perotsystems®

Healthcare SOA: From Requirements to Deployment

A Service-Oriented Analysis & Design (SOAD) Example

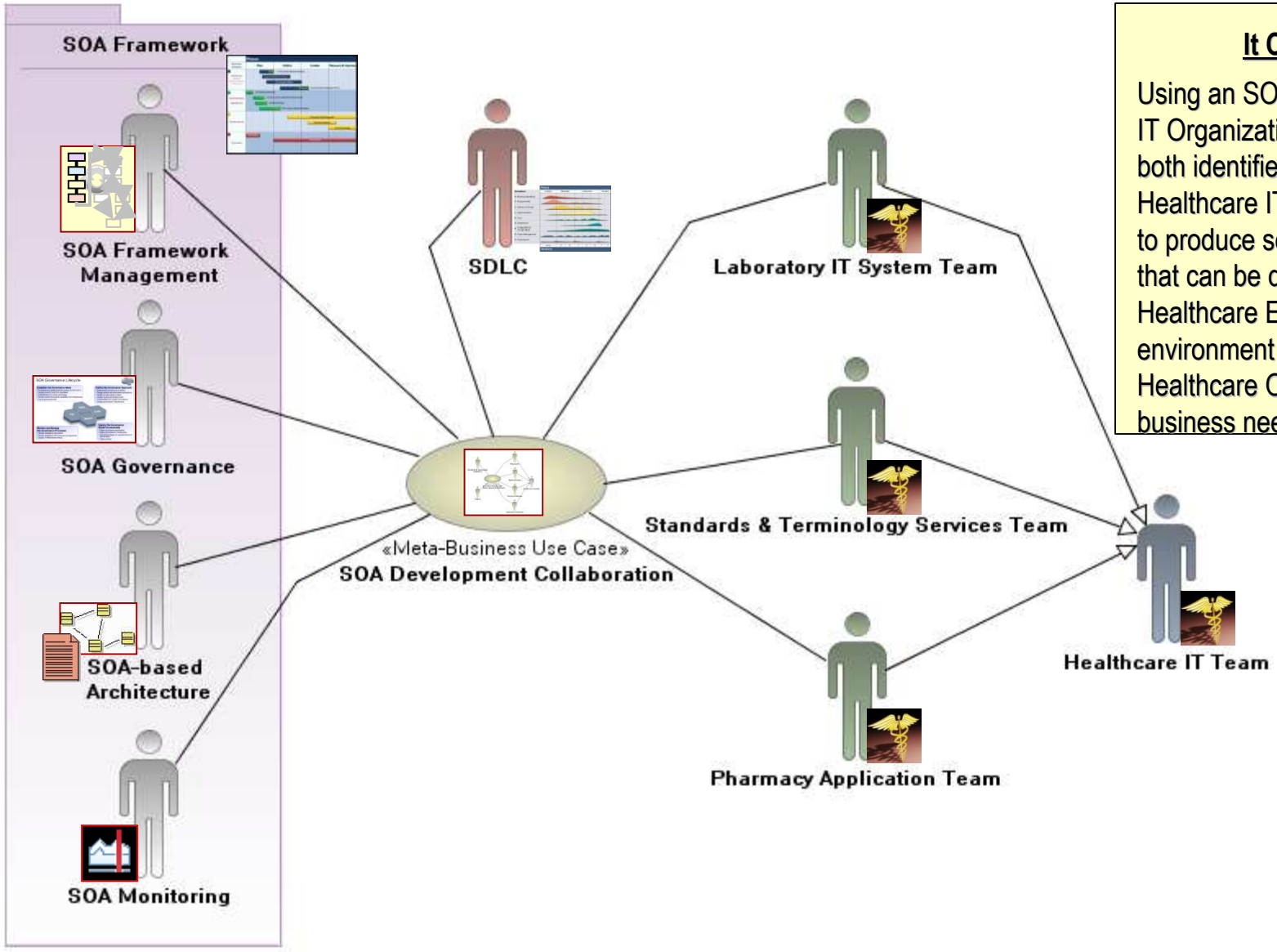
Demetrios Yannakopoulos

Demetrios.Yannakopoulos@mail.ps.net

perotsystems®

Operational Need

Coordinating SOA Assets, Teams, Methods, Theory, and Policies



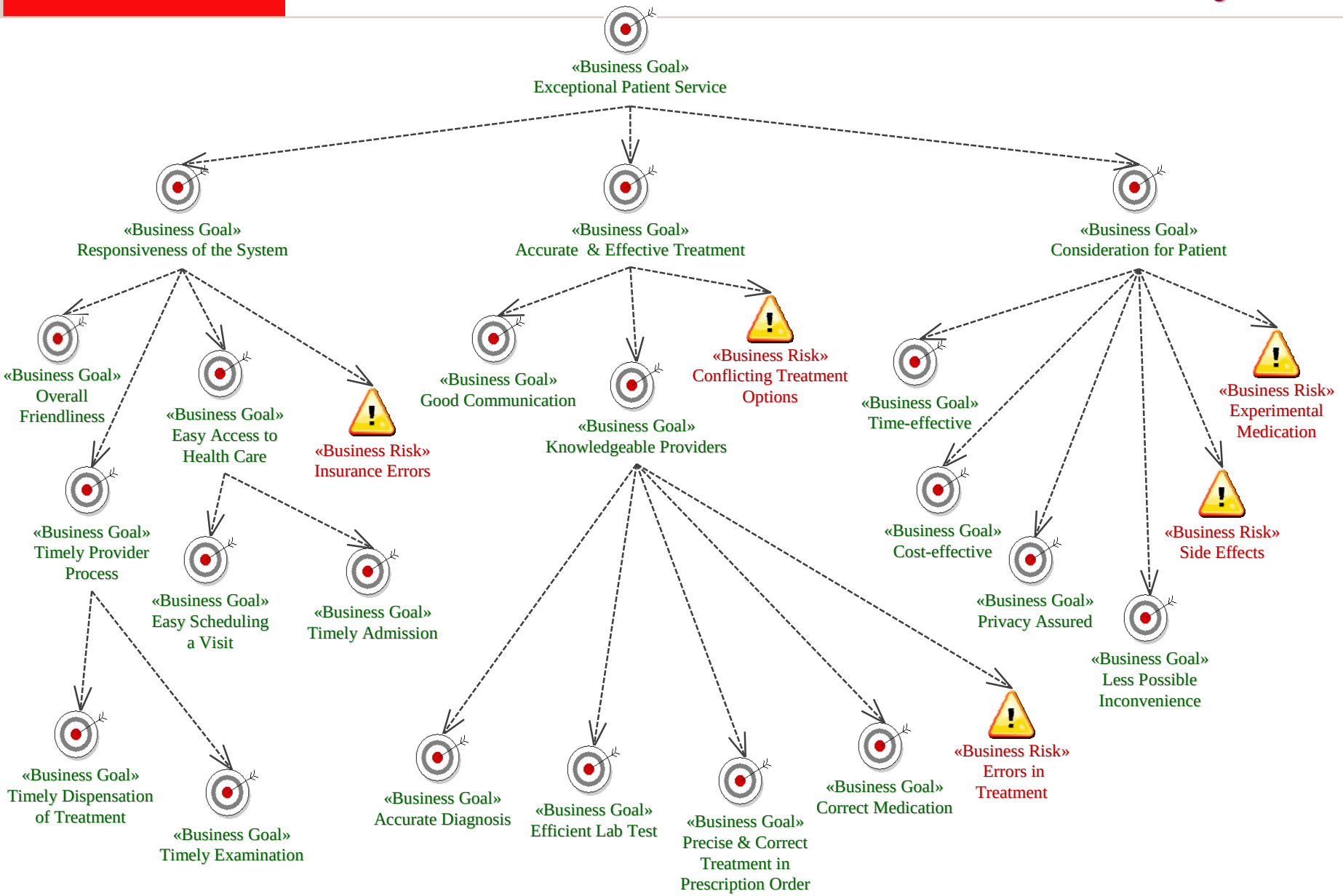
It Can Be Done

