Business Entities:
An Artifact-Centric Approach for Business Process Management

Terry Heath - Senior Software Engineer
IBM T.J. Watson Research Center
December 1st-16th 2009  Università di Roma "La Sapienza"

Overview of Lectures

- **Lecture #1**
  - Artifact-Centric Approach
  - Comparison of Business Process Management Approaches
  - Brief introduction to Siena
  - Siena Demo

- **Lecture #2**
  - Review Siena Meta model
  - Review Siena Hotel Example (Deep Dive)
  - Install Siena

- **Lecture #3**
  - External Service Integration (REST and WSDL)
  - More Siena Examples (Patrizia and Alessio)
  - How to use Siena
  - Possible Homework Assignment

- **Lecture #4**
  - Review homework
  - Artifact Design Patterns
  - Open issues for Siena system
  - Future Artifact Designs
Lecture #1

- **Artifact-Centric Approach**
  - Introduction to Business Entities

- **Comparison of Business Process Management Approaches**
  - *Process-Centric* approach using Hotel Scenario
  - *Artifact-Centric* approach using Hotel Scenario

- **Brief introduction to Siena**
  - Overview
  - Architecture

- **Siena Demo**
  - Review Hotel example

---

**A Key Challenge in Business Process Management**

*Many Stakeholders in an Enterprise*

- **Operations need to be**
  - **Faithful**
  - **Measurable**
  - **Flexible**

- **Speak in terms of**
  - “Functional Decomposition”
  - “Business Components”

- **“Impedance Mismatch”!!**

- **Speak in terms of**
  - “Workflow”
  - “Activity-flow”
  - “DB schema”

---

**Bridge the Gap between Business and IT**

- **Executive**
- **Business Goals**
- **Business Architecture**
- **Business Optimization**
- **IT Architect**
- **Systems Integrator**
- **Customers**
- **Employees**
- **Partners**
- **Resources**
Basic Challenge: Today’s approach to BPM environments is fundamentally disjointed (Many disjoint Models)

If Guest is paying by AMEX
Then give 5% discount

Core BP model is based on activity flows

The data being manipulated is often an afterthought, not related to other conceptual models

Lack of coherence adds substantial complexity to an already very complex environment

“Business Entities”: data + process combined to form a new, “holistic” foundation for BPM

- **Business Entities are Unifying Business Constructs**
  - Provides a skeleton that cuts across the Business
    - e.g., **Guest Stay**
      - From CheckIn to CheckOut
      - Blending of Data, Rules, Process, Measurements in the context of a Guest Stay

- **Includes specification of both**
  - The *information model*, to hold relevant data about an artifact as it moves through the workflow, and
  - The *possible lifecycles* it might follow

- **Insight**: Gives business managers a unified, end-to-end view of their business operations
- **Communication**: Numerous stakeholders have a common basis for understanding
- **Actionable**: Natural mapping to organization & IT levels
Brief comparison of BPM approaches

**Process-Centric Approach**

- **Business Data is**
  - NOT the primary focus
  - Business data is merely an after thought

- **Process Steps are the main concern**
  - *What* do humans do in the business
  - *What* systems need to be integrated

**Artifact-Centric Approach**

- **Business Data is**
  - The PRIMARY focus

- **Process steps occur in context**
  - The “Business Entity” needs *which* humans to do something to it.
  - The “Business Entity” needs to integrate with *what* certain systems.

Review of Hotel Scenario for comparison

- **Posting charges to the guest folio during a hotel stay**

- **Base scenario**
  - Guest checks in
  - Room charges are posted by the Night Audit process
  - Guest dines in the hotel restaurant
  - Guest checks out

- **Scenario evolution 1 – Handling of “lost” charges**
  - Guest has breakfast after checking out

- **Scenario evolution 2 – Handling of charges by “drop-ins”**
  - A non-guest dines at the hotel restaurant
Process-Centric approach for Hotel scenario

Check in- Checkout Process
(Process-Centric Approach)

- Waiting for Checkout Request

- Long Running Flow:
  - Remains active until Guest checks out
  - The Guest Stay information is lost in the process instance data
Night Audit Process – Modeled as separate process
(Process-Centric Approach)

- Multiple instances of the process created for each day for each guest
- No direct link between check in process and night audit process.

