

Adeptia Enterprise Architecture Framework

Author: Deepak Singh, CTO
312-229-1727 x113
deepak.singh@adeptia.com

www.adeptia.com



A D E P T I A

Topics



- ◆ Why have Enterprise Architecture Framework?
- ◆ Business drives Enterprise Architecture
- ◆ Functional View
- ◆ Technology View
- ◆ Adeptia: Technology platform for this Framework
- ◆ Example:
 - Framework applied to a sample project (An extranet Portal)
 - How will it ensure success?



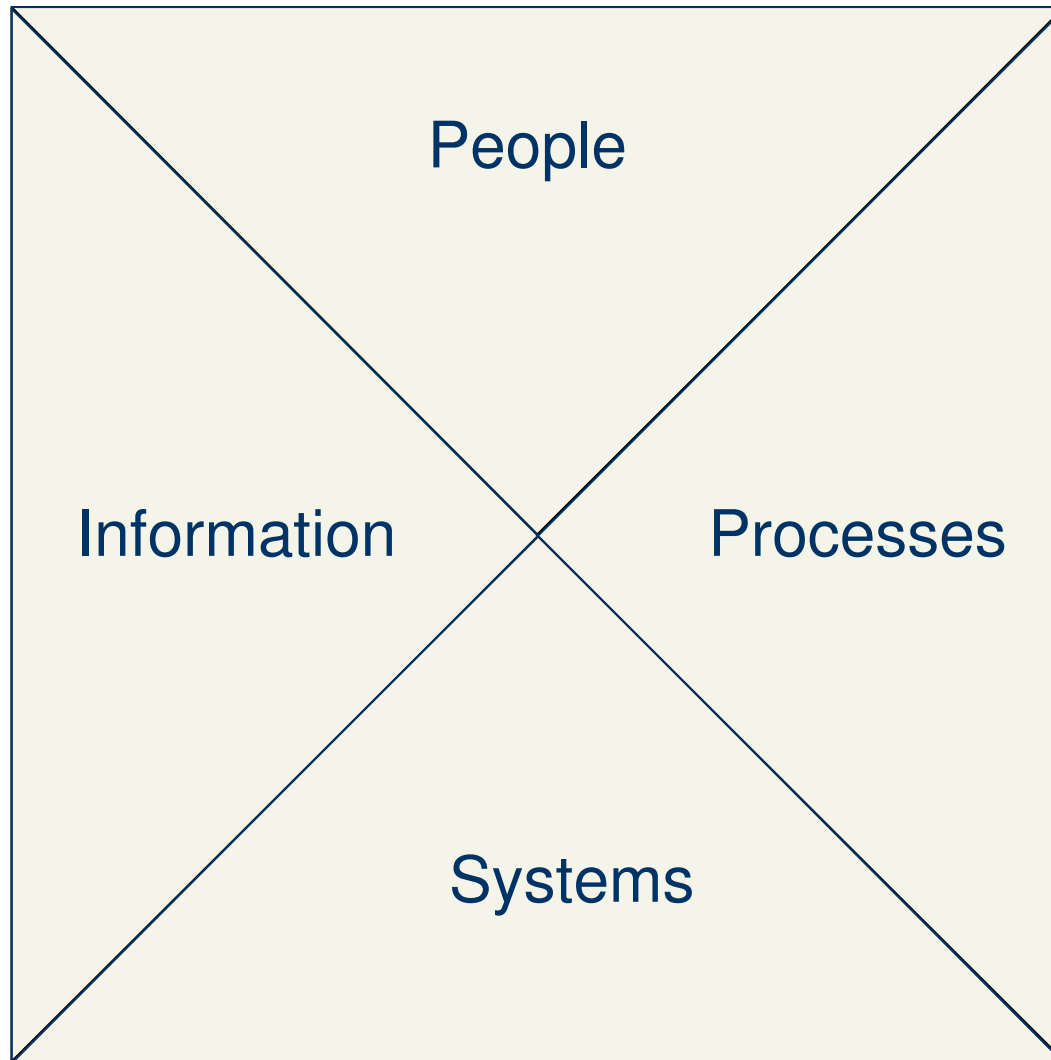
Why have Enterprise Architecture Framework?