Using an SOA Framework and an IT Organization's existing SDLC – both identified as Actors – Healthcare IT Teams collaborate to produce service components that can be deployed in an Healthcare Enterprise IT environment and implement the Healthcare Organization's business needs.

perotsystems®

Business "Goals & Risks" Hierarchy

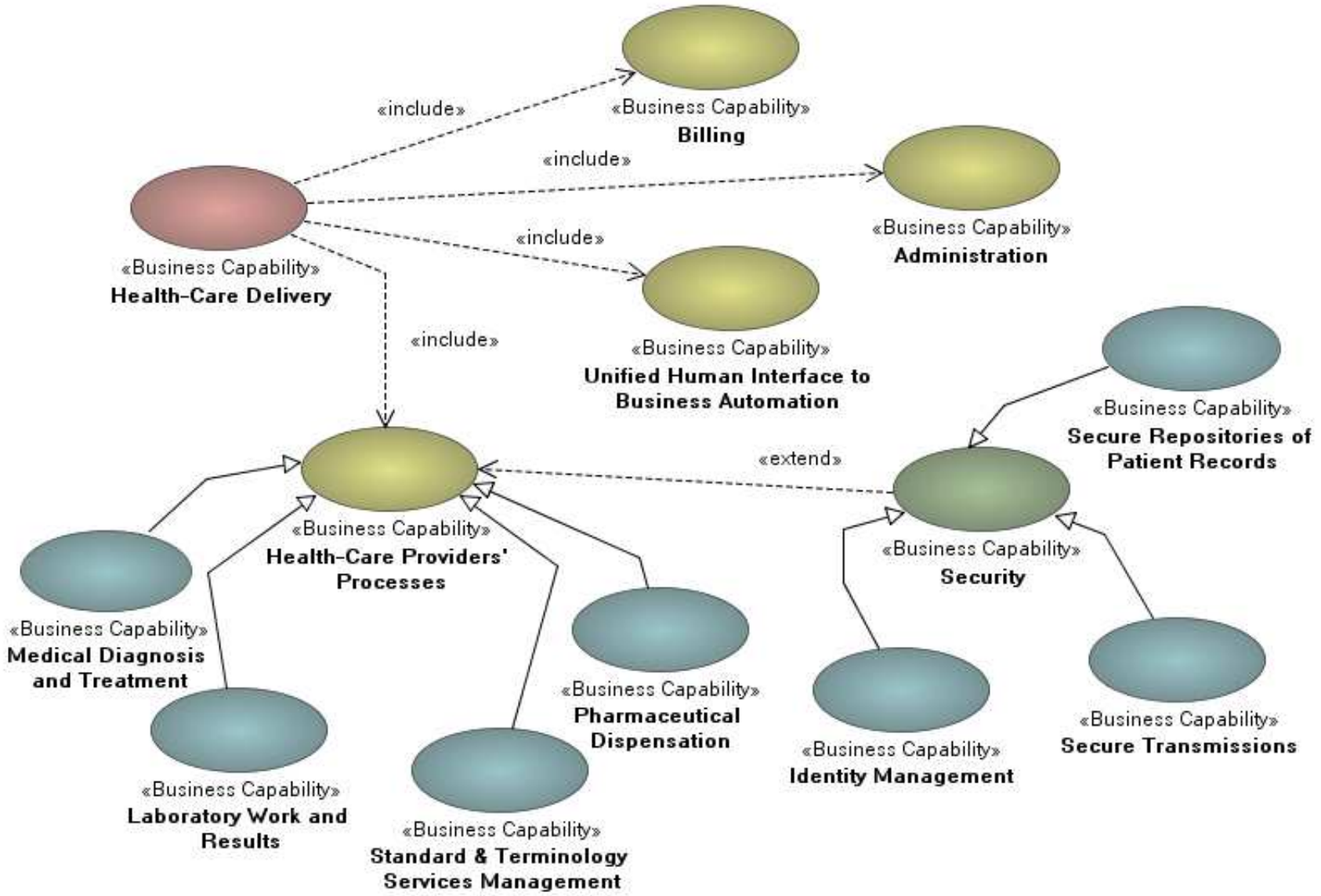
A "General Picture" View of the Organization



