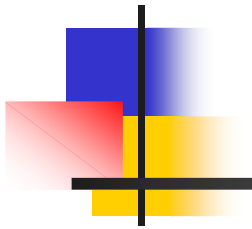
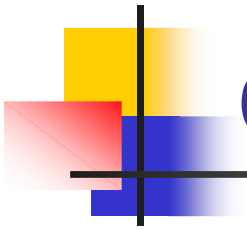


AcademicEdge

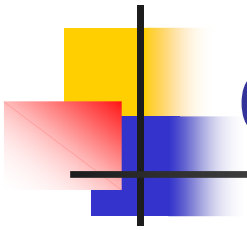
Build a comprehensive information system using best-in-class software for colleges and universities.

Siba Mohanty, Ph.D.



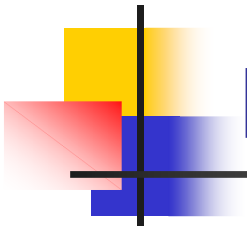
Colleges & Universities

- Diverse student population
- Intellectual staff
- Multiple regulatory requirements
- Non-Profit Status
- Limited budget



Colleges & Universities

- Use home grown or multiple third-party software systems to teach, administer and manage their institution
- Have complex system integration requirements

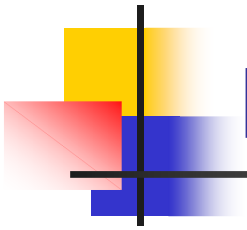


Buy, Make or Integrate?

For Enterprise Applications and Data, the Question Is Not Make versus Buy

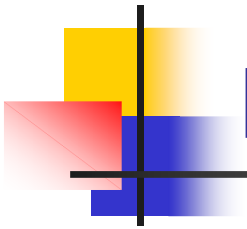
… because, while we can do both, we really need to be good at "assemble" if we're to own our own application and data destinies.

Bob Weir & Rick Mickool
VP & Exec. Dir.
Information Services
Northeastern University
EDUCAUSE Quarterly, (Number 1, 2003)



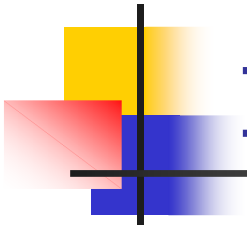
Buy: Pros & Cons

- One-stop shopping
- "…implicit assumption that solution vendors have every thing you need. This is rare in industry and impossible in higher education given the diversity of needs & institutions."
- Forced release migration



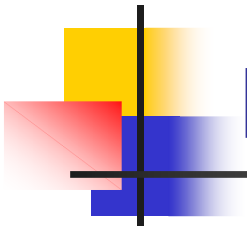
Make: Pros vs. Cons

- Making provides ultimate in control
- Investment
- Discipline
- Support



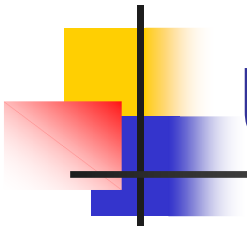
Integrate:

- Ideal middle ground – buy some, build some and integrate



Northeastern University

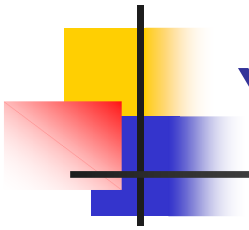
- 50,000 customers
- PeopleSoft, SCT, CA, College Board, home grown legacy apps
- DB2, Oracle, SQL Server, IDMS
- IBM, Sun, Compaq servers
- 11 terabytes of data



Univ. of California

"Buy versus build" is a decision that campus computing officials face each time they must upgrade any component of their administrative-computing systems. "It's a constant thing," and "nothing is pure buy-or-build anymore"

John W. McCredie, associate vice chancellor for information technology. Univ. of California, Berkeley



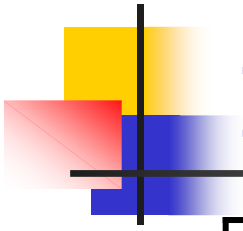
Yale University

Yale … never had great confidence in E.R.P. vendors who promised "a unified, soup-to-nuts system" for higher education. The university has stuck to a strategy of buying and integrating what it considers to be "the best of breed" business software.

Mr. Daniel Updegrove, CIO

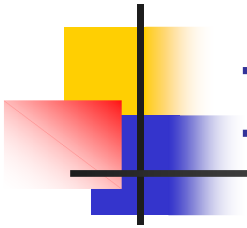
Independent Systems

Admissions	Recruitment PLUS, Liquid Matrix
Bookstore	Nebraska Prizm
Course Management System	WebCT, Blackboard, IBM LearningSpace
Campus Residential	RMS, Adirondack
Collection System	Sallie Mae, infiNET
Dept. of Education	Dept of Ed. Annual Specification Releases, ED Express, SAIG
Educational Planning/Advising	DARS (Univ. of Miami), LAT's of PLA
Scheduling/Events System	Ad Astra, Schedule 25/Resource 25, Corporate Time
Financial Aid	Power FAIDS, Wolfpack, Sigma, SAFE
Accounting/Financial Records	Great Plains, Financial Edge, PeopleSoft, SCT, Datatel, Jenzabar, Sap, Lawson



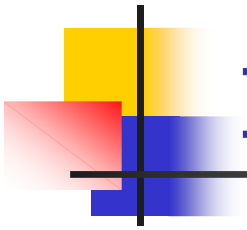
Independent Systems

Groupware	Lotus Notes, MS Exchange, Novell Groupwise
HR/Payroll	ADP Horizon, Peoplesoft, Oracle, SCT
Institutional Advancement	Raisor's Edge, New Millennium, BSR
ID Card	OneCard
Library	Innovative, Digital Library
Procurement System	HigherMarkets
Student Information System	SCT, Datatel, Jenzabar, PeopleSoft, SAP, Oracle
Student Billing	Infinity, others
State Reporting	Texas State Reporting, others
Web Portal	uPortal, Campus Pipeline, CNAV, Timecruiser, EPOS,TouchNet



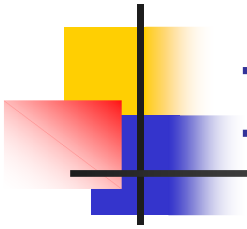
Independent Systems

Food Services	Foodtrak
Campus Security	
Visa	Student & Exchange Visitor Information System (SEVIS)



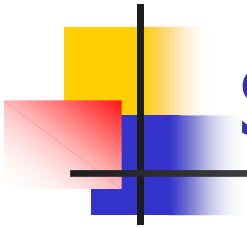
Installed Independent Systems

- MIP (Fund Accounting) – 100
- Best (Fund Raising) – 150
- Blackbaud (Institutional Advancement) – 300
- Power FAIDS (Financial Aid/college Board) - 500
- Recruitment PLUS (Recruitment/College Board) – 150
- SAFE (Financial Aid) – 250
- Grate Plains (Accounting/HR/Payroll) – 100



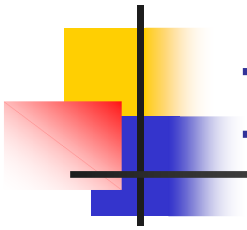
Installed Integrated Systems

- SunGard SCT (Banner, SIS, ABT Campus) – 2000
- Jenzabar (CARS, Quodata, Campus America, CMDS) – 700
- Datatel – 600
- PeopleSoft – 400
- BiTech – 100
- Oracle
- SAP
- DAG (mostly in Canada)



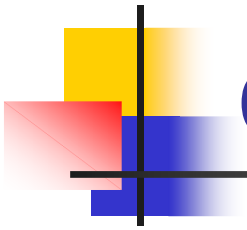
Student & Faculty Requirements

- Learning Tools/Technology for student & faculty
- On-line student services/functions
  - Application, registration, course catalog
  - grade reports, financial aid reports
- Student/staff/faculty collaboration



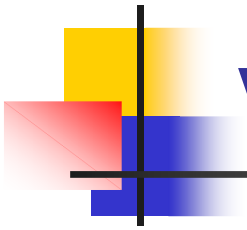
Institutional Business Requirements

- General Ledger
- Bank Reconciliation
- Fixed Asset Management
- Payable/Receivables Management
- Purchase Order Processing
- Human Resources
- Payroll
- Grants Management
- Fund Accounting
- Project Accounting
- Financial Statements
- Financial Reporting
- Secured, web access to business information



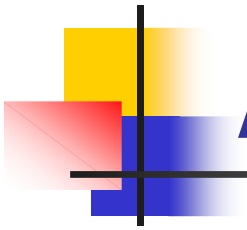
Current Thinking

- Current SIS are expensive, tightly integrated and incur significant annual maintenance cost
- There is significant interest in an open source, open standard Student Information System in the academic institutions
- We have a fully functional system that is based on SOA. It has reusable services for creation & execution of business processes



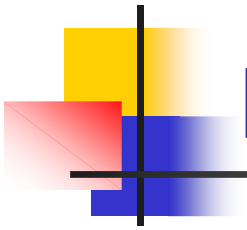
Value Proposition

- Low cost, high payoff SIS built on state-of-the-art technologies
- SOA framework extends current system capabilities at minimal cost
- Built on Open Standard – Java/JEE Technologies
- System requires customization at additional cost



AcademicEdge SIS

- AcademicEdge SIS has the required functionalities
- It can integrate with third party software systems
- Java/JEE 5 Technology Infrastructure is used for deployment & integration



Fully Functional

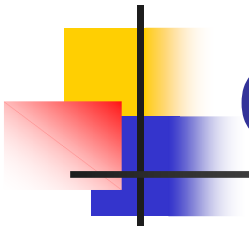
- Includes:
  - Recruitment
  - Admission
  - Registrar
  - Transcripts
  - Financial Aid
  - Bursar
  - Alumni, and
  - Business Reports

AcademicEdge Functions

	Recruitment	Admission	Financial Aid
Student	▪ Request Information ▪ Inquire about degree programs ▪ Inquire about financial aid	▪ View admission requirements ▪ Apply On-Line ▪ Check Admission Status	▪ Apply On-line ▪ View application status ▪ Change information
Staff	▪ Interact with prospects ▪ Maintain and track prospect data	▪ Monitor on-line applications ▪ Post admission decisions ▪ Post policy changes	▪ Monitor on-line applications ▪ Post aid decisions ▪ Post policy changes
Faculty	▪ Provide academic program information to prospects	▪ Make admission decisions ▪ Provide academic advising	▪ Provide provisional information to prospects ▪ Provide satisfactory academic progress reports

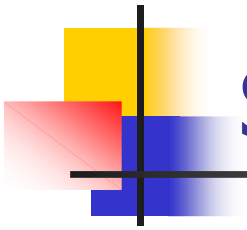
AcademicEdge Functions

	Student Trans.	Registrar	Bursar
Student	▪ View Grades ▪ Request Modification	▪ View course catalog ▪ Register On-Line ▪ Check grades ▪ View academic programs	▪ Check account status ▪ Pay fees
Staff	▪ Post grades ▪ Request transcripts ▪ Receive transcripts	Post ▪ graduation results ▪ grades to transcript systems ▪ policy changes	▪ Draw down funds ▪ Disburse funds ▪ Prepare fund reports ▪ Post policy changes
Faculty	▪ Post grades	Review/change ▪ academic programs ▪ course catalog, grades ▪ Academic advising	