- ◆ Aligns business strategy with technology investment
- ◆ Creates Vision and Strategy for Information Technology
- ◆ “Blueprint” on how to view and approach IT projects
 - Presents a Functional and a matching Technology view for every IT initiative
- ◆ Helps put a plan in place on how to get to the vision in an evolutionary, incremental way
- ◆ Adeptia Framework:
 - ➔ Vision for agile, integrated enterprise
 - ➔ Process-centric, SOA-based approach
 - ➔ Provides a structure for consistent IT decision making



Adeptia Enterprise Architecture Framework



◆ Key ELEMENTS of framework:

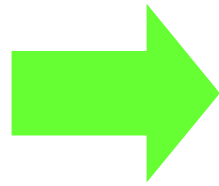
- People
 - Who does the work, for whom
- Processes
 - How work is done
- Systems
 - Which tools are used
- Information
 - Information = Data + Context
 - Inputs and Outputs of work



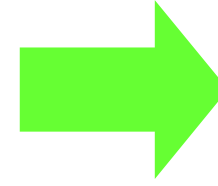
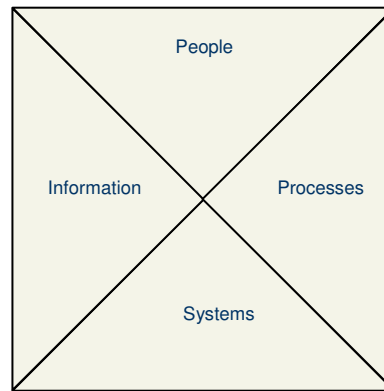
Business Drives Enterprise IT Architecture