perotsystems®

Business Capabilities

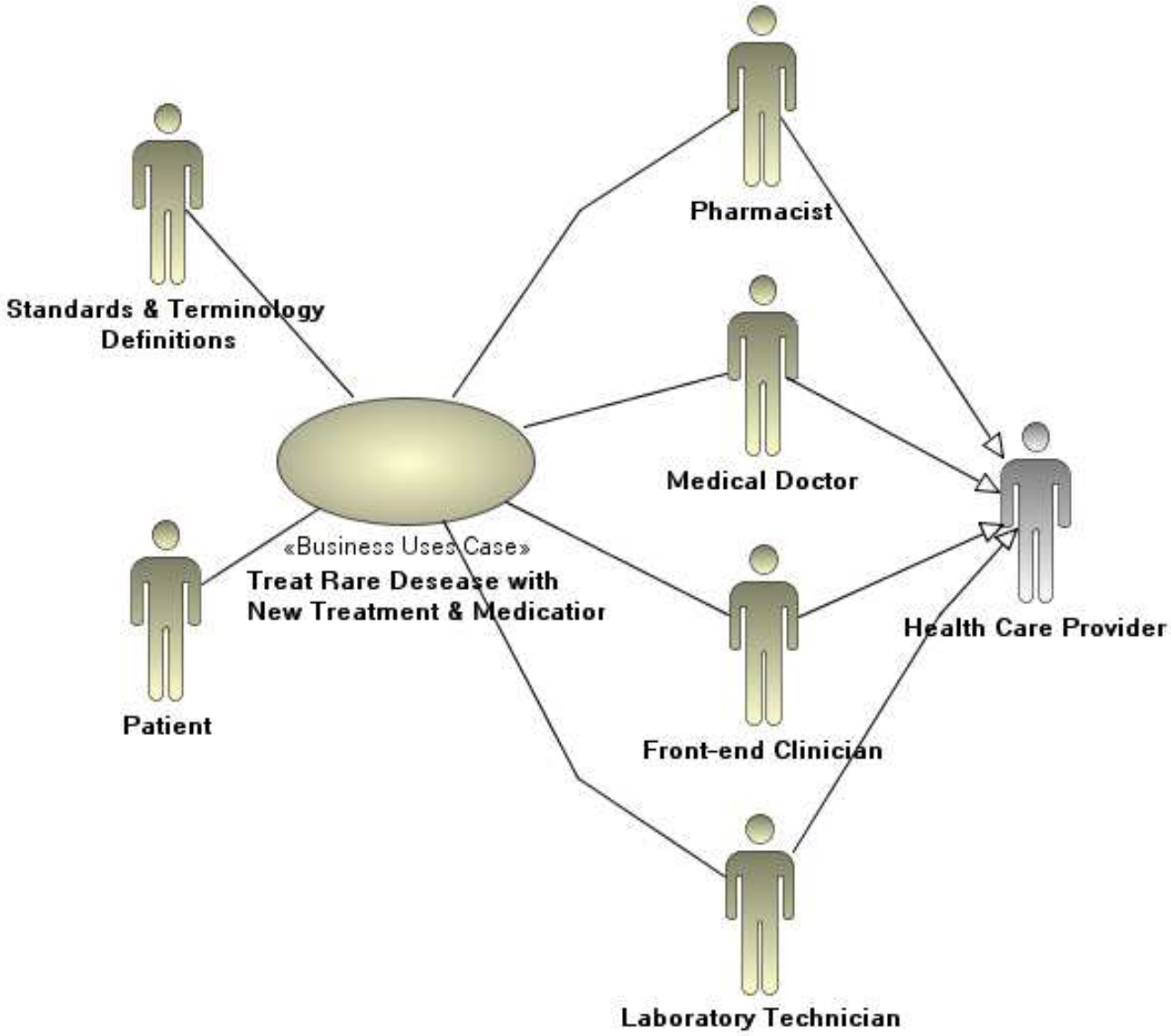
Based on *Business Goals*



perotsystems®

Business Use Case

A View of Specific Business Need



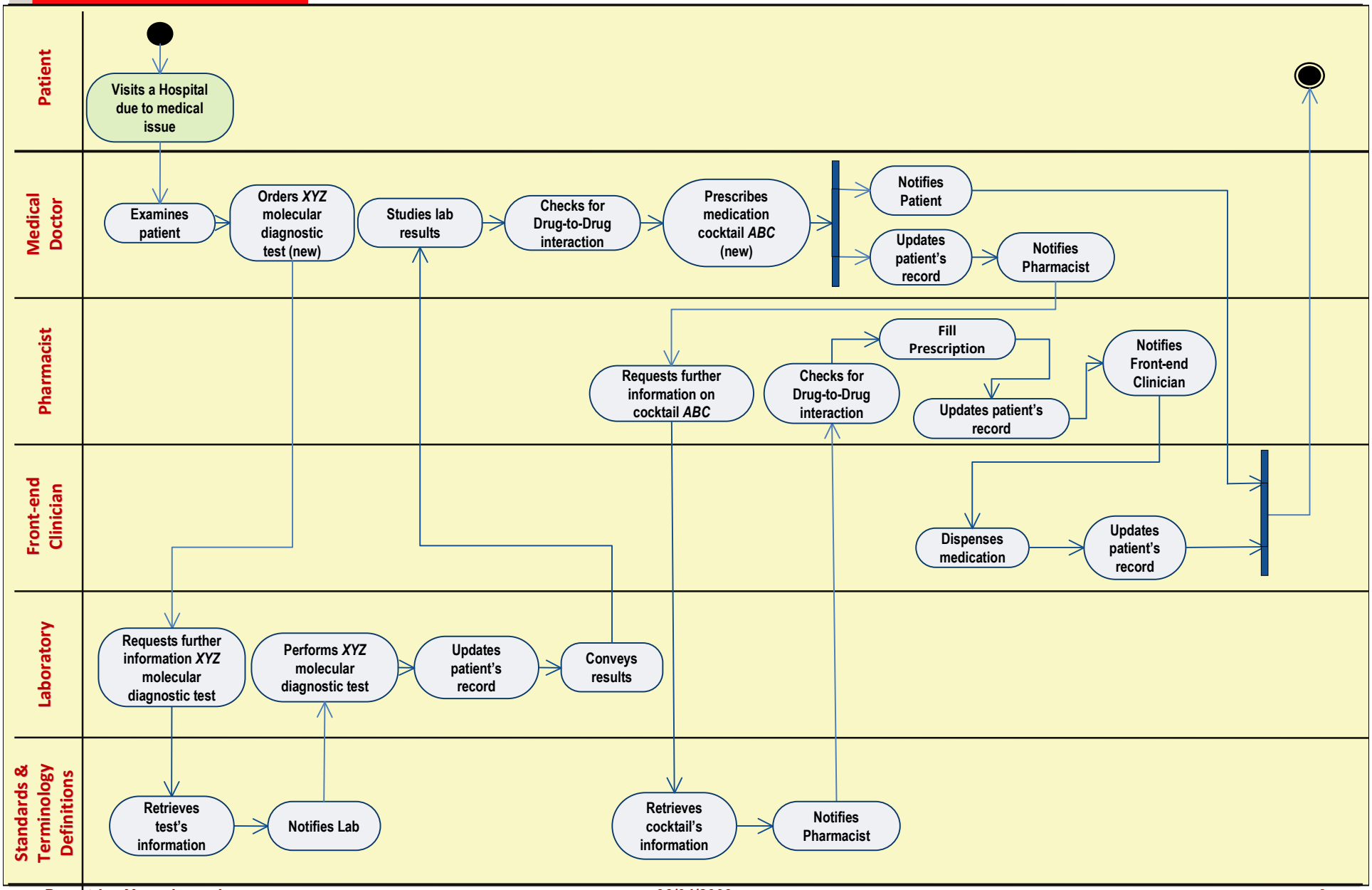
Description

- A patient is examined by a physician.
- The physician's diagnosis and subsequent treatment call for a new lab test.
- The physician notifies Lab.
- Lab, requests information about the new test from Standards & Terminology Definitions, performs the test, and notifies the physician about the results.
- The physician then prescribes a new experimental drug and notifies Pharmacy.
- The pharmacists requests further information about the drug from Standards & Terminology Definitions, and based on this he/she fulfills the prescription.
- The front-end clinician is notified, and dispenses the medication to the patient.

perotsystems®

Business Activity Diagram