Combined check in- Checkout and Night Audit Process
(Process-Centric Approach)

Night audit process modeled as part of main flow to avoid multiple process instances being created.
Point of Sale (POS) Process  
(Progress-Centric Approach)

POS modeled as separate process as this can be instantiated independently any number of times.

Summary  
(Progress-Centric Approach)

- Discrete/Disjoint Processes
- Data is an after thought
- Guest stay information lost in long running process instances
- Lots of additional coding needed to integrate to Databases and Services
Entity-Centric approach of Hotel Scenario

- **Identify key Business Entities**
  - **Guest Stay**
    - **States**: Started, CheckedIn, Room Assigned, CheckedOut
    - **Information**: Stay_ID, CheckInDate, CheckOutDate, Guest Name, Guest Profile, Guest Type, Room Rate, Room Preferences, Room Number Assigned, Folio Info
  - **Guest Folio**
    - **States**: Started, Active, Settled
    - **Information**: Folio_ID, Guest Name, Room Number Assigned
  - **Charge**
    - **States**: Start, Pended, Paid, Posted, Lost, Cancelled
    - **Information**: Charge_ID, Date Incurred, Charge Type, Room Number, Payment Type, ItemInfo(code, desc, qty, cost), TaxInfo(Tax Rate, Desc, TaxTotal)
  - **Night Audit**
    - **States**: Started, InProgress, Completed
    - **Information**: Audit_ID, Stay_ID, Folio_ID, GuestName, Room Number, DataAuditStarted, DateAuditCompleted, reconciledCharges(1..n)

---

**Guest Stay** Entity
(Artifact-Centric approach)
**Folio Entity**  
(Artifact-Centric approach)

- **Guest Folio**
  - Started
  - Active
  - Settled

- **Default info**
- **Guest Info**
- **Room Info**
- **Charge Info**

**Information Model**

**Charge Entity**  
(Artifact-Centric approach)

- **Charge**
  - Started
  - Pended
  - Paid
  - Canceled
  - Lost

- **Default info**
- **Guest Info**
- **Charge Info**
- **Tax Info**

**Information Model**
**Night Audit Entity**
(Artifact-Centric approach)

- **Lifecycle**
  - Started
  - Room Charges Created
  - Charges Reconciled
  - Completed

- **Information Model**
  - Folio info
  - Guest Info
  - Reconciled Charge Info

---

**Business Entity Lifecycles and Business Entity Interactions**

- **Guest Stay**
  - Started
  - Checked In
  - Checked Out

- **Night Audit**
  - Started
  - Room Charges Created
  - Charges Reconciled
  - Completed
  - Begin night audit

- **Guest Folio**
  - Started
  - Active
  - Settled
  - Add Charges
  - Get Pending Charges For Guest
  - Post Charge

- **Charge**
  - Started
  - Pended
  - Paid
  - Lost
  - Canceled

- **Create POS Charge**
- **Settle Folio**
- **Create Room Charges**

© 2009 IBM Corporation
Some comparison points

- **Process-Centric Approach**
  - Process flows act as controllers
  - Data is an after thought
  - Some Operational data hidden in long running process data
  - Humans Work on "blocked" tasks in long running flows
  - Performance of long running flows not desirable

- **Artifact-Centric Approach**
  - Entities act as controllers
  - Data is Core
    - Business Entities accessible in DB
  - All Operation Data store in Business Entities
    - Queryable, Trackable, Measurable
  - Humans work on Business Entities that are ready
  - Performance of Entities (info, lifecycle, micro flows) considered acceptable