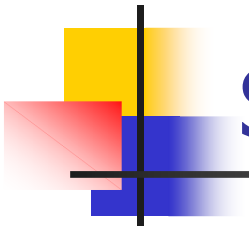
Open Standard Technologies

- Uses Java/JEE 5 technologies
- Service Oriented Architecture (SOA)
- Creates reusable web services to expose system capabilities
  - Uses Entity Centric Business Services to implement institutional activities – admission, course registration, etc.
- Business process are created out of web services & executed in a BPM engine



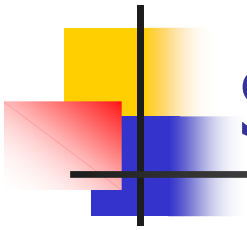
SOA Framework

- SOA Framework allows
  - Design
  - Development
  - Deployment of web services that can be:
    - Distributed across organizational boundaries
    - Independently scaled
    - Reconfigured into new business processes



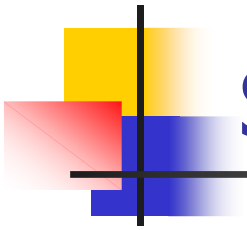
SOA framework

- The framework is used for:
  - Development
  - organization
  - identification
  - management, and
  - monitoring of reusable services using Java Messaging and Service Infrastructure



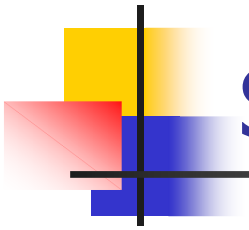
SOA benefits

- Benefits include:
  - re-usable service components
  - Modular
  - Agile/flexible development – adapts to changing business requirements,
  - Interoperable – standard based interface facilitates easy integration
  - Leverage – extend the capabilities of existing systems



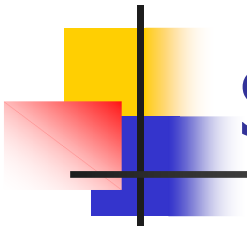
SOA Project – Key Questions

- How to:
  - Build processes that span service enabled applications & legacy systems?
  - Provide required performance while accommodating changes in demand?
  - Isolate applications from faults from server and communication failure?
  - Manage processes that will interact with services across organizations?
  - Manage & monitor the infrastructure, processes and services?
  - Allow to change processes, rules, data mapping and relationships between applications with minimal disruption?



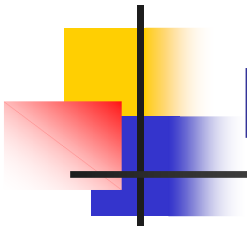
SOA Infrastructure

- Enterprise Service Bus (ESB)
- IDE or Workbench
- Orchestration Server
- XML Server
- Collaboration Server
- Database Server



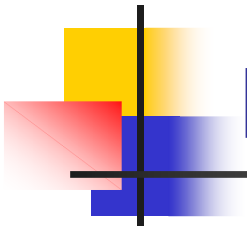
SOA are Composite Applications

- Services are integrated through mediation service and business processes to create business applications



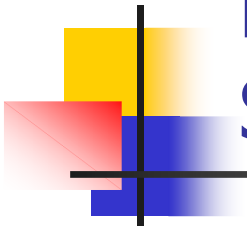
Enterprise Service Bus (ESB)

- Core to IT infrastructure supporting SOA is the ESB. It:
  - Connects
  - Mediates, and
  - Controls all communication & interaction between services
  - Provides error and exception handling.



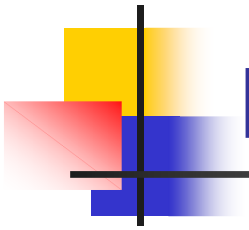
ESB provides mediation service

- ESB provides mediation service to transform and route information among services in a heterogeneous operating and network environment
- These mediation service allows flexibility to integrate new components as services into the SOA environment without changing existing components



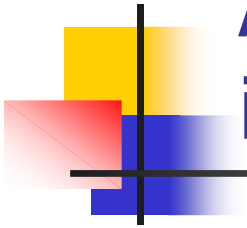
ESB mediates communication among Services

- Communication among services is message based.
  - Request/Response from a service is usually in XML
  - The ESB mediation service transforms these messages as appropriate



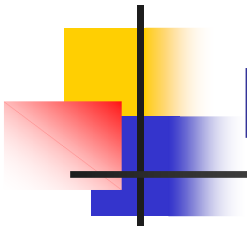
ESB Performs Routing

- ESB supports content based routing from one service to another
- ESB supports interconnection of services in asynchronous/synchronous mode
- ESB help create business process out of services



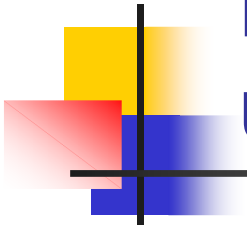
Advantages of transformation & Routing in ESB

- Performing these services in various application requires dependency analysis, coordinated development & deployment
- Could result in application deadlock
- ESB supports loose coupling. Consequently these transformation & routing can be done in ESB from a single console



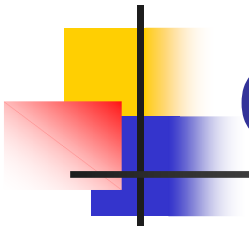
ESB Services

- **ESB Mediation Service**
  - **XML Transformation Service using XSLT and/or JavaScript**
  - **Content based Routing Service – direct messages to other services endpoints using routing rules or JavaScript rules**
- **User Defined Service**
  - **User defined mediation service – data conversion other than XML (to/from legacy apps in proprietary formats)**
  - **User defined routing service**
  - **User defined orchestration service**
  - **User defined business service – implement core business logic**
- **Advanced Service**
  - **Orchestration Service – business process management**
  - **Collaboration Service – B2B protocol, like ebXML**
  - **Workflow Service**
  - **Database Service**
  - **XML Service**



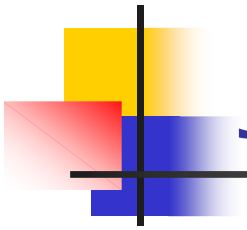
Messaging and Service Infrastructure used in AcademicEdge

- Open-Enterprise Service Bus (ESB) -- Used for loose coupling of web services via standard based technologies and tools
- NetBeans IDE for development of services
- Hibernate O/R Mapping for Persistence
- MySQL Relational database
- EJB 3.0 metadata annotations
- JSF for UI development



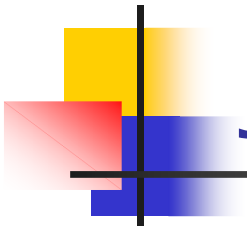
Open ESB

- Open ESB implements an ESB runtime using Java Business Integration as the foundation. It allows to:
  - Integrate enterprise applications, web services as loosely couples composite application
  - Compose/recompose the composite application realizing the benefits of SOA.



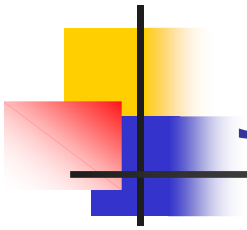
JBI

- Java Business Integration allows a standard (as opposed to proprietary solutions) way to integrate applications and exchange data
- For example, BPEL allows to develop business process to be executed in a BPEL engine. JBI allows a standard way to use a BPEL engine (start/stop/deploy/…).



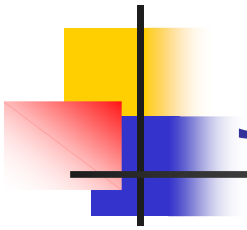
JBI

- Service engine is a runtime process to which a developer deploys an artifact – BPEL code, business rules
- Binding components are bridge to external systems using SOAP, SMTP, FTP, etc.



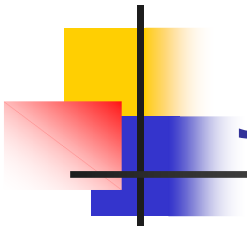
JBI Components – Service Engines

- Service Engines:
  - Java EE Service Engine
  - Intelligent Event Processor
  - ETL (Extract/Transform/Load) SE
  - SQL SE
  - XSLT SE
  - BPEL SE
  - …



JBI Components -- Bindings

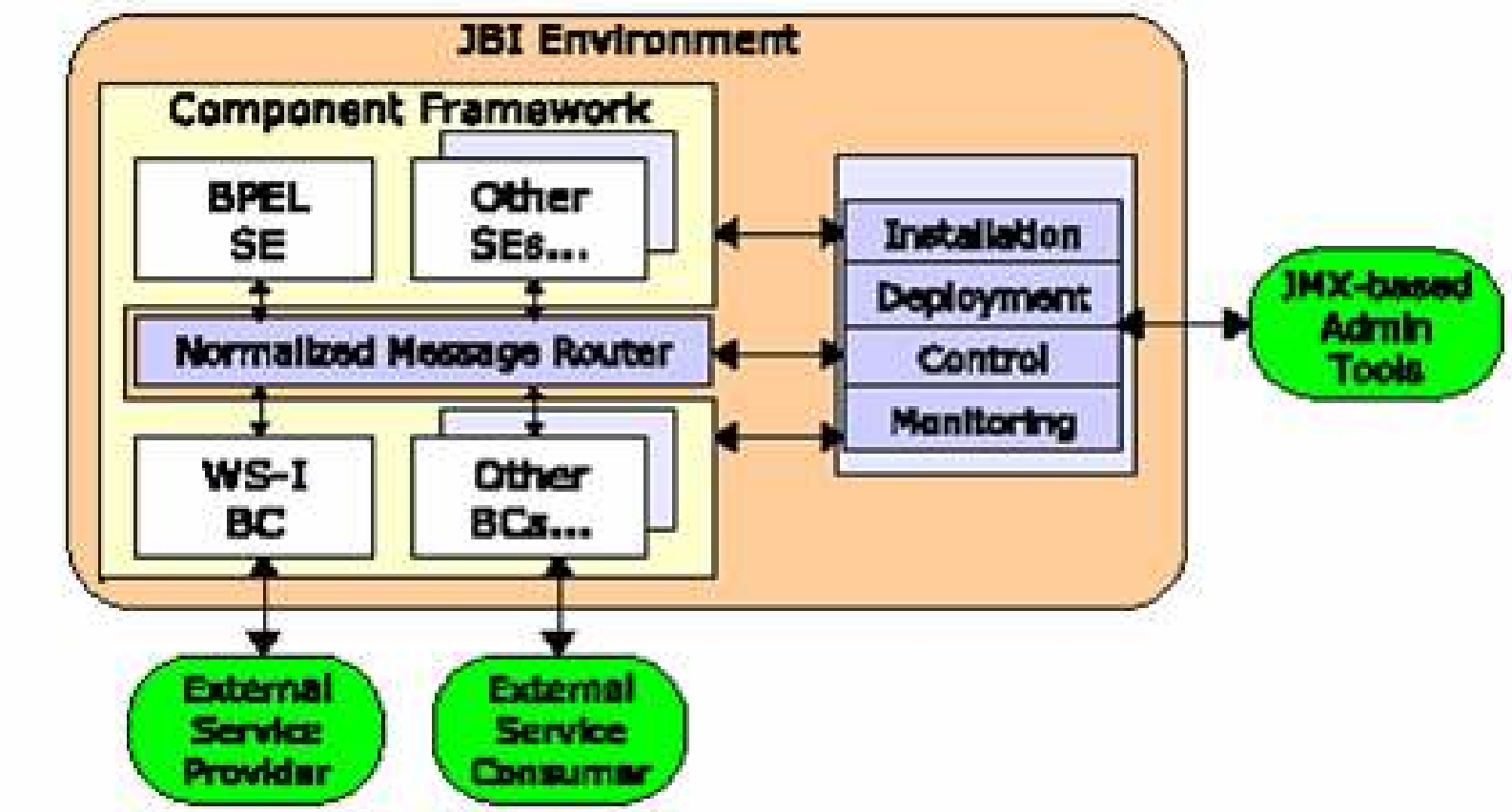
- **Bindings**
  - **CICS BC**
  - **CORBA BC**
  - **DCOM BC**
  - **File BC**
  - **FTP BC**
  - **HTTP BC**
  - **IMS**
  - **JMS BC**
  - **JDBC BC**
  - **MQ Series BC**
  - **SAP BC**
  - **SMTP BC**
  - **UDDI BC**
  - …