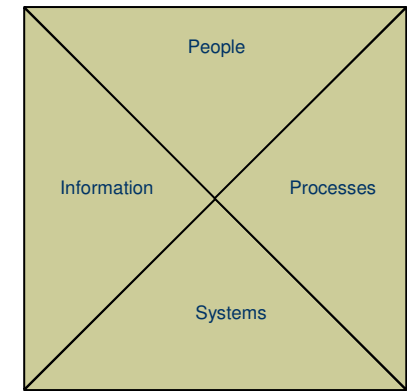
Business Model
+
Business Strategy



Functional View



Technology View



Executives,
Business Managers

CIO



Business Managers

CIO,
IT Management

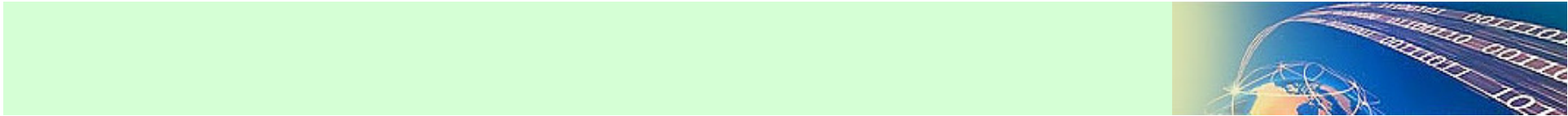
Business Analysts



CIO,
IT Management

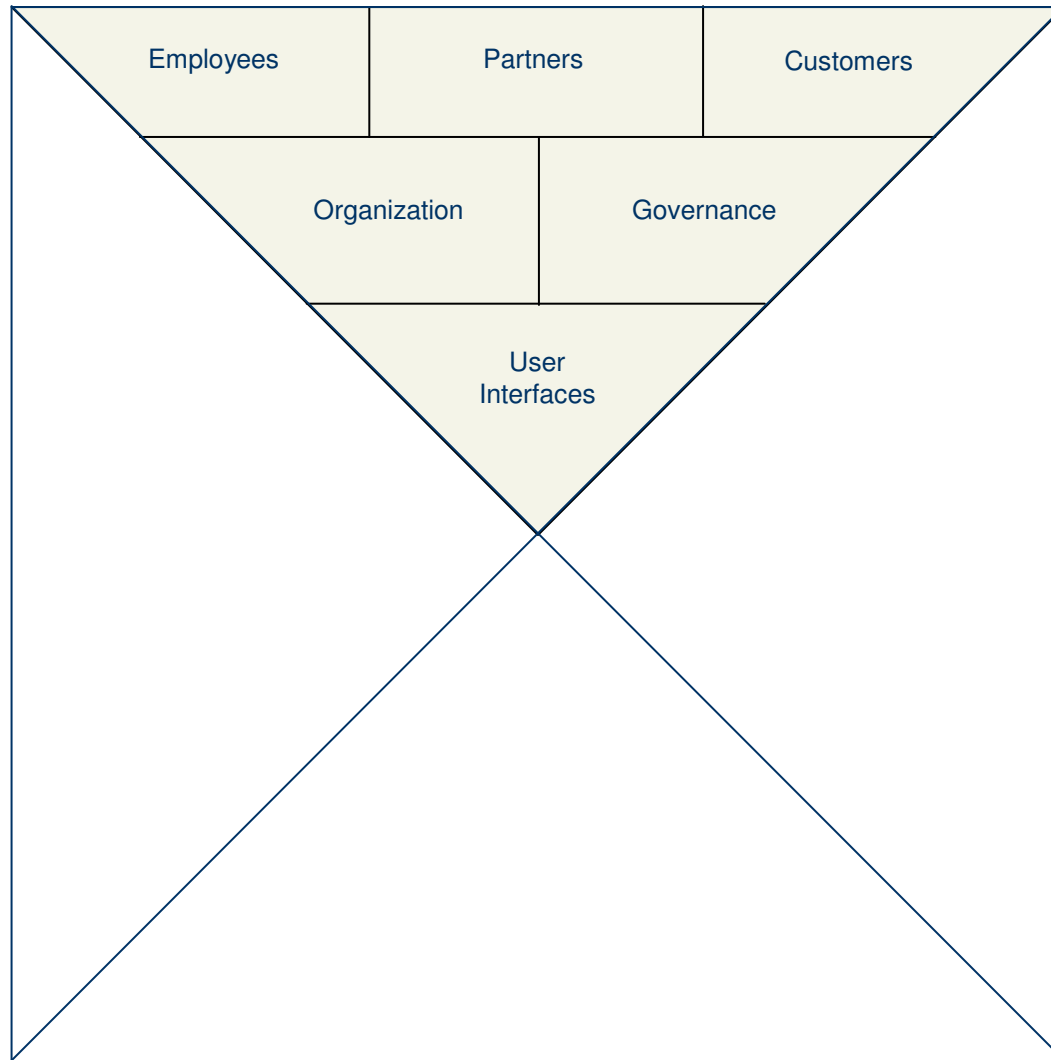
Architects, Tech Leads,
Developers





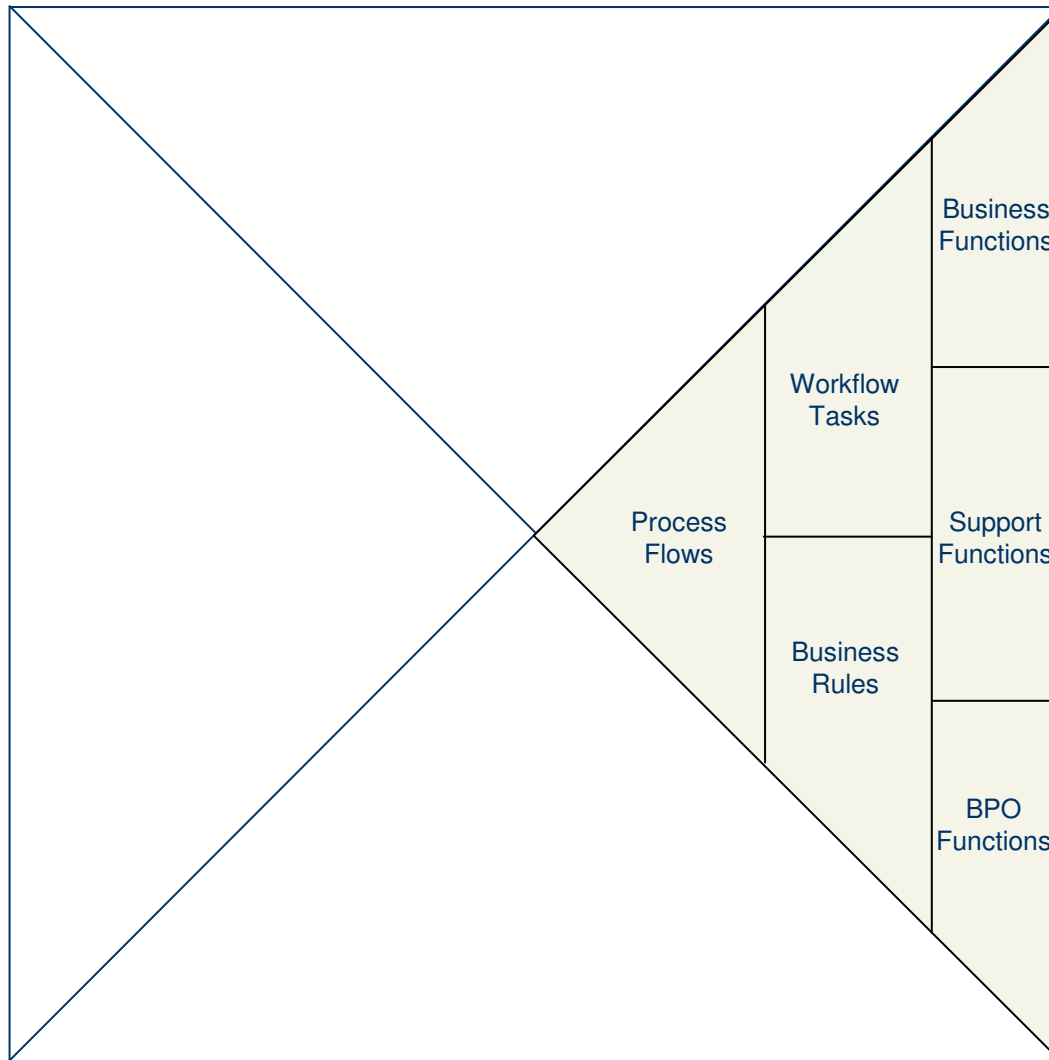
Functional View

Functional view: People



- ◆ **Employees**
 - Business Users
 - IT users
- ◆ **Partners**
 - Suppliers, Vendors, BPO (Outsourcers)
 - Distributors, Agents
 - Regulatory Bodies
- ◆ **Customers**
- ◆ **Organization**
 - Organizational chart
 - Responsibilities, reporting and permissions
- ◆ **Governance**
 - Work and Project tracking
 - Sarbanes-Oxley compliance and controls
 - Business Case, Benefits analysis
- ◆ **User Interfaces**
 - Website / Portal / Forms, Collaboration
 - Decision support / Reports
 - Human-Applications interface