Based on the *Business Use Case*



perotsystems®

Business Processes

Based on *Business Capabilities* as well as *Business Goals & Risks*

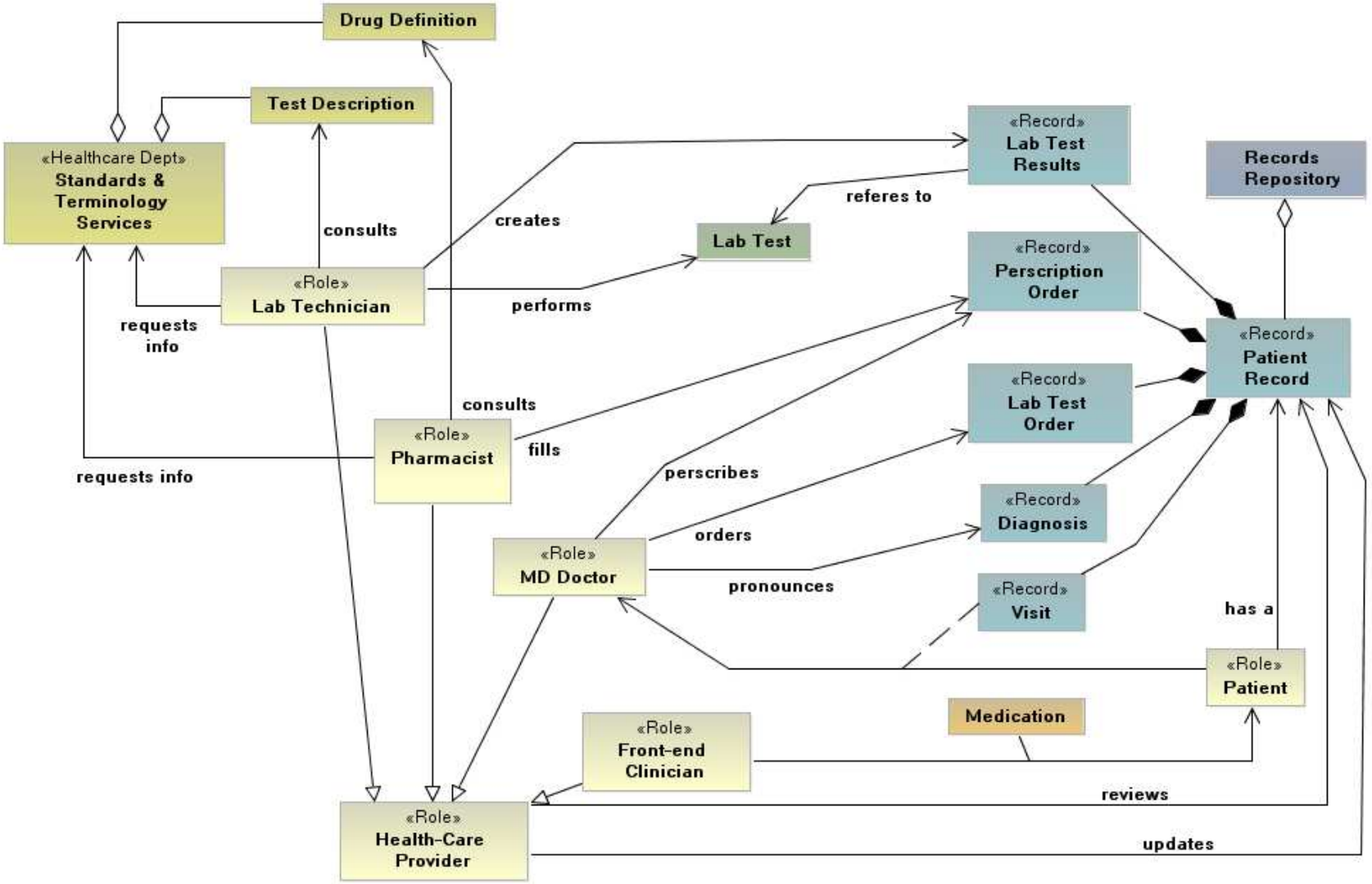
Business Event (*Trigger*)	Business Process	Business Goal	Business Risk	Business Event (*Output*)
Patient's New Visit	**Diagnose Medical Condition**	*Accurate* Diagnosis	• *Erroneous* Diagnosis • *Conflicting Options in* Diagnosis	*New* Lab Test Order
New Lab Test Order	**Perform Lab Test & Communicate Results**	*Efficient* Lab Test	*Erroneous* Lab Test	Lab Test Results
Diagnosis *Requiring Treatment*	**Issue Prescription Order**	*Precise & Correct Treatment In* Prescription Order	*Erroneous* Prescription Order	*Notification to Dispense* Medication
Notification to Dispense Medication	**Fill & Dispense Medication**	*Correct* Medication	*Erroneous* Medication	*Dispensation of* Medication