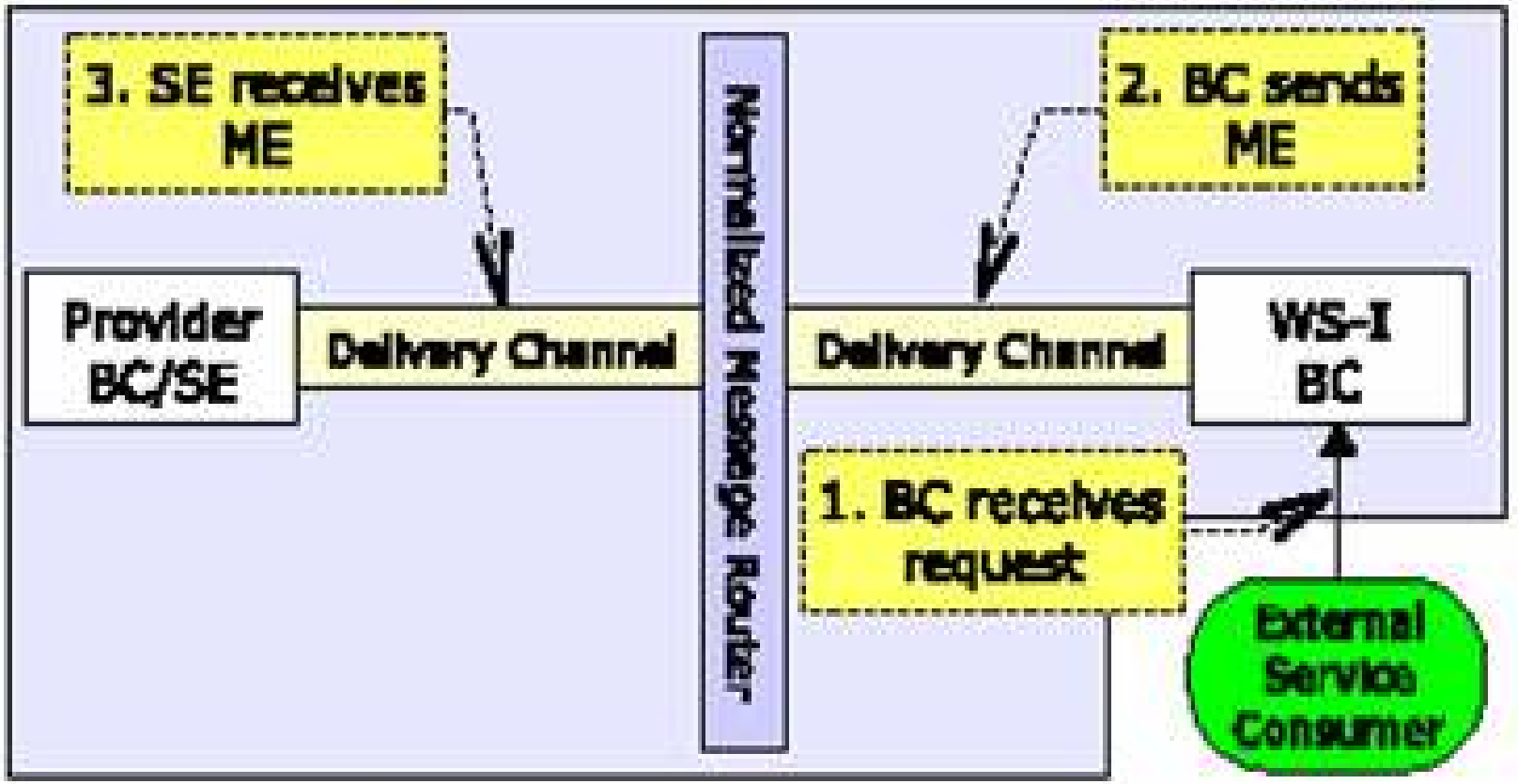
JBI Architecture

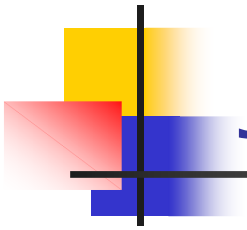
- **Messaging based, plug-in architecture**
- **Allows third-party components to be plugged into a standard infrastructure, and allows those components to interoperate**
- **It does not define the pluggable components themselves, but the container/framework interfaces, behavior and services**
- **Is itself a service-oriented architecture**
- **Components describe their capabilities via WSDL**
- **Key components**
  - **Service Engines (SE) - pluggable business logic**
  - **Binding Components (BC) - pluggable external connectivity**
  - **Normalized message router (NMR)**

JBI Architecture



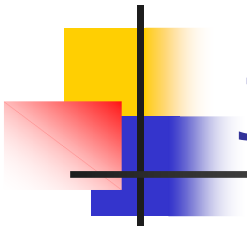
JBI Message Exchange Example





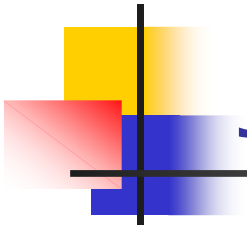
JBI Normalized Messages

- Standard format forms the foundation of the interoperability between JBI components
- The primary content of a normalized message is always XML
- There are two distinct parts to a normalized message
  - Content - the message data, as described by the abstract WSDL (represented as an XML transform Source, e.g. DOM or SAX Source)
  - Context (Meta-data in message properties, such as a security subject or protocol headers)



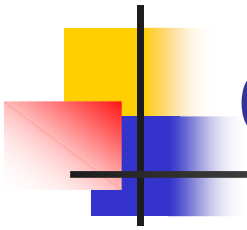
JBI - Service Units and Service Assemblies

- **Binding Components and Service Engines can act as containers**
- **Service Units (SU) can be deployed to installed BCs and SEs**
  - **This allows the deployment (and undeployment) of component-specific artifacts (e.g. concrete WSDLs)**
  - **Can describe what services are provided and consumed by the component**
  - **Besides the standard descriptor, the component is responsible for interpreting the contents of the SU jar**
- **A Service Assembly (SA) can package multiple service units and defines the target components to deploy them to**



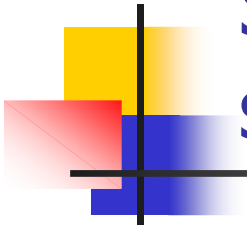
JBI – Integration Example

- BPEL process can read a file containing a credit application
- Pass the credit information to a credit-rating web service via SOAP and get a response (credit score) via SOAP
- Use a business rules engine to process the result (loan interest rate based on credit score)
- Communicate the decision to the applicant using SMTP (email notification)



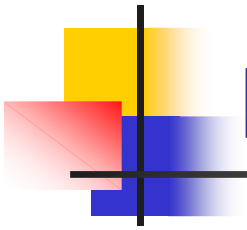
Open ESB and GlassFish

- Open ESB runs in GlassFish/Sun Java Application Server 9 and JBoss Application Server (experimental)
- Open-ESB is a container within a container – a hosted application in GlassFish
- Use NetBeans IDE 5.5 to develop service engines and binding components



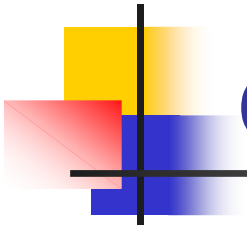
Sun SOA development & deployment suite using JBI

- Implement enterprise-class SOA and next generation web applications
  - Sun Java System Application Server 9
  - JBI Runtime with BPEL
  - Java EE 5
  - Portlet Container
  - NetBeans IDE 5.5 with NetBeans Enterprise Pack, Web Pack



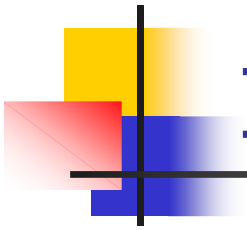
NetBeans 5.5 IDE

- NetBeans IDE with Enterprise Service Pack includes:
  - Sun Java Application Server 9
  - JBI Service Engines (JavaEE SE, ETL SE, …)
  - JBI Binding Components (FTP, JDBC, JMS, MQ Series, SMTP)



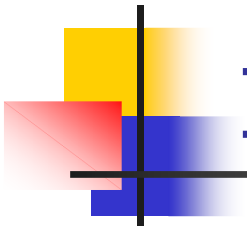
Open Source Tools/Technologies

- IDEs – NetBeans 5.5, Sun Studio Creator 2
- Web Services using Java/JEE, JSF, XML, WSDL, UDDI, SOAP, AJAX
- May use JCA, EDI, FTP for composite services
- May use JMS for transmitting encapsulated messages
- Sun Java Application Server 9, Sun Portal Server
- JBoss Application Server, JBoss Portal Server
- MySQL database



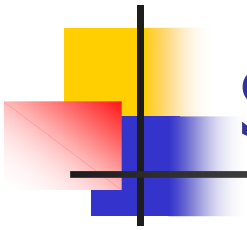
Implementation Approach - I

- Implement some or all components of AcademicEdge SIS as services
- Create services from other existing systems
- Use BPEL to create & execute business processes
- Create Portlets for student, staff, faculty
- Use WSRP service to embed Portlets in Portal (uPortal)