Business-Entities provide improved communication among stakeholders in the business

- **“Along” the artifact:**
  - People at “PENDED” can discuss meaningfully with people at “POSTED”
  - Can discuss attribute values produced, needed by different tasks

- **“Across variations”:**
  - Different regions can communicate using shared abstract model
    - (Variation of Rules in lifecycles and Process Steps)

- **“Up/down management chain”**
  - Artifact approach lends itself to more abstract / more detailed specifications
Business Entities give context for Service Invocations
External Service Integration (REST and WSDL)

Large Collection of Externals Services Contextualized by Business Entities

Business Rules Constrain Access, Lifecycle, and Behavior

- Rules define how lifecycles can be traversed
- Rules can also define how flows, and data can be manipulated
Introduction to Siena
*(Light-weight Artifact-Centric Modeling and Execution tool)*

- **Empower SME’s to easily Innovate new processes**
  - Tools and Runtime often too heavy and hard to understand
  - Innovators currently dependent on IT teams

- **Radical Simplification of Tools and Runtime**
  - **Siena Core Meta Model**
    - Describes semantics for all modeling constructs of Business Entities
  - **Model Management Engine**
    - *Web UI Modeling* tool for producing Business Entity models
  - **Model Execution Engine**
    - Default Execution UI for deploying and executing Business-Entity models
    - Directly executes Business Entity models
    - No coding necessary, No code generation
    - Small footprint can be easily hosted anywhere including laptop

---

**Innovator’s toolkit for Business Process Modeling**
*(Democratization of Innovation — Eric Von Hippel)*

---

**Manufacturer-centered innovation**

**Innovator-centered innovation**
Radical Simplification of Tools and Runtime
(Supporting Business Process Management Applications using Entity Centric Modeling)

Dreaming of being lighter and more Agile

50 Megas Download/Disk Space, 5-10 minutes, < 1 Gig Memory

Reduced set of BPM abstractions to define and create BPM solutions.

Tooling Stack

- WBI Modeler
- Rational Software Modeler
- WSAD-IE / WID
- Websphere Process Server

10 Gigs Download/Disk Space, 1-2 Days successful installation, At least 2 Gigs Memory

Siena Architecture Diagram

Artifact Model Editor
- SmartGWT

Artifact Model Import
- ZIP File

Custom UI (application specific)
- Rich Client

Default UI (dynamically generated)
- Google Web Toolkit

UI Toolkit (Artifact-centric widgets & helpers)

Runtime Container
- Artifact Model Management
- Data Access
- Behavior Services (lifecycle, flow, GSM engines)
- Access Control
- Rule Evaluation (JEXL, OCL)
- Monitoring & Analytics
- Extension Points

External Services
- REST Services
- Web Services

Artifact Data
- Apache Derby
- IBM DB2

Artifact Models
- XML

REST & Web Services
- Apache Tomcat
- WAS
- Geronimo

REST Services
- WSDL
Review: What is a Business Entity? (Deeper Inspection)

- It's a Unifying Business Construct:
  - **Structured by**
    - Core business data models
    - Artifact lifecycles
      - State machines (Siena)
      - Declarative stages (Project ArtiFact™)
  - **Providing Services**
    - Data services
    - Flow services
  - **Protected by Access Control**
    - Users and Roles
    - Entitlements
      - Data access rights
      - Service access rights
  - **Constrained By Business Rules:**
    - Data, services, lifecycles, flows, behavior
  - **Has measurable features**
    - Data, lifecycles, flows, tasks

Siena Entity (The Core of Siena)
Siena Demo: Example of an Entity-Centric Solution

- Review Hotel Design
- Run Hotel Design

Demo Wrap Up  Questions and Answers
Lecture #2

- **Summary of Lecture #1**

- **Review Siena Hotel Example (Deep Dive)**
  - Review the Siena Modeling Tool
  - Review the Model Execution Engine

