A Modeling Environment for Patient Portals

Sean Duncavage, Janos Mathe, Jan Werner, Akos Ledeczi, Bradley Malin, Janos Sztipanovits

Dept. of Electrical Engineering and Computer Science
Dept. of Biomedical Informatics
Vanderbilt University
November 14, 2007
Introduction

- Clinical Information Systems (CIS) integrate IT with organizational components across healthcare environments

- Potential to increase productivity and patient safety, but…

- Must handle complex infrastructures & human interactions
  - Poorly-designed CIS can cause major system and care errors
    - Difficult to detect what, or where, errors occur
    - Not easy to audit, evolve, or reconfigure

- Goal: Provide a formal way to represent and evaluate CIS
  - Separate high-level abstractions from implementation details
  - Reason about the current, but also future, system
Overview

- Introduction
- Background & Motivation
  - Portals
  - Service Oriented Architectures
- Methods
- Results
- Discussion
- Conclusions
Why Portals?

- Online availability and archiving of medical records is a complex societal challenge
  - Potentially affects the health and well-being of every citizen
  - Embeds the need for critical infrastructure
  - Substantial computer and network security requirements
  - Regulatory and ethical mandates for data privacy protection

- Growing trend in healthcare to address the challenge is the "patient portal"
  - Secure and personalized customer services over the Internet
  - Opportunity to deploy individualized services
  - Can implement diverse health-related functions
  - Patients are proactive in the maintenance of their medical records and care decisions
Portals, Privacy, Security, & Access

- NIH has supported projects to provide patients with secure access to their medical records via the Internet for over a decade
  - PCASSO (UCSD)
  - PATCIS (Columbia)
  - My Doctor’s Office (Colorado)
  - Web messaging (UC Davis)

- Summary of Findings:
  - Personal health information has value to patients
  - Patients want electronic access
  - Providers fear being overwhelmed by patient interactions and ‘information toxicity’ will occur when patients see technical info they don’t understand
  - Security breaches not reported (yet) in portal systems
Behind the Portal: Workflows & Services

Appointments Messaging EMR access

Electronic Medical Record System

 HIPPA Policy
 Local Policy
 4-Eyes-Principle...
Overview

- Introduction
- Background & Motivation
  - Portals
  - Service Oriented Architectures
- Methods
- Results
- Discussion
- Conclusions
Existing Architecture & Framework

- Service Oriented Architectures (SOA)
  - Rely on existing standards, such as SOAP, WSDL, WS-Security, XACML
  - Exploit open-source implementation of integration platforms (Active BPEL, Apache ODE)

Standards do not guide integration of security technologies with applications.
SOA, Models & the Clinical Realm

- SOA is applicable to existing CIS*
  - Aids the design of medical decision support systems
  - Facilitates the integration of standards, such as HL7

- Model-based approaches support documentation, communication, and standardized development of health information systems**
  - Model-driven architectures: Generic approach isolates technology changes from logic, but no unified application;
  - Business Process Modeling: Process abstraction via standardized platforms, but excludes organizational resources, data typing, & business rules

Overview

- Introduction
- Background & Motivation
- Methods
  - Our Software: MODECIS
  - MODECIS Abstractions
  - MODECIS Infrastructure
- Results
- Discussion
- Conclusions
MODECIS: Model-based Design Environment for Clinical Information Systems

- Clinical Information Systems
- Domain X
- Domain Y
- ... 

Models represent the CIS specification

Model translators map the specification onto implementation abstractions of SOA standards

- Data Models
- Service Models
- Deployment Models
- Organization Models

- Translator

- BPEL4WS
- WSDL
- XACML
- XML

Discrete Manufacturing
General Architecture

- Challenges of using the existing infrastructure
  - SOA abstractions may not fit perfectly to the domain
  - Heavy-weight
  - CIS domain has unique requirements
    - Static policy can alter, and usually restrict, service orchestration design languages

- Domain specific modeling abstractions expressed in formally defined DSML-s

- System level
  - Standard SOA-based business process modeling platform

- BPEL Modeling Platform
  - BPEL4
  - WSDL
  - XACML + XSD
  - XML Conf.

- Policy Enforcement Point
  - External
  - Internal

- Policy Decision Point

- Policy Repos.

- BPEL Process Manager

- Config. Engine

- Standard SOA-based BPEL execution platform (simulation and fast prototyping)
Layers of Abstraction

Domain specific modeling abstractions expressed in formally defined DSML-s. TRUST research focus

SOA-based, standard, business process modeling platform

SOA-based, standard, execution platform (simulation/fast proto.tng)
From Language Design to Workflow Execution

1. Via model Based Design (MBD) we express domain specific modeling (DSM) abstractions as formal language (DSML)
2. Configure *Generic Modeling Environment* (*GME*), based on DSML, to build domain specific models
3. Models are translated to Service Oriented Architectures (SOA) standards, including
   - Business Process Execution Language (BPEL)
   - Web Services Definition Language (WSDL)
   - eXtensible Access Control Markup Language (XACML)
4. Translated models can be used to drive an execution engine / platform
5. Models can also be translated for verification or simulation system
1. **Modeling environment**

“Metamodels” define the domain specific modeling language and define the abstract syntax of domain models.