Implementation Approach - II

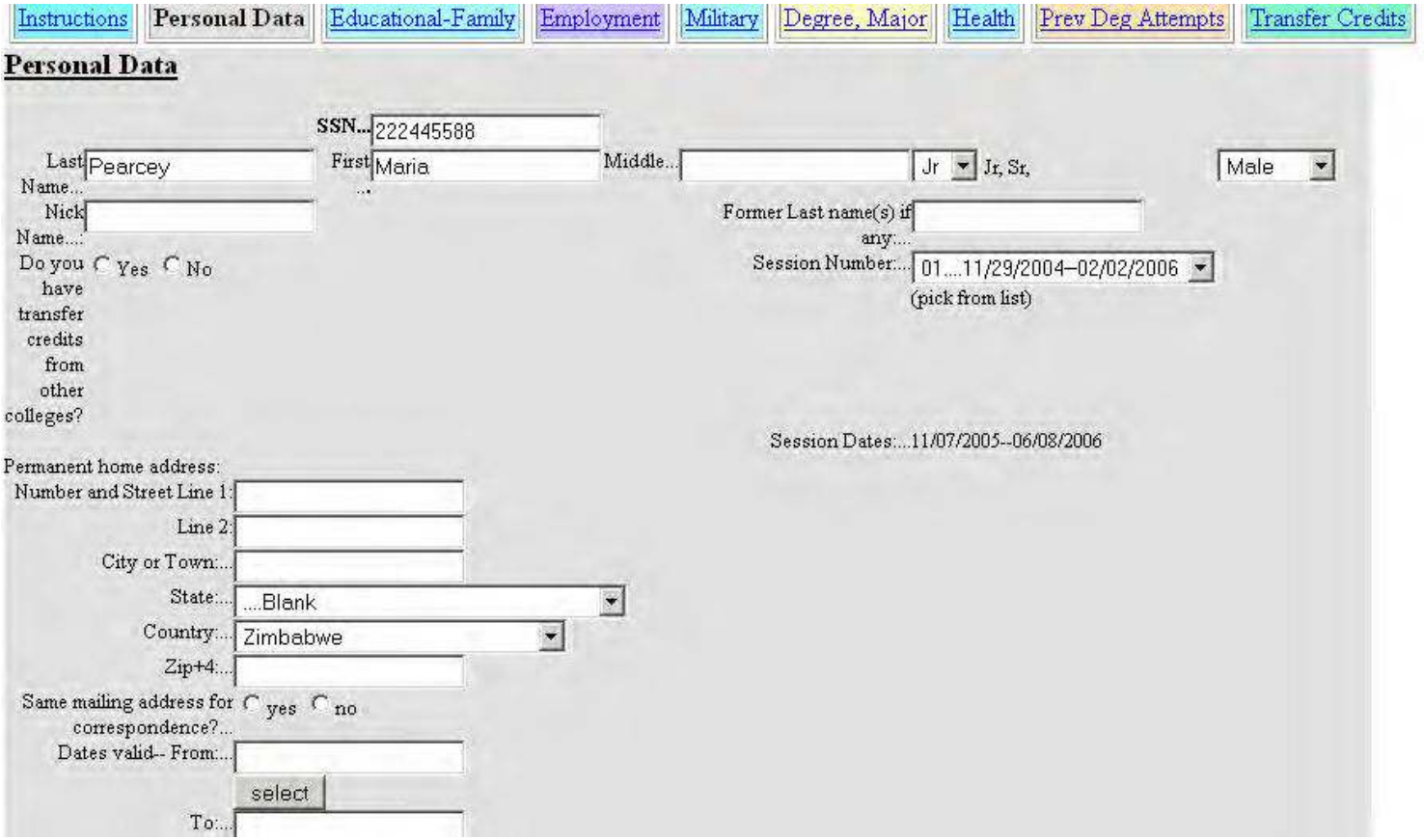
- Create services from existing systems
- Use BPEL to create business processes
- Create Portlets for student, staff, faculty
- Use WSRP service to embed Portlets in Portal (uPortal)
- Replace legacy system modules by AcademicEdge system modules, as required



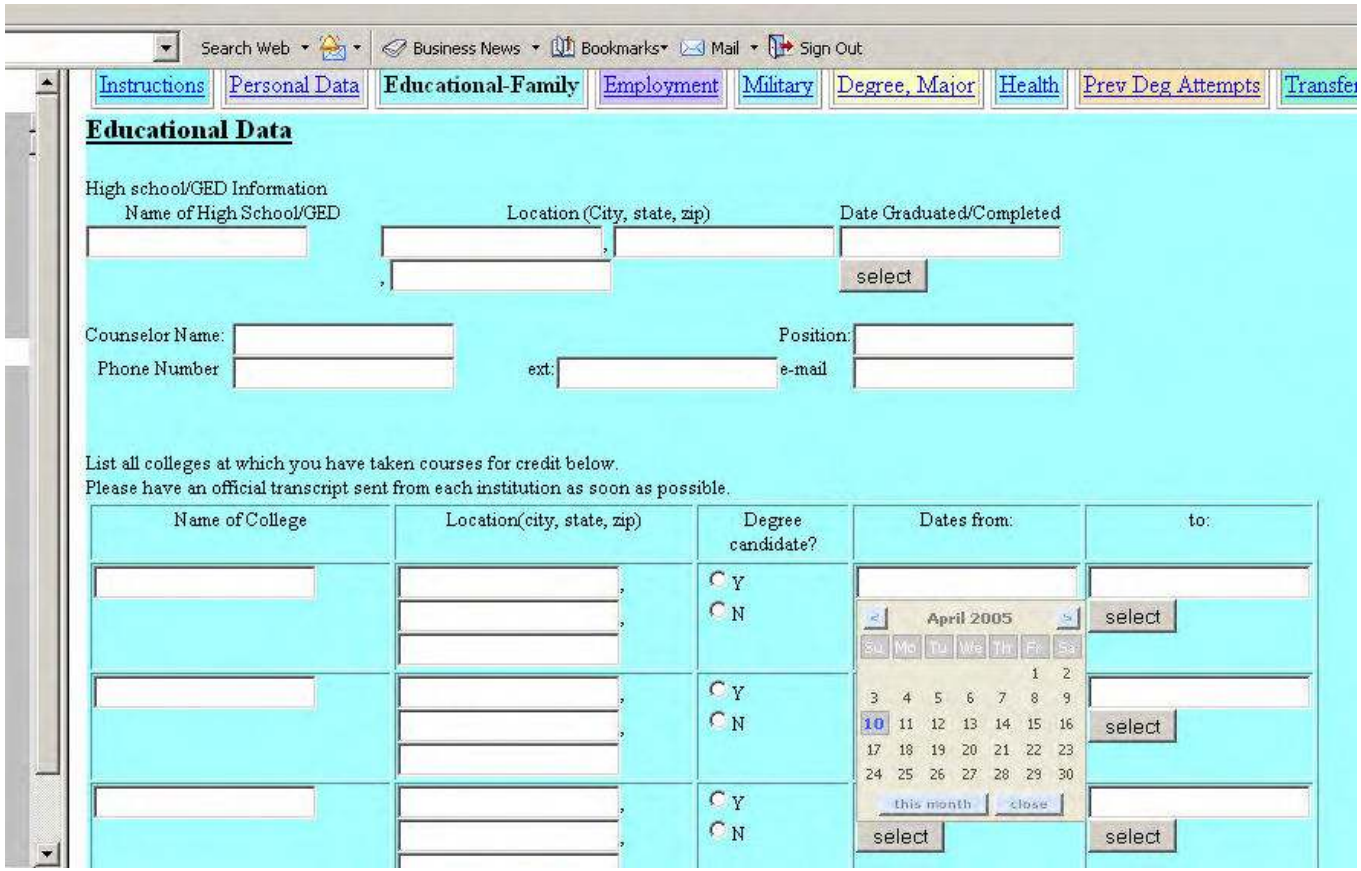
Sample Screen Shots

- The following screen shots are example of current templates in the system. All existing templates will be used as a basis for customizing your application.