- **Install Siena**
  - Hello World (Follow Instructor)

- **Possible Homework Assignment** Due on Lecture #4 (Review)
  - Homework Lab: (No External Services)
    - Procurement (Purchases with multiple items)
    - Insurance (Claims with payments or fraud detection)
    - Banking (accounts with deposits and withdrawals)

  - Final Project
    - Use some external services
      - Internet or Custom Servlets

---

**Review: A Key Challenge in Business Process Management**

*Many Stakeholders in an Enterprise*

- Executive
- Process Owner
- Business Architect
- Solution Designer
- IT Architect
- Systems Integrator

Bridge the Gap between Business and IT

- Speak in terms of
  - “Functional Decomposition”
  - “Business Components”

- “Impedance Mismatch” !!

- Speak in terms of
  - “Workflow”
  - “Activity-flow”
  - “DB schema”
Basic Challenge: Various disjoint BPM models

*Disjoint Models lacking coherence*

- Data Models
- Organization
- Rules
- Statemachine
- Metrics

Lack of coherence adds substantial complexity to an already very complex environment

Solution: Unified Business Construct

Business Entity: (e.g. Purchase Order)

Alignment of Models
Contextualized as a Business Entity
Coherence Achieved
**Brief comparison of BPM approaches**

**Process-Centric Approach**
- **Business Data is**
  - NOT the primary focus
  - Business data is merely an after thought

- **Process Steps are the main concern**
  - *What* do humans do in the business
  - *What* systems need to be integrated

**Artifact-Centric Approach**
- **Business Data is**
  - The PRIMARY focus

- **Process steps occur in context**
  - The “Business Entity” needs which humans to do something to it.
  - The “Business Entity” needs to integrate with what certain systems.

---

**Review: Hotel Scenario**

Hotel

Guest Stay

- Check In
- Check Out
- Purchase

Front Desk Clerk
- Create
- Settle

Cashier
- Create

Guest

Non Guest

Paid Charges
- Create

Pended Charges
- Create

Audit Report
- Get and Post Charges To Folio
- Audit

Night Auditor

Check In

Check Out
Check in- Checkout Process
(Process-Centric Approach)

Waiting for Checkout Request

Long Running Flow:
- Remains active until Guest checks out
- The Guest Stay information is lost in the process instance data

Point of Sale (POS) Process
(Process-Centric Approach)

POS modeled as separate process as this can be instantiated independently any number of times.
Night Audit Process – Modeled as separate process
(Process-Centric Approach)

Invoked in Batch via Scheduler

Posting to Folio

- Multiple instances of the process created for each day for each guest
- No direct link between check in process and night audit process.

Summary
(Process-Centric Approach)

- Discrete/Disjoint Processes
- Data is an after thought
- Guest stay information lost in long running process instances
- Lots of additional coding needed to integrate to Databases and Services
Entity-Centric approach of Hotel Scenario

**Identify key Business Entities**

- **Guest Stay**
  - Information
  - States

- **Guest Folio**
  - Information
  - States

- **Charge**
  - Information
  - States

- **Night Audit**
  - Information
  - States

**Guest Stay Entity**
(Artifact-Centric approach)
**Folio Entity**
( Artifact-Centric approach)

- **Default info**
- **Guest Info**
- **Room Info**
- **Charge Info**

**Lifecycle**

**Charge Entity**
( Artifact-Centric approach)

- **Default info**
- **Guest Info**
- **Charge Info**
- **Tax Info**

**Lifecycle**
Night Audit Entity
(Artifact-Centric approach)

Information Model

Lifecycle

Folio info
Guest Info
Reconciled Charge Info

Night Audit

Started → Room Charges Created → Charges Reconciled → Completed

Determine Entity Interactions

- Direct link between check in and night audit.