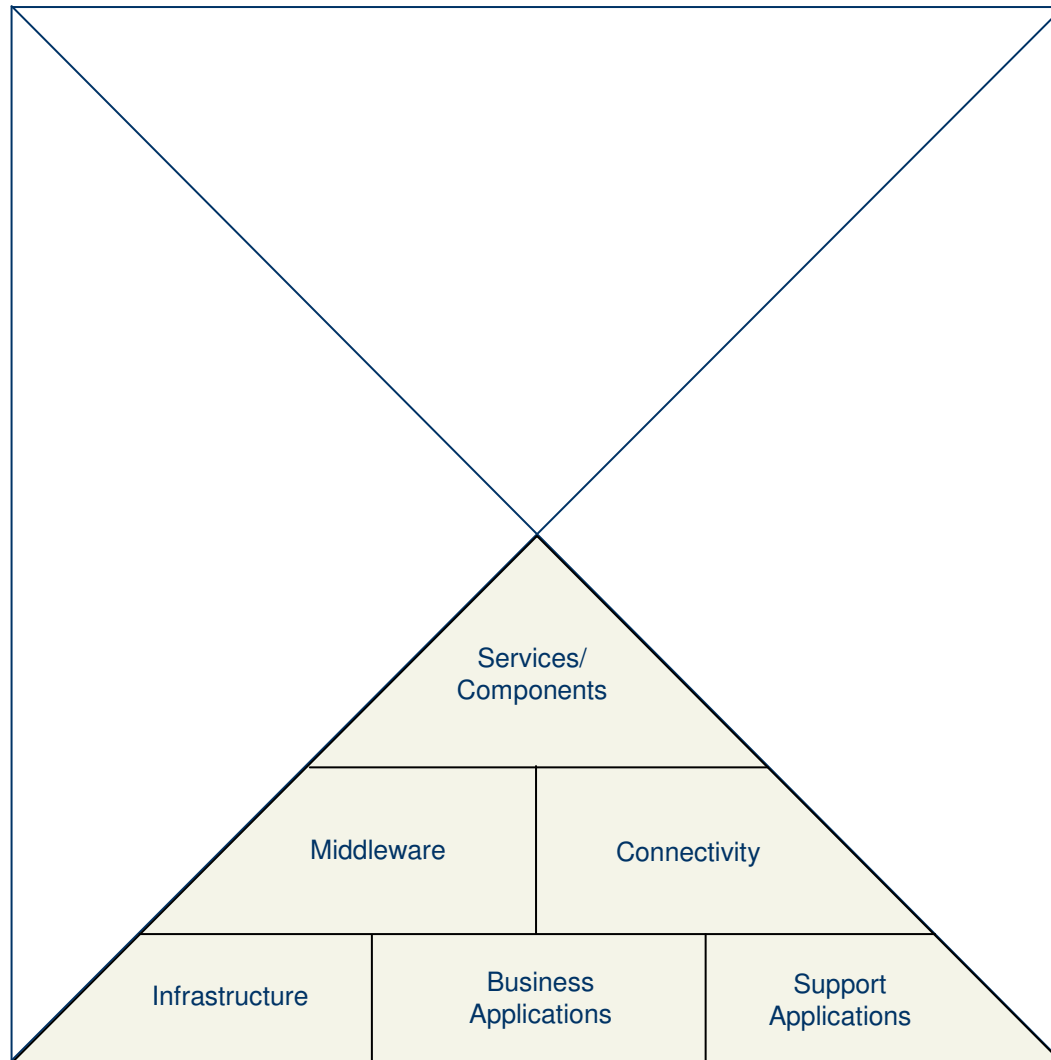
Functional view: Processes



- ◆ **Business Functions**
 - Activities performed for each department
 - Functional depts.
 - Underwriting, Claims, LoBs, etc
- ◆ **Support Functions**
 - Sales & Marketing: Lead Mgmt, Trade shows
 - Accounting and HR: Monthly close, Expense Reimbursements, New Hires
 - Customer Care: Order mgmt, Training, Support
- ◆ **BPO (Outsourced) Functions**
- ◆ **Workflow tasks**
 - Manual tasks required of business users such as Reviews, Decisions, Exception handling etc.
- ◆ **Business Rules**
 - Routing of work to business users
 - Automation of repetitive decisions
- ◆ **Process Flows**
 - HR: New Hire orientation, Leave approval
 - Underwriting: Quote request/response, Agent downloads