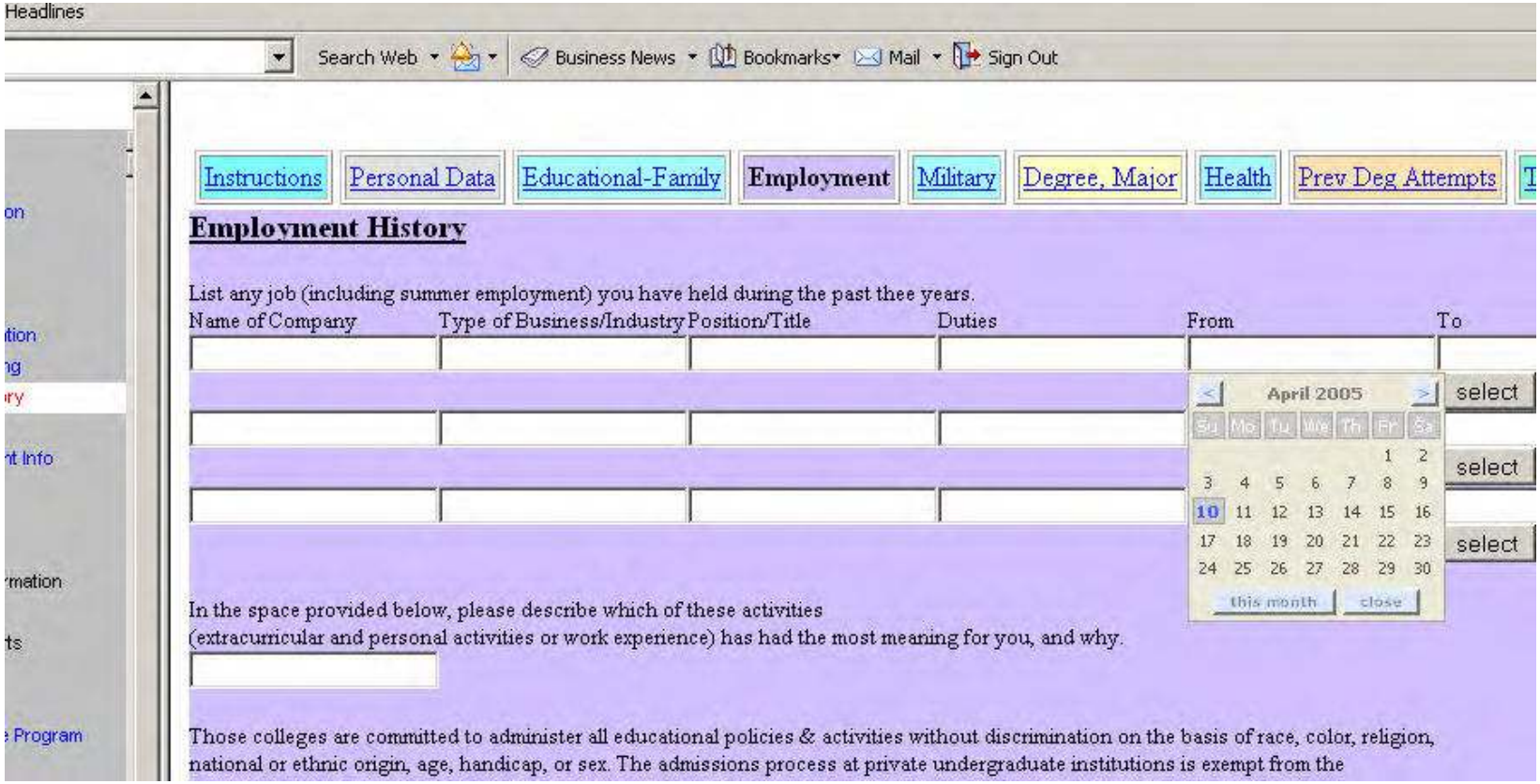
Application – Personal data



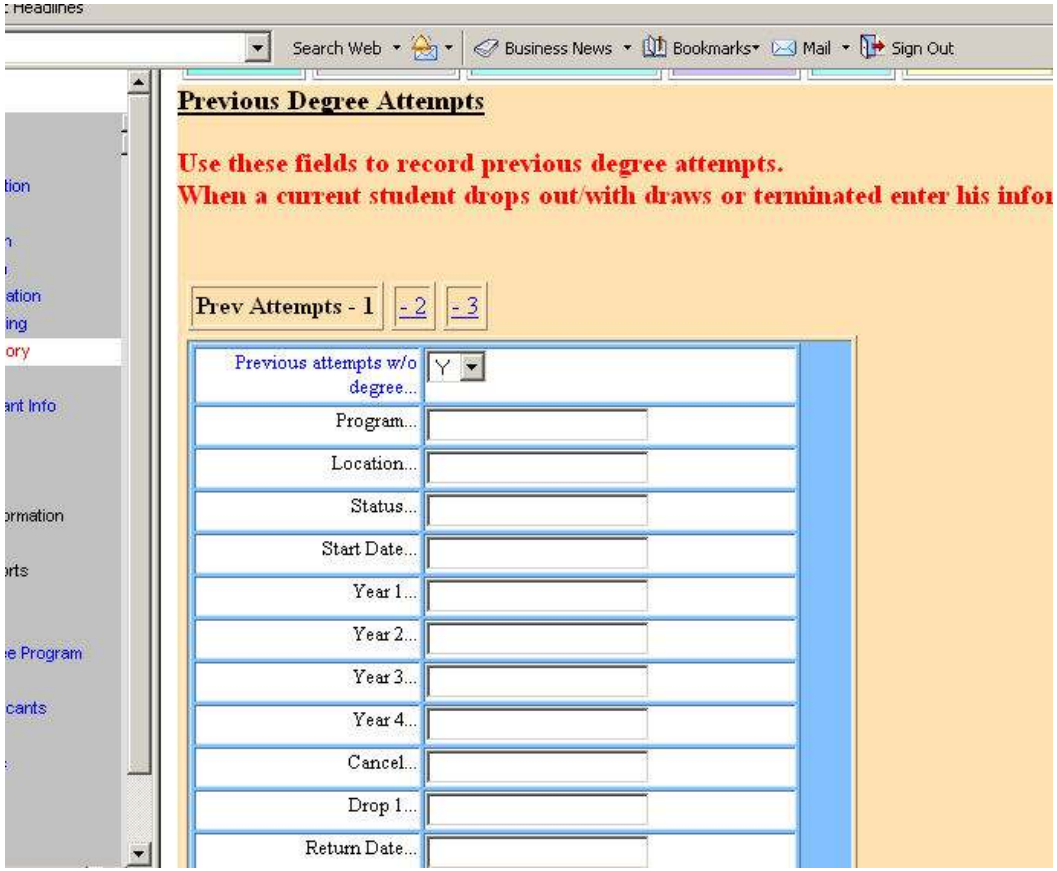
Application – Educational data



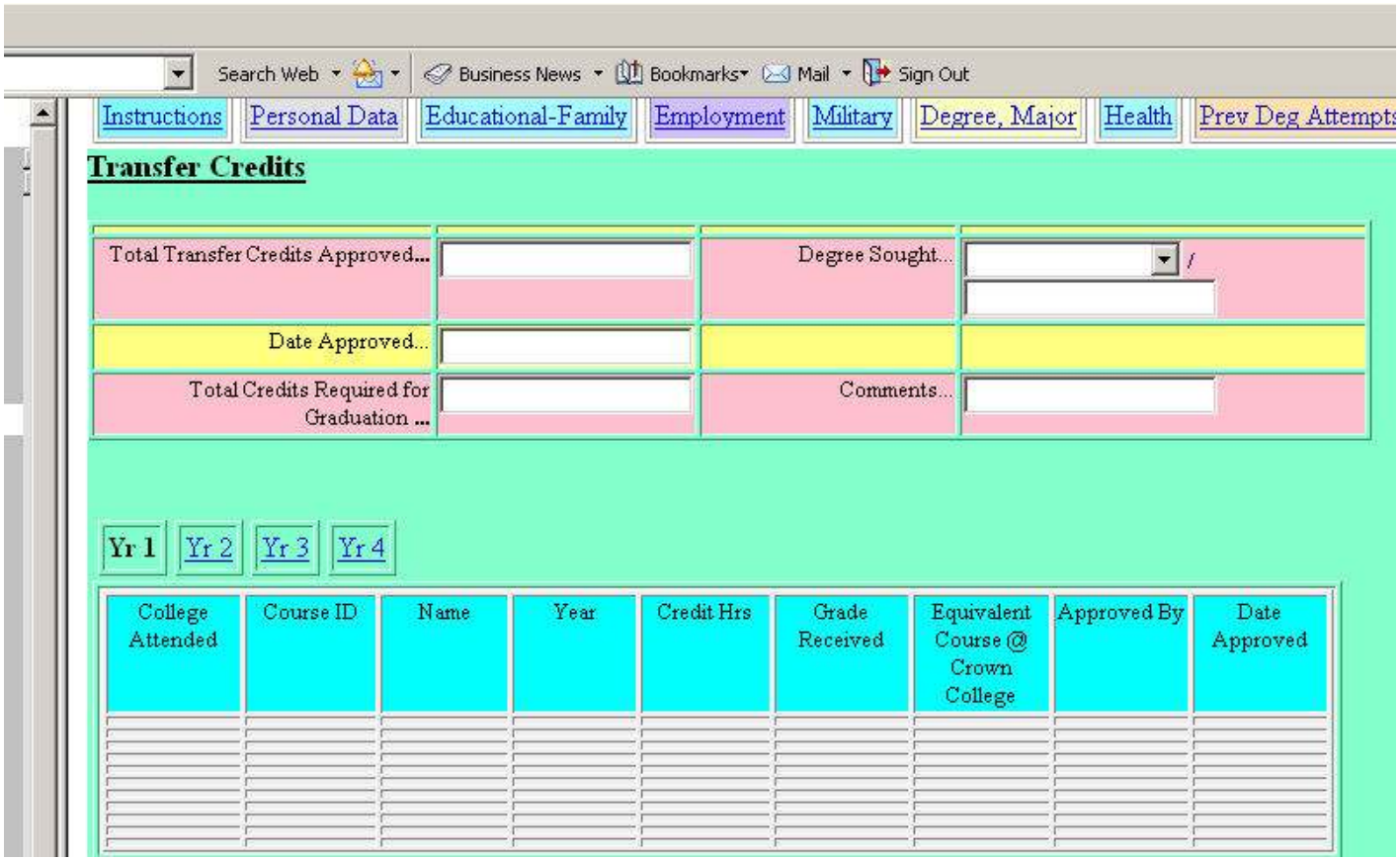
Application – Employment history



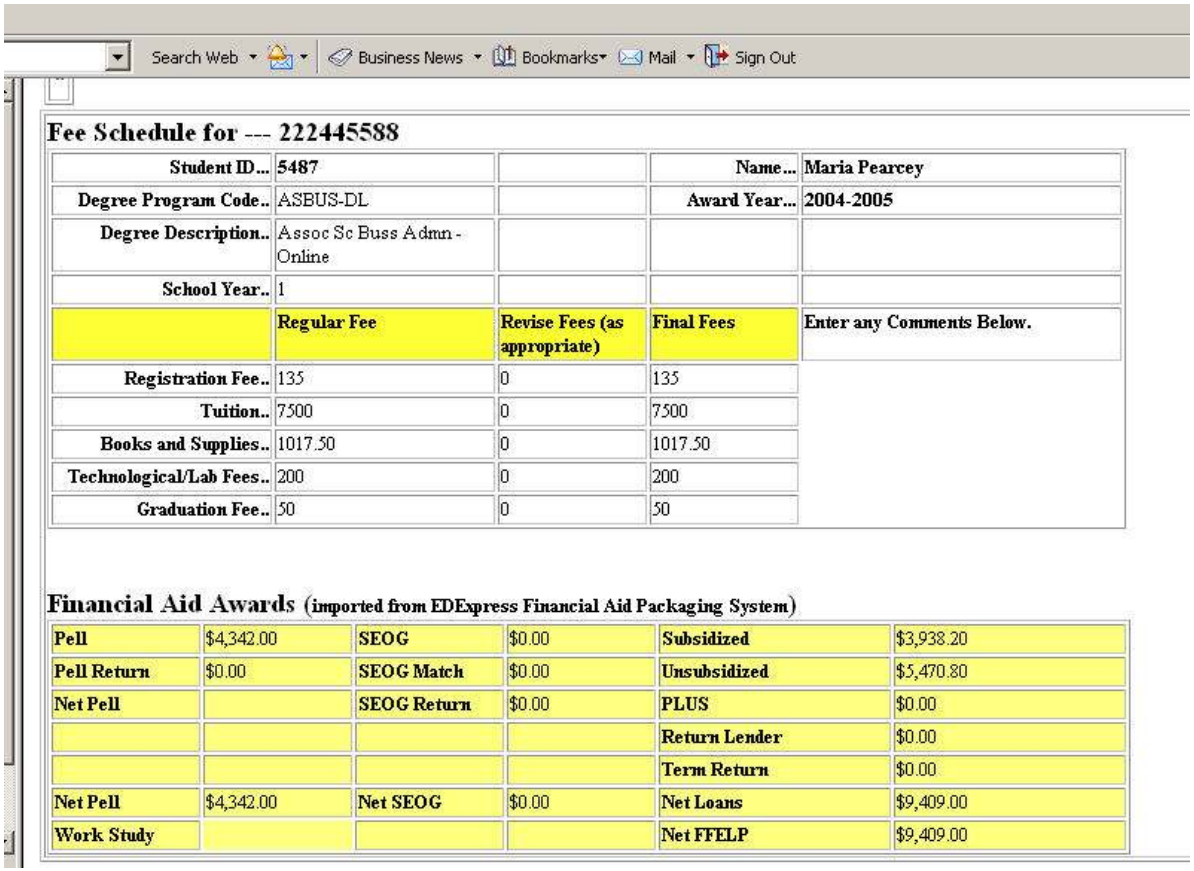
Application – Prior degree attempts



Application – Transfer credit



Financial Aid information



Search Web · Business News · Bookmarks · Mail · Sign Out

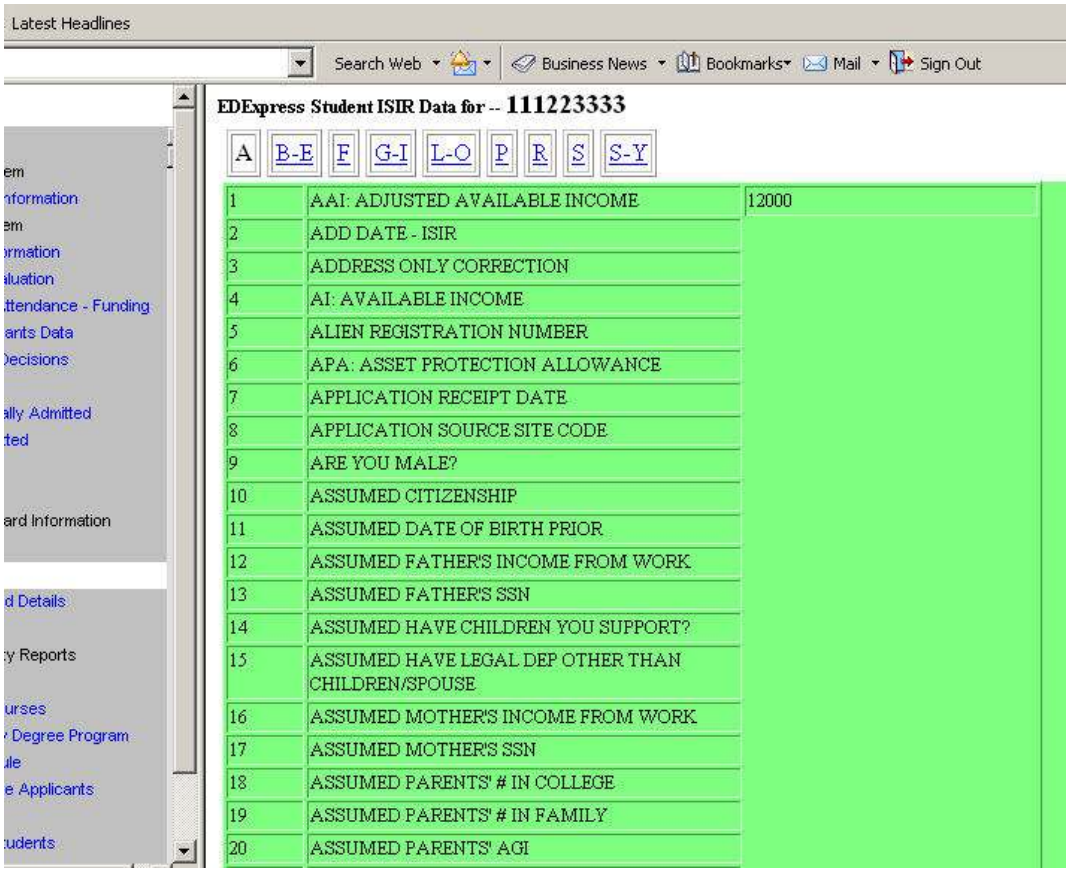
Fee Schedule for --- 222445588

Student ID...	**5487**		**Name...**	**Maria Pearcey**
Degree Program Code..	ASBUS-DL		**Award Year...**	**2004-2005**
Degree Description..	Assoc Sc Buss Admn - Online			
School Year..	1			
	Regular Fee	**Revise Fees (as appropriate)**	**Final Fees**	**Enter any Comments Below.**
Registration Fee..	135	0	135	
Tuition..	7500	0	7500	
Books and Supplies..	1017.50	0	1017.50	
Technological/Lab Fees..	200	0	200	
Graduation Fee..	50	0	50	