User models represent a specific CIS instance through a set of modeling abstractions.

**Technology Applied**

Generic Modeling Environment (GME)
MODECIS Architecture

1. Metamodel Editor
   - Metamodels
   - Metamodel Translator

2. Translators
   - Transform user models into BPEL deployment code
   - Create XACML policy decision points

3. Execution Environment
   - Front End
   - Execution Engine
     - Process Manager
     - Policy Decision Point
   - XACML Policy Set
   - Formula (Prolog Solver)

Technology Applied
- GREAT
- Builder Object Network (BON) interface
MODECIS Architecture

3. Execution Environment

BPEL execution engine

Policy execution engine

Web server for user interaction

Technology Applied
- OracleBPEL
- ActiveBPEL
- SunXACML
Overview

- Introduction
- Background & Motivation
- Methods
- Results
- Discussion
- Conclusions
Technical goals

- Eliminate manual processes
- Secure information exchange with patients
- Log communications in patients’ charts
- Monitor patients’ conditions remotely

Growing set of individualized services including messaging, scheduling, billing, test results, prescription refills

> 25,000 enrolled patients & approx. 50,000 care providers
Design and Development Process

1. Design Workflow (System & Data)

Automatically Generate Code for System Evaluation

Research Team

MHAV Team

Review By MHAV / VUMC Administrators and Personnel

Recommendation to MHAV System Designers

MHAV System System Revision
Service Abstractions

- Design Environment (GME) – Combined View

- Service models capture business logic:
  - Workflows of hospital staff and portal-related software
  - Control flows for service invocations
  - Data flow for transmission of information
Organizational Abstractions

- **Interdepartmental**: communication between separate clinical entities
  - e.g. hardware servers and human care providers in different departments (referrals)

- **Intradepartmental**: information flows within single clinical department
  - Entities modeled with multiple roles to reflect assignments to multiple departments
  - Ex: a billing assistant that works for the gastroenterology and emergency depts.
Data Abstractions

- Data models
  - Specify the information in the CIS
  - Simple and compound data types in hierarchical form

Patient Information
Medical Record Number (MRN)

State Variables
Current System Time (CST)

Compound Variables
\[ URL = MRN + CST + \text{Service Call} \]
Deployment Abstractions

- Deployment models: capture coordination of machines in CIS
- Network Architecture
  - Servers and workstations
  - Service deployment
  - Secure sessions
  - Access control

- Depict hospital servers and workstations with services they provide
- Ex: MHAV server is housed separately than hospital’s EMR servers, but both contribute to patient portal services
MODECIS Example

a simple service that checks the user’s credentials and authorizes access to other services
Overview

- Introduction
- Background & Motivation
- Methods
- Results
- Discussion
- Conclusions
Design Opportunities

- **Perspective:**
  - formal modeling of system designs
  - Policy-driven control of information flows
  - formal modeling of access control and privacy policies

- **Enable systems design that satisfies high-level requirements**
  - privacy, secrecy,
  - integrity,
  - non-repudiation,
  - dynamic access control,
  - rights delegation
Policy Abstractions

- Policy models
- Static policies that can be evaluated based on system specifications
- Dynamic policies that can be evaluated at run-time
Overview

- Introduction
- Background & Motivation
- Methods
- Results
- Discussion
- Conclusions
Conclusions

- MODECIS tool suite provide a graphic modeling environment tailored to CIS
  - Initial support for BPEL and XACML code generation
  - Supports rapidly reconfigurable design of complex clinical environments

- Future Work
  - Create translators for
    - Security enforcement
    - Front-end generation
    - Model verification
  - Disseminate and conduct studies on usability
Acknowledgements

- **Research Team**
  - Sean Duncavage
  - Janos Mathe
  - Jan Werner
  - Akos Ledeczi, Ph.D.
  - Brad Malin, Ph.D.
  - Janos Sztipanovits, Ph.D.

- **MyHealth@Vanderbilt**
  - Jim Jirjis, M.D.
  - Sue Muse
  - Jim Weaver

- **Vanderbilt**
  - John Doulis, M.D.
  - Dario Giuse, Dr. Ing.
  - Jun Kunuvat
  - Dan Masys, M.D.
  - Bill Stead, M.D.
  - Gaye Smith
Acknowledgements

- National Science Foundation
  - NSF CCF-0424422
Questions? Comments?

Brad Malin, Ph.D.
b.malin@vanderbilt.edu