Business Objects are underlined

perotsystems®

Business Classes

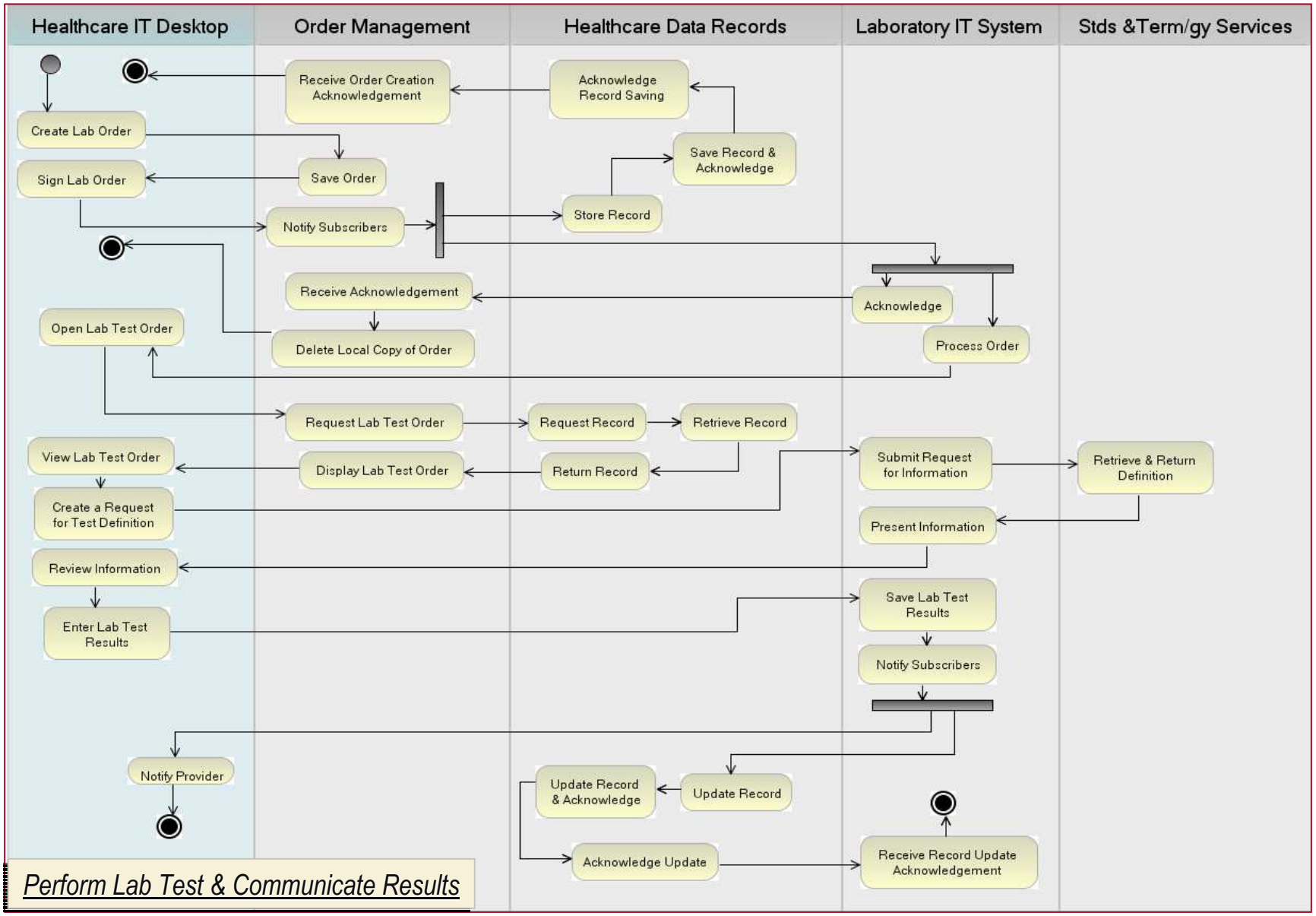
Based on *Business Activity Diagram*



perotsystems®

System Activity Diagram

Based on *Business Processes* as well as *Business Classes*



perotsystems®

Service Implementation Layers

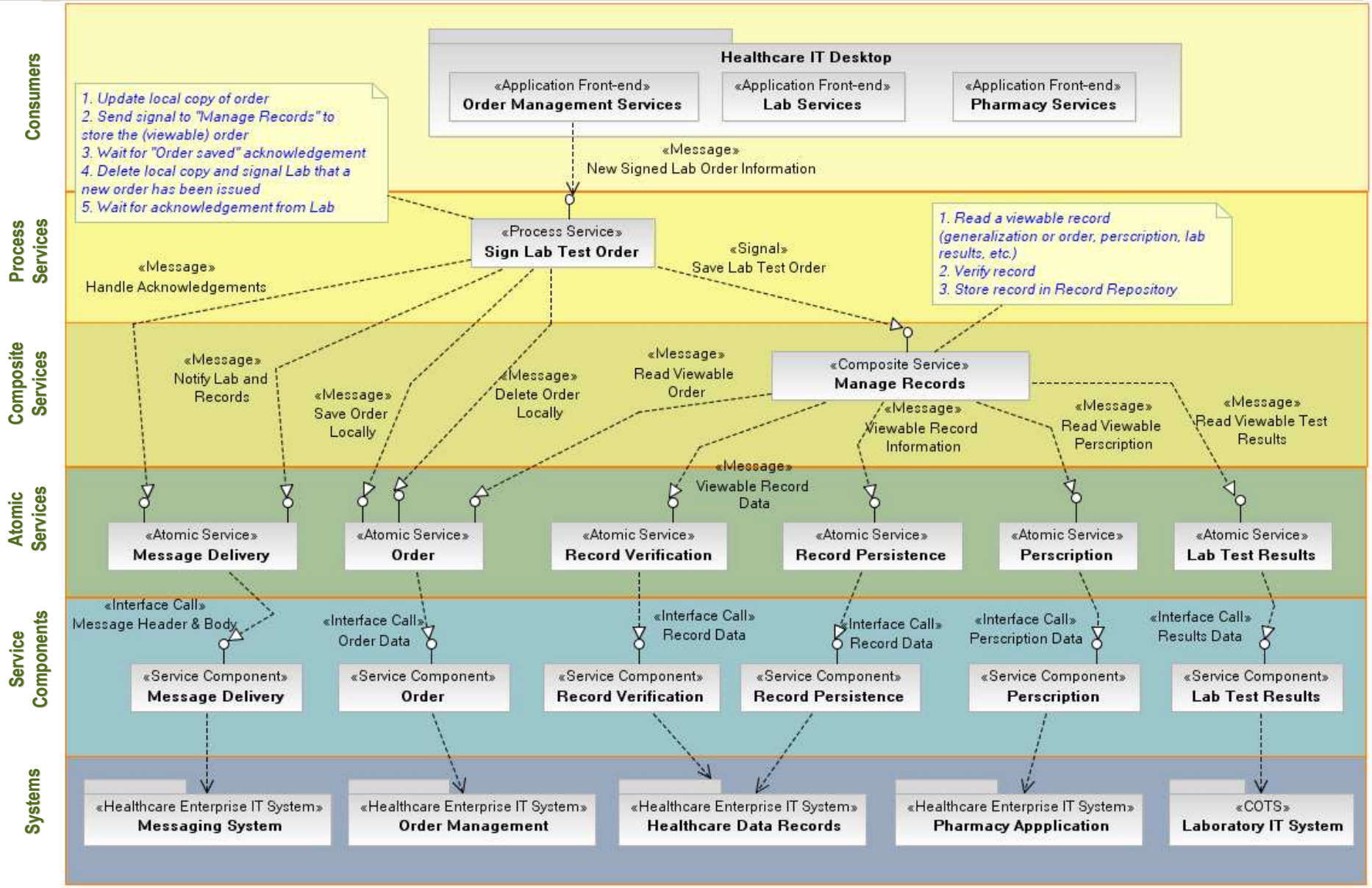
Brief Background

Consumer Layer	Provides the capabilities required to deliver IT functions and data to end users to meet specific usage preferences. It can also provide an interface for application-to-application communication and for human-to-computer interactions. Service consumers can utilize services in the atomic, composite, and process layers. (E.g. The "Order Management Services" Application)
Process Service Layer	They manage a business process flow. They may consist of atomic, composite, and other process services. The process flow may be controlled by one process service (orchestrated) or collaborated between process services (choreographed). They have high frequency of change, low level of reuse. Highly complex to implement and control and maintain their state. (E.g. The "Sign Lab-Test Order" Process Service)
Composite Service Layer	One/more atomic and/or composite services that are bound together in a new service. They act as both service providers of the (composite) services and as a service consumers of its child services. They have a low level of reusability and are infrequently changed. They are moderately complex to implement and are stateless. (E.g. The "Manage Records" Composite Service)
Atomic Service Layer	Self-contained services that do not invoke any other services. They can be simple data-centric services, logic-centric services, or a combination of the two. They are: Highly reusable and infrequency changed. Of low to moderate complexity to implement. Stateless. (E.g. The "Order" Atomic Service)
Service Component Layer	Software components, each of which provide the implementation for, realization of, or operation on a service. Service components reflect the definition of a service, both in its functionality and its Quality of Service (E.g. The "Order" Service Component)
System Layer	Systems that are service enabled: Commercial Off The Shelf (COTS), Government Off The Shelf (GOTS), or custom developed systems (E.g. The "Order Management" System)