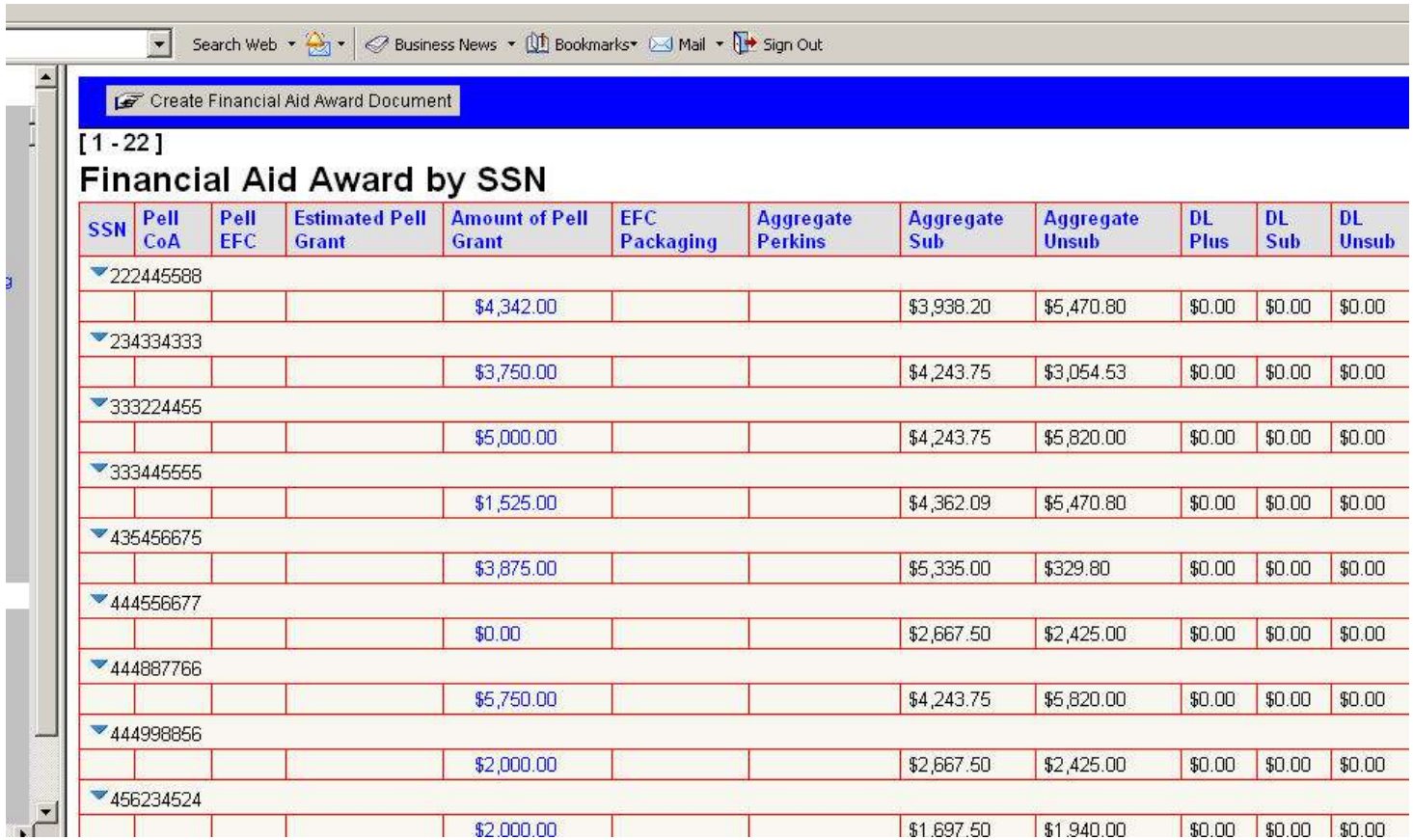
Financial Aid Awards (imported from EDExpress Financial Aid Packaging System)

Pell	$4,342.00	**SEOG**	$0.00	**Subsidized**	$3,938.20
Pell Return	$0.00	**SEOG Match**	$0.00	**Unsubsidized**	$5,470.80
Net Pell		**SEOG Return**	$0.00	**PLUS**	$0.00
				Return Lender	$0.00
				Term Return	$0.00
Net Pell	$4,342.00	**Net SEOG**	$0.00	**Net Loans**	$9,409.00
Work Study				**Net FFELP**	$9,409.00

Import Financial aid data from EDExpress



Financial aid award report



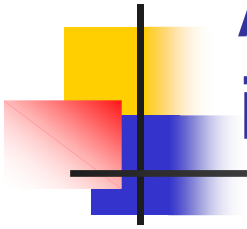
Search Web | Business News | Bookmarks | Mail | Sign Out

Create Financial Aid Award Document

[1 - 22]

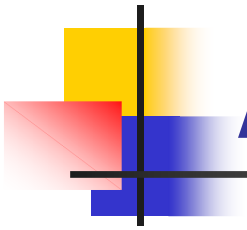
Financial Aid Award by SSN

SSN	Pell CoA	Pell EFC	Estimated Pell Grant	Amount of Pell Grant	EFC Packaging	Aggregate Perkins	Aggregate Sub	Aggregate Unsub	DL Plus	DL Sub	DL Unsub
222445588											
				$4,342.00			$3,938.20	$5,470.80	$0.00	$0.00	$0.00
234334333											
				$3,750.00			$4,243.75	$3,054.53	$0.00	$0.00	$0.00
333224455											
				$5,000.00			$4,243.75	$5,820.00	$0.00	$0.00	$0.00
333445555											
				$1,525.00			$4,362.09	$5,470.80	$0.00	$0.00	$0.00
435456675											
				$3,875.00			$5,335.00	$329.80	$0.00	$0.00	$0.00
444556677											
				$0.00			$2,667.50	$2,425.00	$0.00	$0.00	$0.00
444887766											
				$5,750.00			$4,243.75	$5,820.00	$0.00	$0.00	$0.00
444998856											
				$2,000.00			$2,667.50	$2,425.00	$0.00	$0.00	$0.00
456234524											
				$2,000.00			$1,697.50	$1,940.00	$0.00	$0.00	$0.00



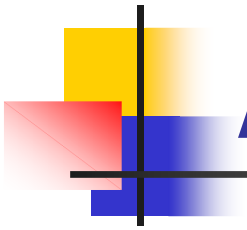
AcademicEdge SIS on open source infrastructure

- We can deploy AcademicEdge on available open source infrastructure. A possible deployment configuration:
  - Linux operating system
  - Sun Java Application Server 9, JBoss Application Server, JBoss Portal Server
  - MySQL database



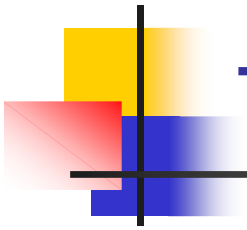
AcademicEdge Java/JEE SIS

- Deployed application will have same (or similar) user interface as shown in various images at this web site.
- The Forms can be opened in a browser, filled out, submitted and the results are displayed in the browser.
- Admission, Registration, financial aid, bursar, COD, Alumni functions are all performed over the web using a browser.



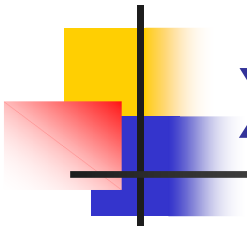
AcademicEdge Web Services

- Certain services of AcademicEdge can be exposed as Web Services so that other applications can access those services using appropriate methods.
- Lets us take a quick tour of Web Services and related technologies.



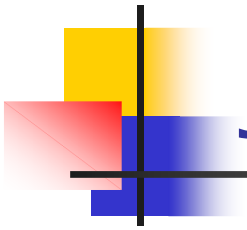
Typical Web Services Scenario

- An application sends a request to a service at a given URL using the SOAP protocol over HTTP.
- The service receives the request, processes it, and returns a response.



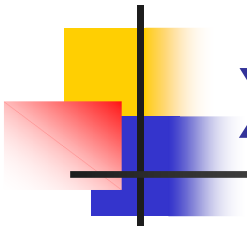
XML and Java in Web Services

- XML is used for data representation
- Java is used for processing logic.



JEE Platform for Web Services

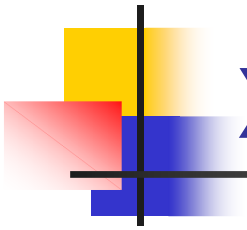
- For enterprises, Web Services need to be scalable, Secure and efficient.
- Java Enterprise Edition 5 (JEE 5) is especially designed to meet these requirements.



XML

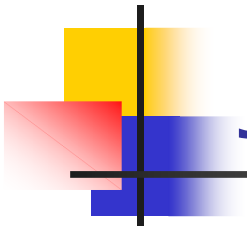
- XML is an industry-standard, system-independent way of representing data. XML encloses data in tags, that relate to the meaning of the enclosed text. An example:

```
<priceList>
  <coffee>
    <name>Mocha Java</name>
    <price>11.95</price>
  </coffee>
</priceList>
```



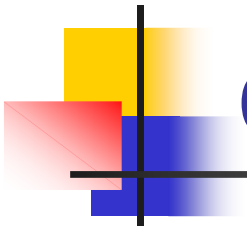
XML DTD/Schema

- Describes the structure of a XML document. For the previous example, XML dtd/schema specifies the location of coffee name, coffee price and their attributes.
- Any XML document that follows the constraints established in a dtd/schema is said to conform to that dtd/schema.



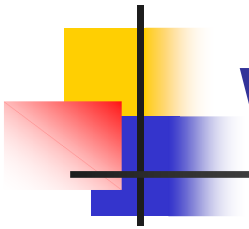
JAX - WS

- Java API for XML Web Services – JSR 224 complaint web services
- A typical WS uses SOAP (Simple Object Access Protocol) to request a service (e.g, student grade) and the result is returned via SOAP.