Guest Stay

Started → CheckedIn → CheckedOut

Night Audit

Started → Room Charges Created → Charges Reconciled → Completed

Guest Folio

Started → Active → Settled

Create Folio → Settle Folio

Add Charges

Get Pending Charges For Guest

Charge

Started → Pended → Paid → Lost → Canceled

Create POS Charge

Post Charge
Some comparison points

- **Process-Centric Approach**
  - Process flows act as controllers
  - Data is an after thought
  - Some Operational data lost in long running process data
  - Humans Work on "blocked" tasks in long running flows
  - Performance of long running flows not desirable

- **Artifact-Centric Approach**
  - Entities act as controllers
  - Data is Core
    - Business Entities accessible in DB
  - All Operation Data store in Business Entities
    - Queryable, Trackable, Measurable
  - Humans work on Business Entities that are ready
  - Performance of Entities (info, lifecycle, micro flows) considered acceptable

---

Innovator’s toolkit for Business Process Modeling
(Democratization of Innovation -- Eric Von Hippel)

- **Manufacturer-centered innovation**
- **Innovator-centered innovation**
Radical Simplification of Tools and Runtime
(Supporting Business Process Management Applications using Entity Centric Modeling)

Dreaming of being lighter and more Agile

Reduced set of BPM abstractions to define and create BPM solutions.

Siena Entity (The Core of Siena)
Services

- **Service Definitions**
  - Flow Services
  - Data Services
  - External Services

- **Service Invocations**
  - From Tasks
  - From States

Basic Flow Patterns

Business Entity
Advance Flow Pattern

- One Flow can invoke multiple transitions based on Flow outcome
- Each Transition can in turn have Independent Transition Flows as well

Basic Data Access Patterns

Data Access Service Always Available

State Specific Data Access Service
Determine Entity Interactions

- Direct link between check in and night audit.

Business Entities give context for Service Invocations
External Service Integration (REST and WSDL)

Large Collection of Externals Services Contextualized by Business Entities
Time to Install Siena

- Pass USB Sticks around the room and install onto your laptops
  - Install/Unzip into C:\SienaWeb
    - Double click startup.bat  UID: defaultAdmin  pwd:

- Follow Instructor on Presentation Screen
  - Hello World example with 1 artifact
    - Information Model
    - Lifecycle Model

- Work in groups of two or three
  - Discuss steps among each other

- **Possible Homework Assignment**  Due on Lecture #4 (Review)
  - Homework Lab:  **(No External Services)**
    - **Procurement**  *(Purchases with multiple Line Items)*
    - **Insurance**  *(Claims with Payments and Fraud detection)*
    - **Banking**  *(Accounts with Deposits and Withdrawals)*

Lecture #3

- **More Siena Examples  (Patrizia and Alessio)**
  - Radio Sapienza
  - Relay Race
  - Color Bricks
  - Clinical

- **Review Siena Meta Model**
  - Review model elements
  - Review an instance of Siena.XML

- **Model Driven Architecture**
  - Explain Models an MDA.....Explain traditional Models to Code Generation

- **Model Driven Execution**
  - *Explain Model Driven execution.*
Radio Sapienza Overview (Patrizia and Alessio)

Advertisers

Radio Sapienza

Music Companies

???(subscription fee)

Music Listeners

Relay Race Overview (Patrizia and Alessio)

• All Players Run at Random Speeds
• Siena Controls Relay Race Servlets/External Services
CLINIC

- Manages information about the examinations: data of patients, description of the reports, admissions to a ward.
- Manages the ambulances: external services to find destination address and to visualize the map.

COLOR BRICKS

Columns: 4
Rows: 4
Cells: 10

- Plays a game respecting the constraints.
- The user can choose the number of cells and the dimension of the matrix.
- Purpose: Siena is able to manage a big amount of instances.
**Information Model: PLAYLIST**

- **Name** (string):
  - name of the playlist.

- **DurationTot** (long):
  - total duration of the playlist.

- **PlayerID** (long):
  - ID of the player that is playing the playlist.