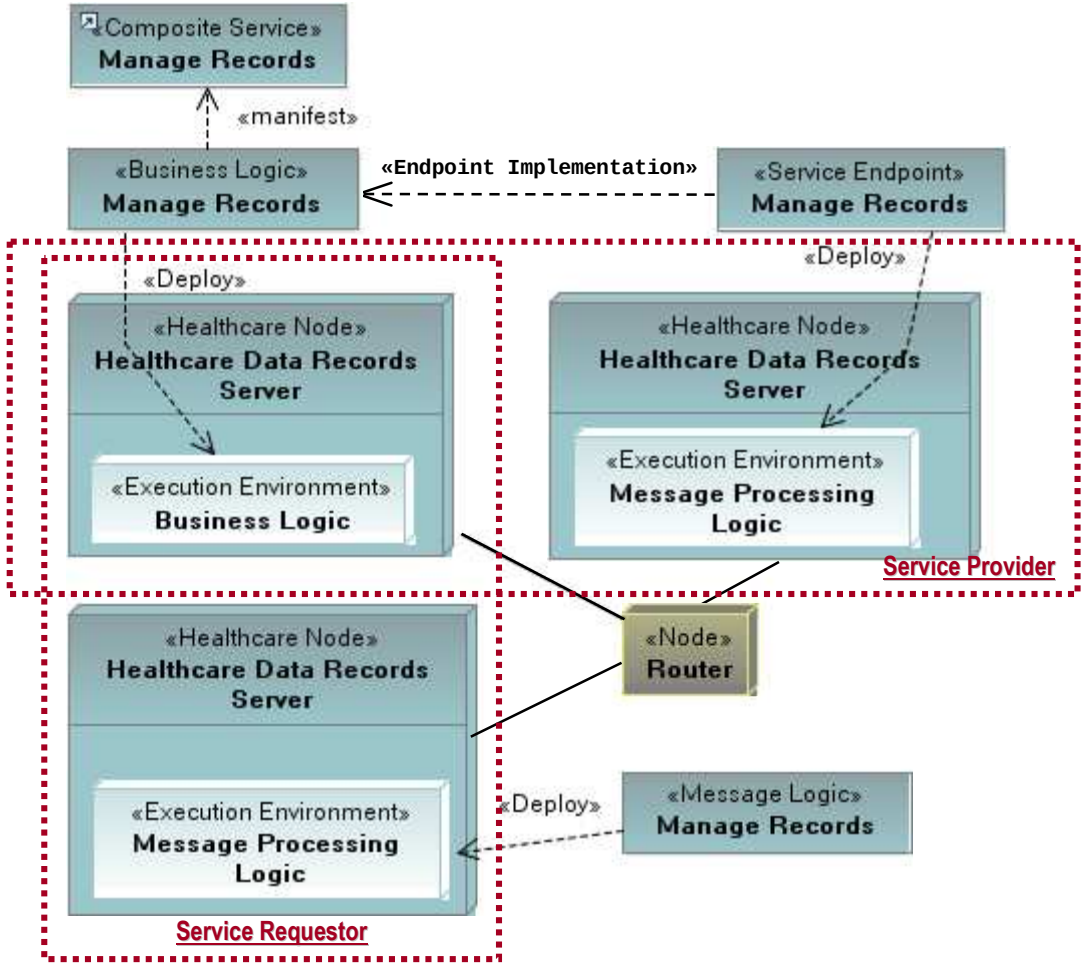
perotsystems®

Service Specification

Based on the *Activity Diagram*



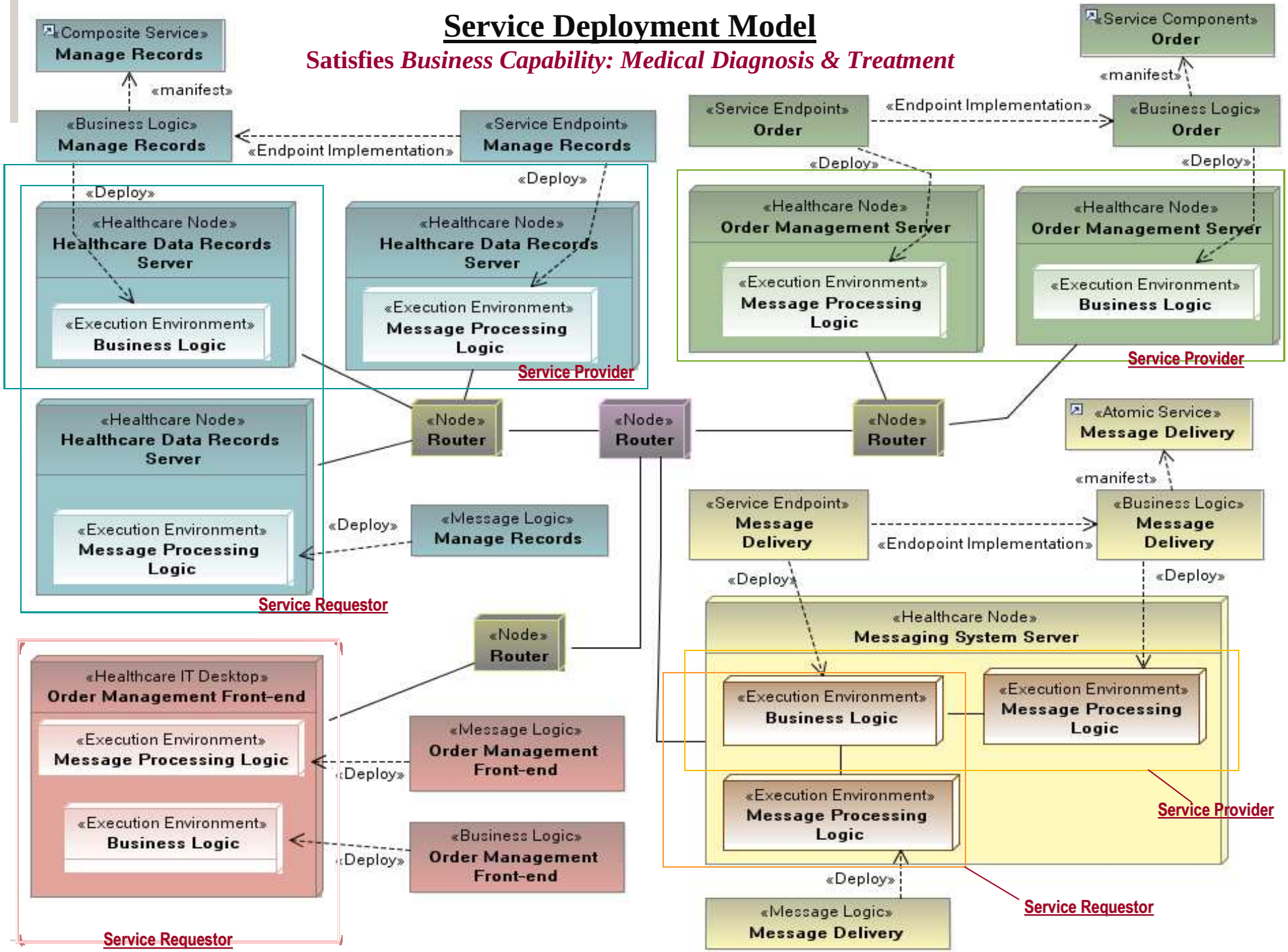
Service Deployment



- **Message Processing Logic**
  - The part of a service and its environment that executes a variety of SOAP messaging tasks.
- **Business Logic**
  - The back-end of a service that performs tasks in response to a receipt of SOAP message contents. It is business-specific with various scopes as defined by WSDL