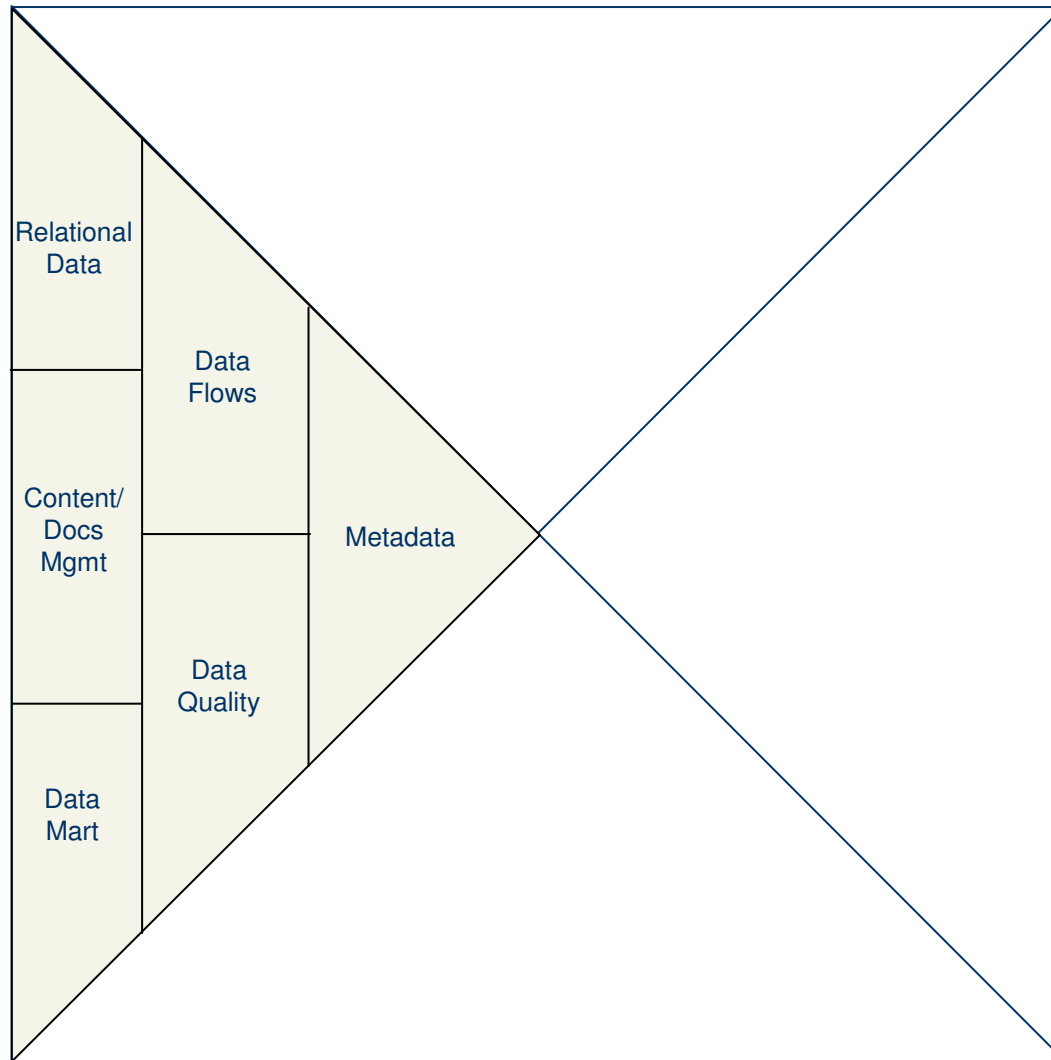
Functional view: Systems



- ◆ **Infrastructure**
 - On premises vs. hosted infrastructure
 - Security requirements and methods
 - Business continuity requirements
- ◆ **Business Applications**
 - Business specific applications: Policy Management Systems, Ratings, Claims apps
- ◆ **Support Applications**
 - Horizontal applications: CRM, Call center, Accounting
- ◆ **Middleware**
 - SOA based integration
- ◆ **Connectivity**
 - Message bus, Application interfaces, Adapters
- ◆ **Services**
 - → Functional components: Discrete IT modules that perform a specific function using underlying applications
 - E.g.: Get customer account number, Generate AL3 download file, Download email attachment