- **TrackList** (TypeTrack):
  - list of tracks that compose the playlist.

- **TrackPlayed** (TypeTrack):
  - informations of the track that is actually played.
Information Model: **TRACK**

- **TempPlaylistID** (long):
  - ID of the playlist that is playing the track.

- **StartTime** (dateTime):
  - date and time in which the track is played

- **PauseTime** (dateTime)

- **ResumeTime** (dateTime)

- **RemainingDuration** (long):
  - remaining duration of the track after a "resume" action.

- **TrackInfo** (TypeTrack):
  - informations of the track.

Information Model: **PLAYER**

- **NumberOfTracks** (int):
  - Number of the tracks played.

- **PlaylistID** (long):
  - ID of the playlist that the player is playing.
**EXTERNAL SERVICE:** **BrowseAmp**

- WinAmp is a media player.
- It is controlled by the plugin “BrowseAmp”.
- BrowseAmp offers a RESTful service
  - Allows us to control WinAmp with simply URLs
    - Play
    - Stop
    - Pause
    - Resume

**OTHER SERVICES:** **ODDCAST E ICECAST**

- We use two free softwares in order to simulate a radio station on internet.
- Oddcast sends the parameters to WinAmp that it is playing on the port 8002.
- IceCast broadcasts on internet the stream of data that arrives on that port.
- The users can listen the playlist using the link: [http://151.100.59.92:8002/Radiosapienza.m3u](http://151.100.59.92:8002/Radiosapienza.m3u) (address of the server in which all the applications run).
State Diagram: **PLAYLIST**

State Diagram: **TRACK**
State Diagram: PLAYER

RELAYRACE: ARTIFACTS

- PLAYER
- TEAM
- RACE
**Information Model: PLAYER**

- **Name** (String):
  - Player's name.

- **Speed** (Float):
  - Player's velocity.

- **Time** (Long):
  - Space / Speed (m/s).

- **TeamID** (Long):
  - ID of the player's team.

**Information Model: TEAM**

- **Name** (String):
  - Name of the Team.

- **TotalTime** (Long):
  - Time spent to finish the race.

- **RaceID** (Long):
  - ID of the race in which the team is playing.

- **PlayerList** (PlayerType):
  - List of players of the team.
Information Model: **RACE**

- **Name** (String):
  - Name of the Race.

- **TeamList** (TeamType):
  - List of team that takes part in the race.

- **Results** (Results):
  - Arrival ranking of the teams at the end of the race.

**EXTERNAL SERVICES**

- We use Java JSP and Java Servlet in order to implement external services useful for our application.
- Siena manages the race, controls the competition and sends to the JSP pages all the attributes required.
- The external services are:
  - **RaceService**:
  - **ShowRace**:
    > [http://localhost:8080/OpenURL/Open](http://localhost:8080/OpenURL/Open) opens a window that shows the race.
  - **Winner**:
    > opens a window with all the informations about the winner.
State Diagram: **PLAYER**

```
CREATED  \-----\  TRAINING  \-----\  ON
       |      \      |        \        \ COMPETITION
STOP    TO TRAINING \STOP RUNNING TO RUNNING

State Diagram: **TEAM**

```

```
CREATED  \-----\  READY  \-----\  PLAYING  \-----\  STOP
       |      \      |        \      \ RACE SERVICE
TO READY \      \ TO PLAYING \      \ INVOKE
TO CREATED \      \ STOP \      \ INVOKE
WINNER CONTROL  \      \ WINNER
```
State Diagram: RACE

- CREATED
- ADD TEAM
- READY TO START
- SHOW
- IN RACE
- START
- RACE SERVICE
- INVOKE
- SHOW RACE
- RESET RACE
- RUN AGAIN
- STOP
- PLAY AGAIN
- RESET