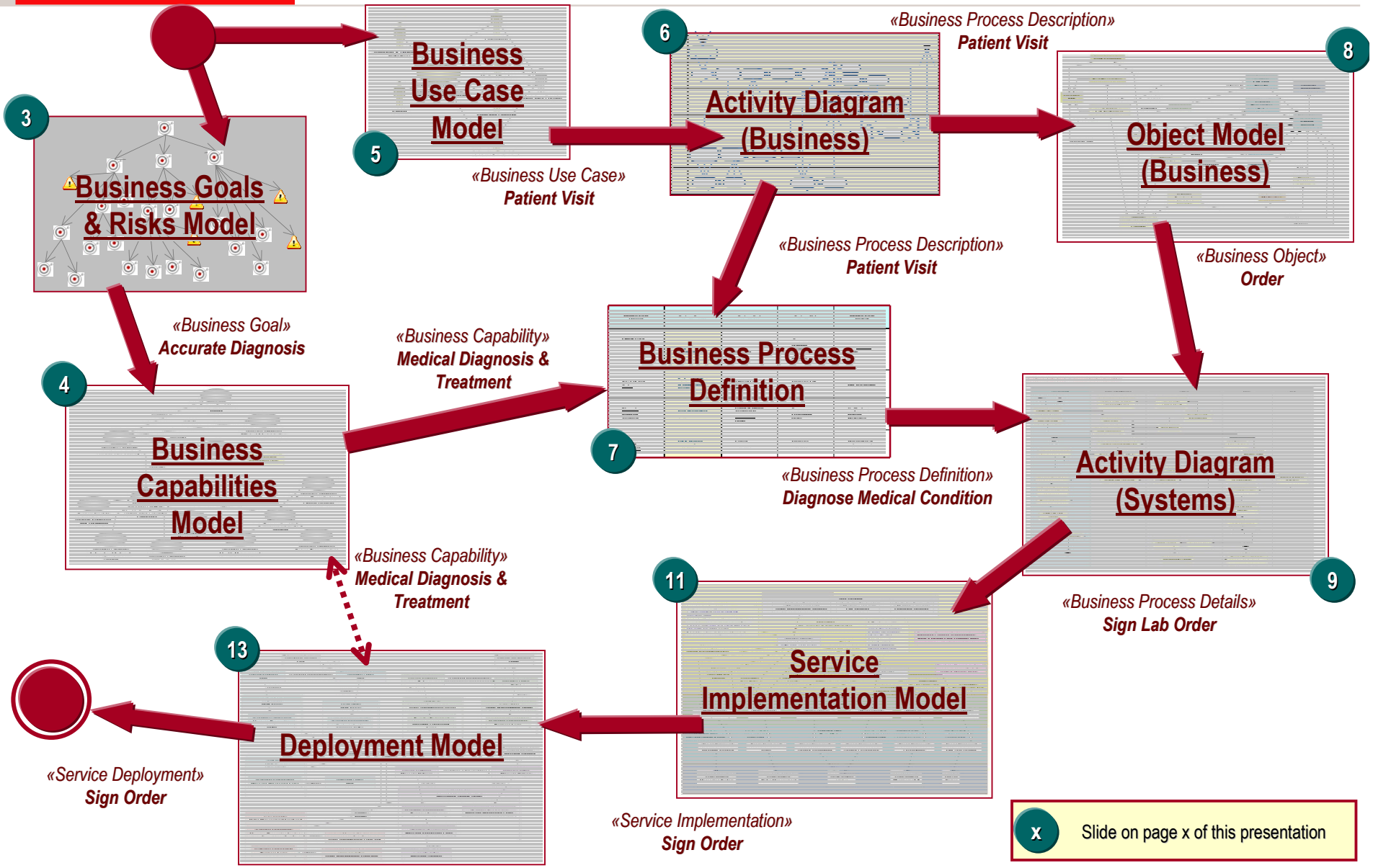
Service Deployment Model

Satisfies *Business Capability: Medical Diagnosis & Treatment*



perotsystems®

Development Procedure Recap



perotsystems®

What is Missing?

- **Elaborate on Business Classes**
  - Add all other UML elements & aspects (Attributes, methods, constrictions, etc.)
- **Elaborate on Service Specification**
  - Why services are layered as shown? What are their contracts' specifications?
- **Elaborate on Service Implementation**
  - How services are provisioned in terms of units of automation
- **Tie ins to SOA Governance: UML expressions of:**
  - Registry and/or repository restrictions
  - Contracts technical specifications
  - Policies enforcement
- **Add, possibly, other UML-based diagrams that would more thoroughly assist developers during implementation**

perotsystems®

References

- **Dept of Veterans Affairs,** ***SOA Framework*****,** various internal documents of SOA Architecture, Monitoring, Management and Governance
- **CBDI Service Oriented Architecture Practice Portal,** ***CBDi Journal,*** various articles on the CBDi Service Architecture & Enterprise (SAE) UML profile (http://www.cbdiforum.com/)
- **Thomas Erl**
  - ***Service-Oriented Architecture (SOA): Concepts, Technology, and Design***, Prentice Hall, 2005
  - ***SOA Principles of Service Design***, Prentice Hall, 2007
  - ***SOA Design Patterns***, Prentice Hall, 2009
- **Dirk Krafzig et al,** ***Enterprise SOA: Service-Oriented Architecture Best Practices*****,** Prentice Hall, 2004
- **IBM,** ***Rational Unified Process, SOA plug-ins*****,** an SDLC Framework expanded to include SOA concepts
- **zapthink,** ***Research Papers,*** various articles on SOA modeling (http://www.zapthink.com/)