Creating Web Service using JAX-WS

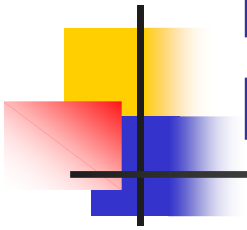
- The service requires
  - interface definition -- that declares service's remote procedures and
  - implementation class -- that implements those procedures.



Web Service Example

- A simple web service will look up a course description from a course code.

Example Web Service – Course Codes Look-up

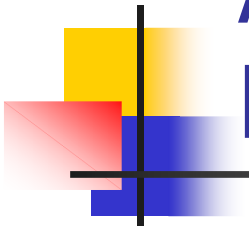


- **courseCodes Method invocation**
- **Method parameter(s)**
- **TypeValueMethod returned**
- **java.lang.String : "[SSC, CMP, CJS, PA, PAwI, MTH, PLS, PAwoI, NAS, BUS, HUM]"**

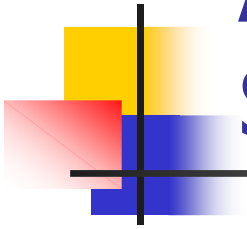
Example Web Service (Continues)

- **SOAP Request**
  - <?xml version="1.0" encoding="UTF-8"?>
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:ns1="http://services.ae.com/"> <soapenv:Body> <ns1:courseCodes/>
    </soapenv:Body>
    </soapenv:Envelope>
- **SOAP Response**
  - <?xml version="1.0" encoding="UTF-8"?>
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:ns1="http://services.ae.com/">
    <soapenv:Body> <ns1:courseCodesResponse> <return>[SSC, CMP, CJS, PA, PAwI, MTH, PLS, PAwoI, NAS, BUS, HUM]</return> </ns1:courseCodesResponse>
    </soapenv:Body>
    </soapenv:Envelope>

Accessing the Web Service via a browser

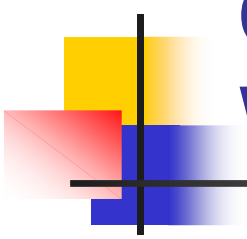


- A browser client application for a Web service has code that invokes the desired method to request a service and receives a response.



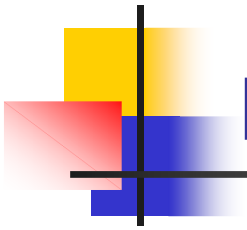
All services can be accessed as Web Services

- All required services at an University or College can be made available as Web Services and accessed by others via a browser.
  - Student services performed by AcademicEdge
  - HR/Finance/Accounting services, and
  - Services from other applications.



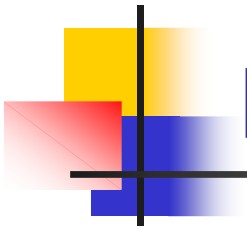
Creating critical Business Processes out of Web Services

- Web Services expose operations of various applications
- We integrate several Web Services to create a Business Process (BP)
- BPEL (Business Process Execution Language) is used for the definition & execution of BPs
- BEPL allows realization of SOA (Service oriented Architecture)



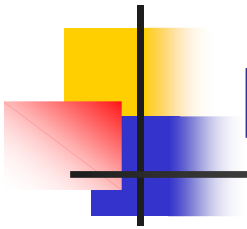
Business Process Execution Language

- BPEL uses XML-based technologies including SOAP, WSDL, UDDI, WS Messaging, WS Addressing, WS Coordination and WS Transactions
- BPEL standardizes application integration including integration of partner services



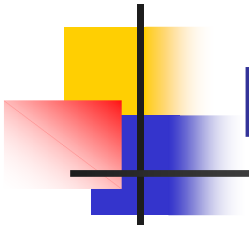
BP development using BEPL

- We use Sun Java Application Server 9 BPEL Process Manager run time for process deployment.



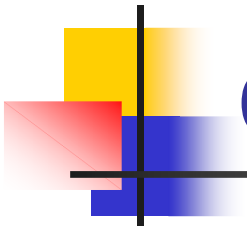
BEPL is the cornerstone of SOA

- Business processes are connected applications built using Web Services
- BEPL address enterprise integration requirements for creating BPs using open standards.



Key Concepts

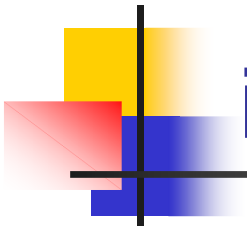
- BPEL can be used to create synchronous & asynchronous services using
  - **Web Services/WSDL as component models**
  - **XML as data model**
  - **Synchronous/asynchronous messaging patterns**
  - **Deterministic/Non-deterministic flow coordination**
  - **Hierarchical exception management**



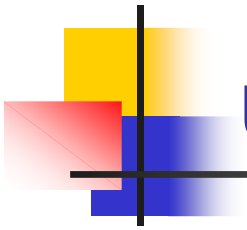
Creating BPs in AcademicEdge

- We use BPEL Designer (in NetBeans 5.5) to create a Business process by combining one or more Web Services
- Then compile & deploy the Business process to the Sun Java Application Server 9 for access via a client.

Web Portal is the gateway to all institutional services

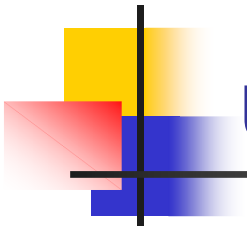


- Web portal is used as the self-service gateway to access all services provided by a college or an university.
- uPortal, open source, is designed to provide such services to higher education community.



uPortal Overview

- uPortal is a sharable portal under development by JA_SIG for institutions of higher-educations. (http://www.uportal.org)
- uPortal is an open-standard effort using Java, XML, JSP and J2EE.
- uPortal is a framework, consisting of a set of Java classes and XML/XSL documents that is used to produce a campus portal.
- Customization allows each user to define a unique and personal view of the campus Web.

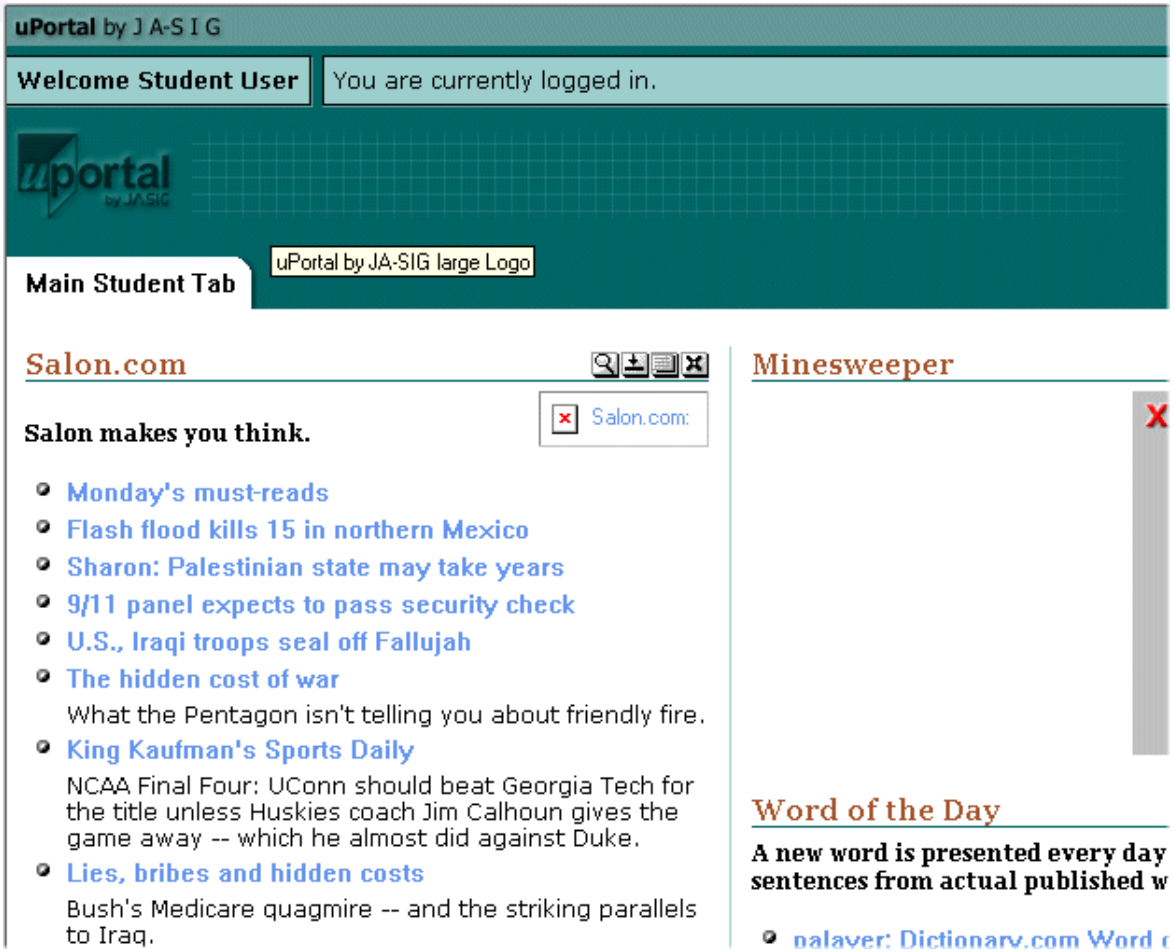


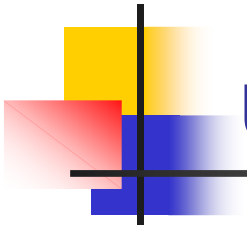
uPortal Deployment & Customization

Requires:

- Application Servers: JBoss, Sun, Bea WebLogic, IBM WebSphere, …
- Relational Database Server
- LDAP Server for authentication