Relay Race

Siena Architecture Diagram

Modeling Clients
- Artifact Model Editor
- SmartGWT

Custom Application Client
- Custom UI (Rich Client)
- Default UI (dynamically rendered)

Execution Clients
- UI Toolkit (Artifact-centric widgets & helpers)
- Google Web Toolkit

Model Execution Engine
- REST & WSDL
- External Services
- Artifact Model Management
- Data Access
- Behavior Services (lifecycle, flow, GSM engines)
- Access Control
- Rule Evaluation (JEXL, OCL)
- Monitoring & Analytics
- Extension Points

Artifact Instances
- Artifacts
- Siena Models
- XSD
- XML
- REST
- WSDL
What makes up a Siena Application?

- Service I/O XSDs
- Artifact XSDs

Siena Meta-Model as XSD

Artifact Schema as XML

- Siena (Application) XML
- External Services
- Artifact
- Information Model
- Lifecycle Model

Siena Schema (Meta-Model)

Composite Application

- Modeling Tools
  Constrained by Meta-Model

- Execution Engine
  Uses Meta-Model to Execute Model Instances
Let's now inspect the model using an XML Editor.

Explore the development environment in Eclipse.
Generate Java from Siena Meta-Model

Siena Meta-Model as XSD

Generate Java Classes

• Java Classes used to process Siena Model Instances

Modeling Clients

XSD 2 Java

Model Driven Architecture

The Model

Deployment Zip File

• Siena (Application) XML
  • External Services
  • Artifact
    • Information Model
    • Lifecycle Model
    • ...

Artifact Schema as XML

• Service I/O XSDs
• Artifact XSDs

XSD XSD XSD
Traditional Approach: **Generate Model Into Code**

- Process Owner
- Now what do I do with all this stuff?
- The Model

- JavaScript
- WSDL
- HTML
- XML
- JSPs
- Servlets
- EJBs
- ACL
- BPEL
- XSD
- SQL
- Lots and Lots of **JAVA**

Siena Approach: **Direct Deploy and Execute Models**

- Process Owner
- So Simple!!!! I can Innovate and Test Immediately!
- The Model

- Deployment
- Zip File

- Direct Deployment and Execution of Models

Lecture #4

- **Review homework**
  - DEMOs
  - Questions and Answers

- **Siena System Internals**
  - Questions and Answers

- **Web Services**
  - Service Facade
    - RESTful
    - WSDL
  - Service Invocation
    - Internal
      - Flow, Data
    - External
      - RESTful and WSDL

- **Future Artifact Designs**
  - GSM style
    - Event Driven
    - No Wires/Transitions

- **Artifact Design Patterns**
  - One-One
  - One-Many
  - Many-Many

- **Feedback**
  - Course improvements
    - Content
    - More Systems?
    - More Database?
    - More Services
    - More???

**Review Homework Problems**

- Homework reviews/demos
- **PLEASE ASK QUESTIONS**
- Questions and Answers
  - Procurement (Purchases with multiple items)
  - Insurance (Claims with payments or fraud detection)
  - Banking (accounts with deposits and withdrawals)
Procurement

Purchase Order

LineItem

Shipment

Insurance

Claim

Fraud

Assessment Report
Banking

Siena System Internals

- Questions and Answers
  - Meta-model
  - Model Instances
  - Modeling UI
  - Model Execution Engine
  - Default Execution UI
  - Custom UIs
Services

- **Service Definitions**
  - Flow Services
  - Data Services
  - External Services

- **Service Invocations**
  - From Tasks
  - From States

REST & WSDL Services

Registering External Services

- **REST**
  - Specify Service End Point URI
  - Import XSD

- **WSDL**
  - Point to Remote WSDL File
  - Import WSDL
**Binding and Mapping External Services**

