• Development of software that is released bug-free, works first time & provides the functionality and performance that was intended
• Provision of Customer Support that is fast, courteous, expert and that provides Rapid Resolution to problems
• Implementation and resourcing of the Product Strategy
• investment in our People
• a Profitable and Growing business

**Interest Rates Flexibility**
• Ability to handle:
  Fixed Rates. Fixed for entire contract period.
  Pegged Rates. Fixed for the Interest Period.
  Floating Rates. Vary within Interest Period.

• Separately definable cycles for:
  Accrual. When Income/Expense is booked.
  Payment. When actual money is paid.
  Compounding. When interest is capitalized.
**Charges Flexibility**

- Charges can be:
  - Incidental. EG: Stop Cheque Charge
  - Periodic. EG: Quarterly Folio Charge
  - Reimbursement. EG: Telex Charge
  - Contract Default. EG: Deposit Foreclosure

- Flexibility to define charge amount:
  - Flat Charge. EG: Stop Cheque Charge
  - Tiered with Min/Max. EG: DD Charge; up to 3 tier-variables permitted

**Specific Module for PDCC**

- Specific module for PACS Loans
- Refinance Accounting for MSCB/NABARD
- Reconciliation between Loans & Refinance
- Module for PACS (Marathi Front-end) already done

**Agricultural Credit Flow**