uPortal Student Web Site

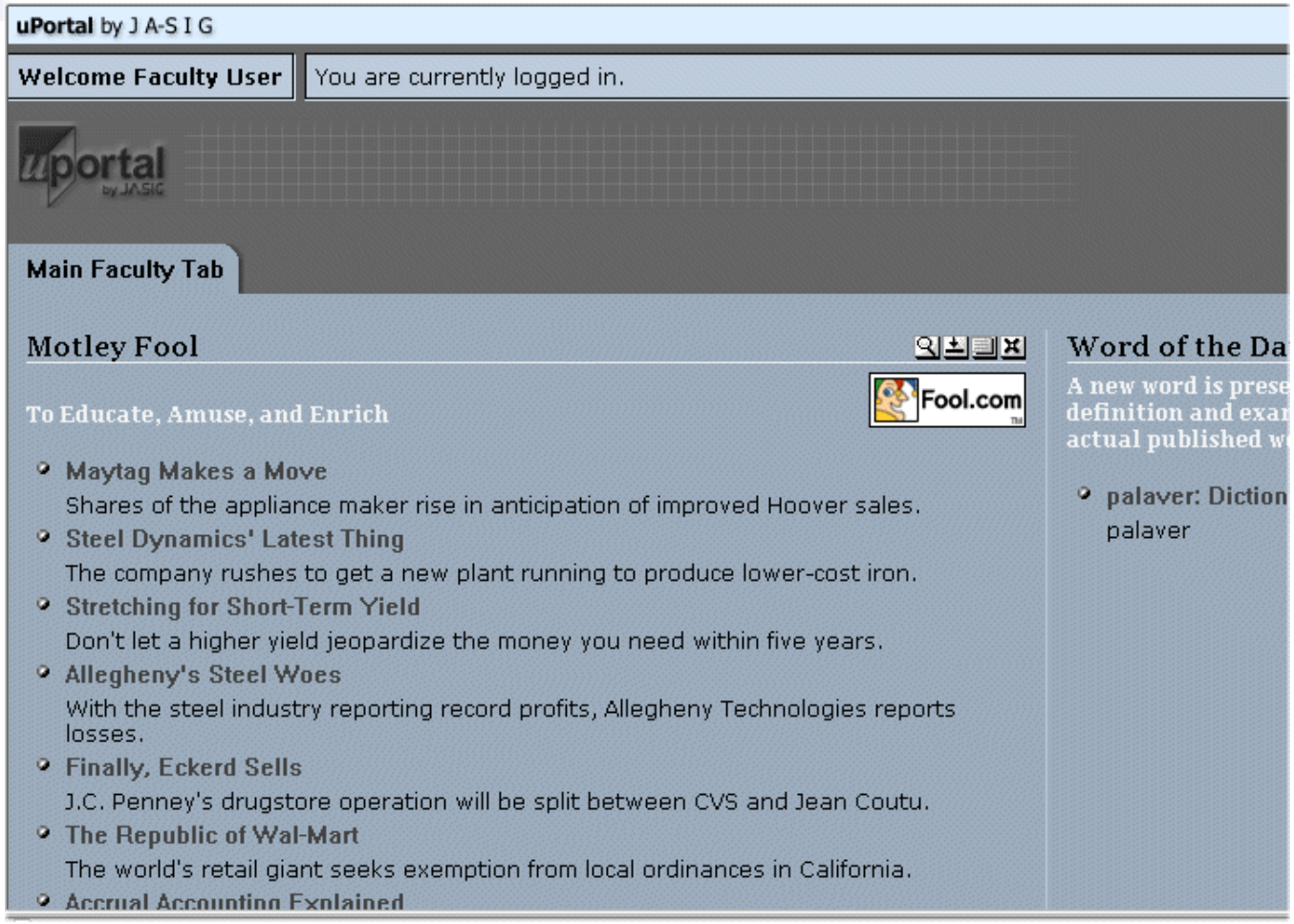


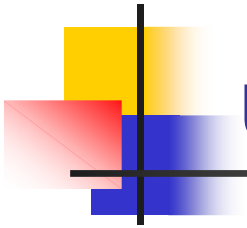


uPortal Student Web Site

- Student specific portlets can be included in these student portals

uPortal Faculty Web Site

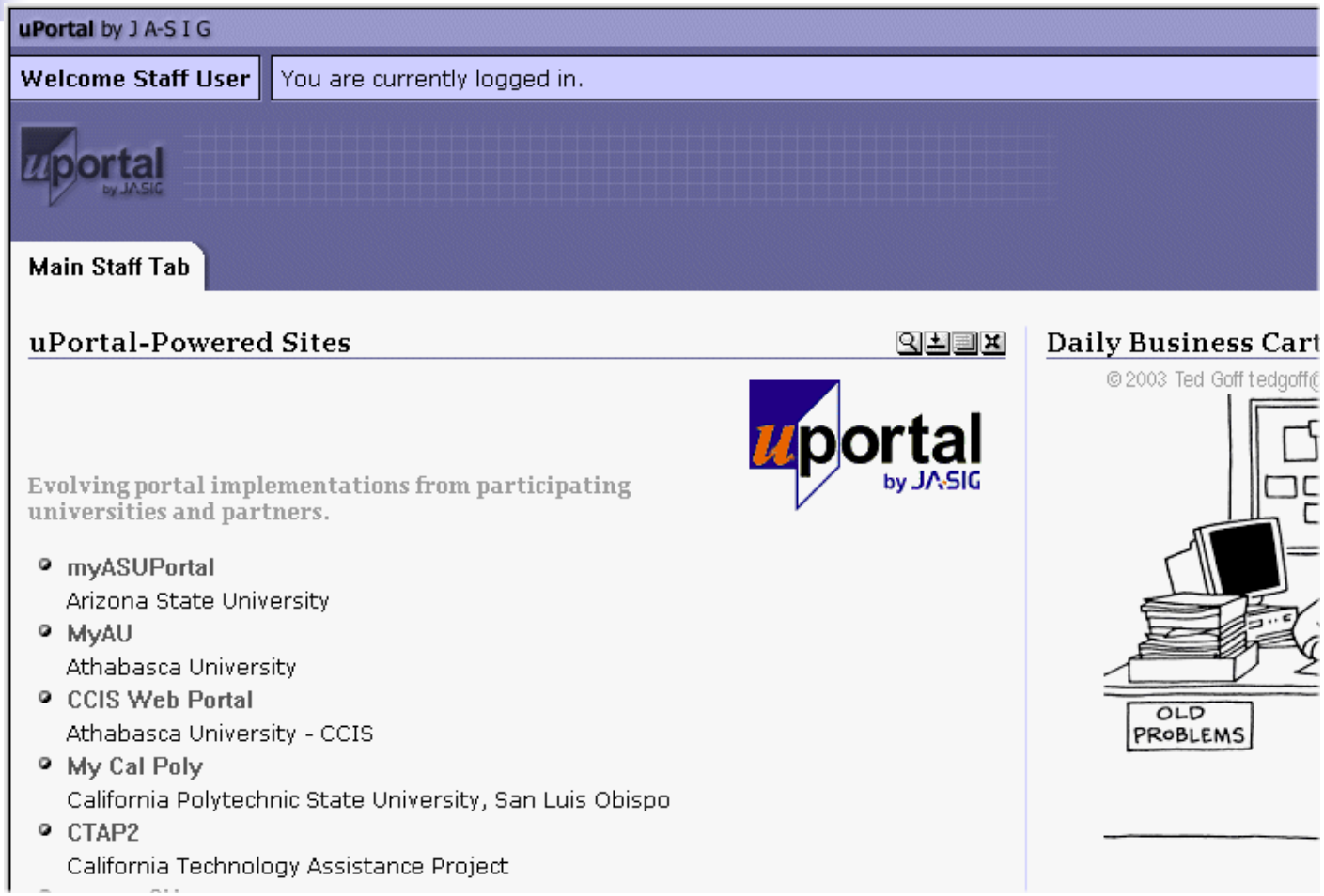


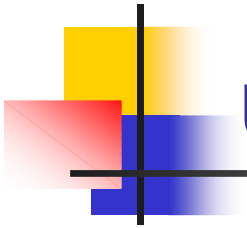


uPortal Faculty Web Site

- Faculty specific portlets can be included in these portals

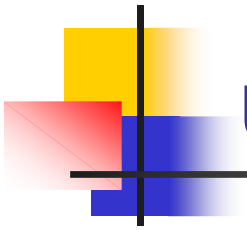
uPortal Staff Web Site





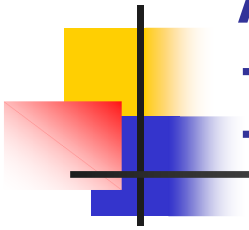
uPortal Staff Web Site

- Staff specific portlets can be included in these portals



uPortal Content Integration

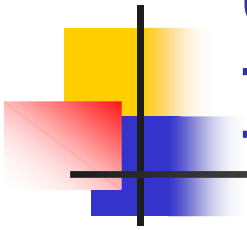
- Through uPortal Channels or Portlets
- Web application, which requires user interaction, would use a Channel or Portlet as its user interface to deliver content to the portal.
- Portlets will include related business processes



AcademicEdge Application Integration with uPortal

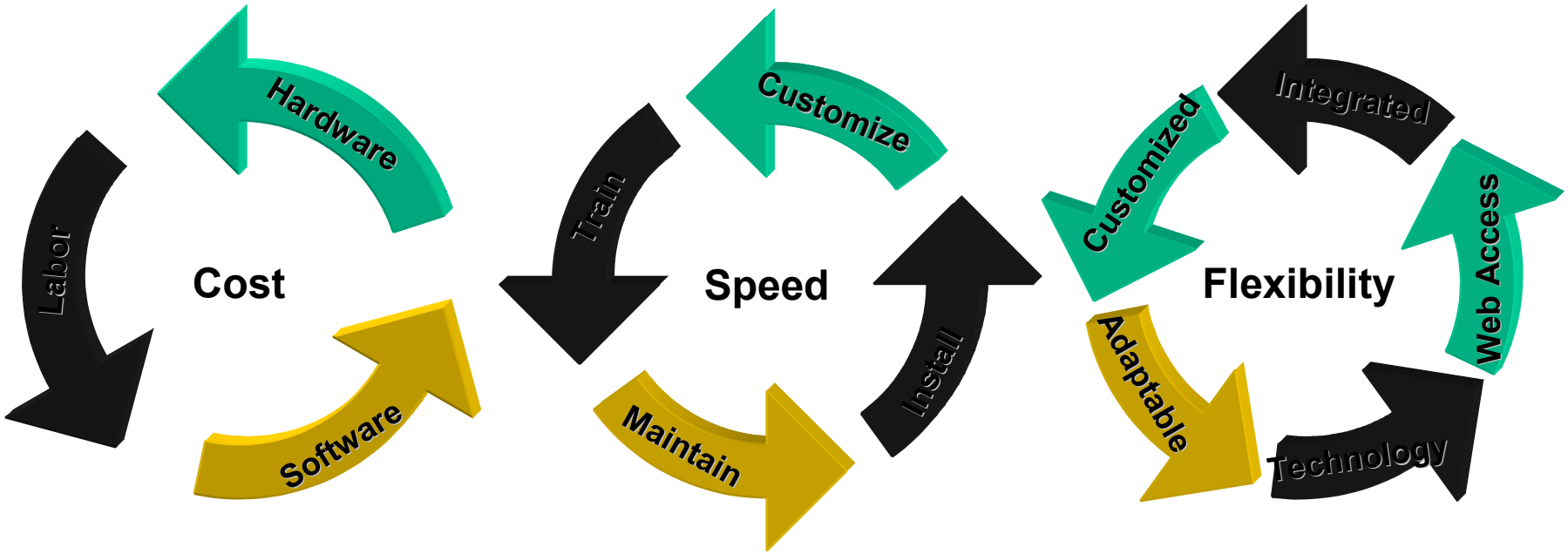
- AcademicEdge application services are JEE 5 applications & web services
- AcademicEdge JSR 168 compliant portlets can be integrated with uPortal Channels to provide a unique, personalized view to the users – students, faculty, staff, parents and alumni.

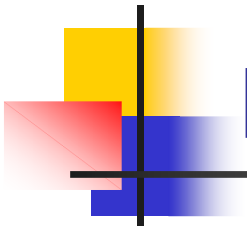
uPortal and other Application Integration



- Other campus applications will have to be web enabled to integrate with uPortal
- Application vendors can provide the necessary services and support for this effort.
- BPEL Process Manager can be used to create appropriate business processes from these applications

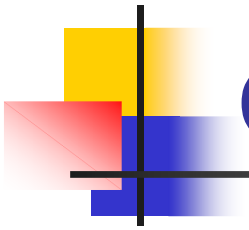
Our Strengths





Key Benefits

- Built on SOA framework
- Comprehensive system features
- Easy to customize, use and maintain.
- Seamless integration with the Web
- Short implementation cycle
- Cost effective solution with high payoff
- Enables group collaboration



Conclusion

- AcademicEdge is a comprehensive SIS
- Its Service Oriented Architecture allows integration with best-in-class software
- It is standards based, and web centric
- It is secure, affordable and scalable
- Contact – info@academicedgeonline.com