- **REST**
  - Specify Service End Point URI
  - Import XSD

- **WSDL**
  - Point to Remote WSDL File
  - Import WSDL

**Client Toolkit: Siena API Façade**

*(RESTful Siena)*

- **SienaServiceClient** API
  - Generic RESTful API to call Siena
  - Support for XML Input, XML Output, JSON Input, JSON Output
  - API:
    - `SienaServiceClient sienaClient = new SienaServiceClient();`
      - Restful Service Façade
    - `sienaClient.invokeXml();`
      - Invoke Services (flow services, data services, transition services)
    - `sienaClient.retrieveListXml();`
      - Retrieve Artifact Instance Lists
    - `sienaClient.retrieveXml();`
      - Retrieve a single Artifact Instance
    - `sienaClient.saveXml();`
      - Save a single Artifact Instance
Inspect Code: using eclipse

```java
static String appName = "ClientTest";
static String dataItemId = "MyArtifact";
static String serviceId = "MyArtifact-ANY-to-Created-0Transition";
static String inProgressServiceId = "MyArtifact-Created-to-InProgress-0Transition";

String ArtifactXML = "<MyArtifact
    attribute1='Via Labicana' attribute2='Roma' attribute3='Terry' attribute4='Heath'>
</MyArtifact>";
// Make a New Client
SienaServiceClient client = new SienaServiceClient("http://localhost:8080/SienaWeb", "defaultAdmin", null);

// Invoke Any to Created
String invokeXmlOutputMessage = client.invokeXml(appName, serviceId, URLEncoder.encode(ArtifactXML));

// Extract newly created ID from new Instance
String id = parseIDFromXML(invokeXmlOutputMessage);

// Invoke Created to InProgress
String inProgInputXML = "<MyArtifact ID='" + id + "'/>";
String inProgOutputXML = client.invokeXml(appName, inProgressServiceId, URLEncoder.encode(inProgInputXML));
```

Run Example Client from Eclipse

- Jump to IDE
- RUN
Client Toolkit: Siena API Façade
(RESTful Siena)

- **WSDL Siena**
  - *Solution Specific WSDL files*
    - Generated into deployed solution
    - Use your favorite IDE to bind to WSDL files and invoke
    - Generate JavaProxy Web Service Client
    - Begin to Invoke WSDL Operations onto the Solution Specific Artifacts

Artifact Relationship Patterns

- **(One to One)**
  - Insurance Claim can related to 1 Fraud

- **(One to Many)**
  - *Parent / Child*
    - Purchase Order
      - Line Items

- **(Many To Many)**
  - *Purchase Order*
    - Each Purchase Order can be delivered in 1 or more shipments
  - *Shipment*
    - Each shipment can contain Line Items from different Purchase Orders
Future Artifact-Centric work

- Further Siena Web Tooling Features
- Optimize Siena Engine
- Project ArtiFact™
  - Declarative Approach (no wires or transitions)
    - Guards, Stages and Milestones

What is Project ArtiFact™ GSM

- Hierarchical Units of Work
  - Units of Work (*Stages*)
    - Launch by Conditions (*Guards*)
    - Completed by Expressions (*Milestones*)
Declarative Artifact-Centric as a unifying basis for future BPM

Rules in vocabulary of the artifacts

Analytics can use artifact info models directly

Declarative Artifacts

Theory of optimization

Hierarchical Stages (Units of Work)
with contextual Rules

A stage focuses on a natural, small cluster of related rules
More on the “guarded” style for lifecycle specs

- **Can put a variety of rules / conditions into the “guards”, e.g.,**
  - Can only enter Customer login stage once
  - If you change Cart you must revisit Shipping Pref
  - Cannot enter Payment Pref until either you are logged in or put stuff in Cart

- **Can vary the guards based on region, customer category, etc.**
  - Variation at any level of hierarchy

Flow charts and state machines useful in some contexts; “Ad hoc” style useful in other contexts

Feedback

- Please Fill out the Feedback Form

- **Course improvements**
  - Too Slow?
  - Too Fast?
  - More Content?
  - More Systems?
  - More Database?
  - More Services
  - More???