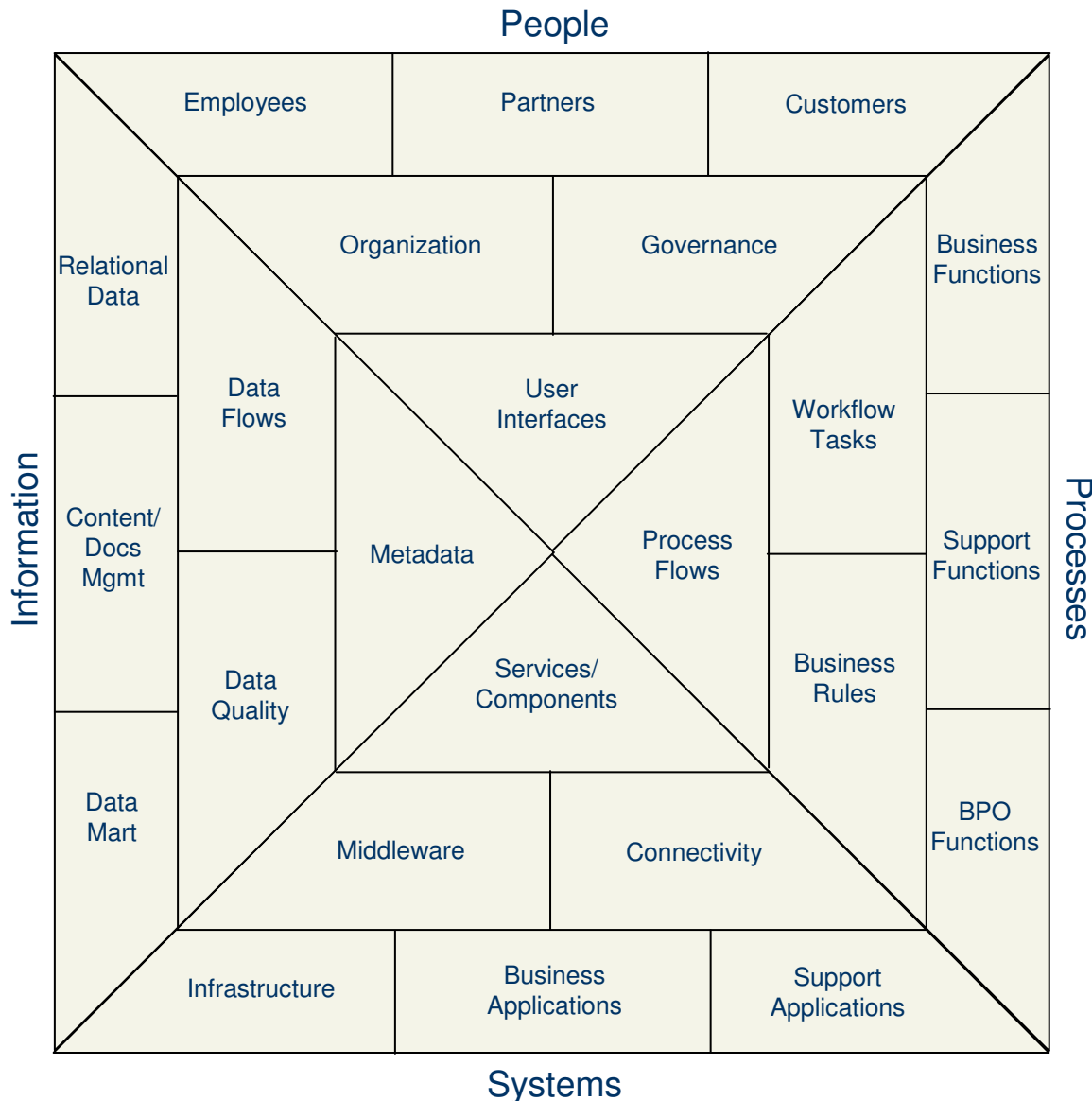


Functional view: Information



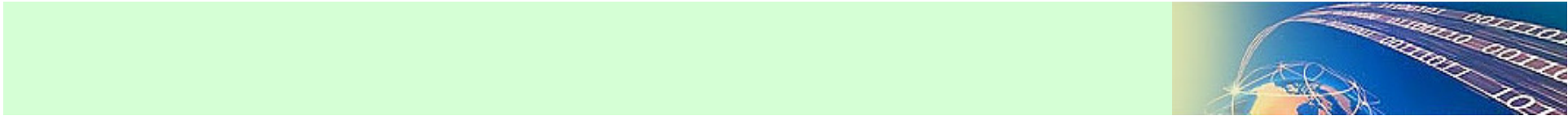
- ◆ **Relational Data**
 - Storage and management of structured data
- ◆ **Content / Document Management**
 - Storage and management of unstructured data
- ◆ **Data Mart**
 - Data warehouse for Business Intelligence, Analytics and Reporting
- ◆ **Data flows**
 - Receive data from where and send where, when
 - Handling of errors and exceptions
 - Sources and Targets
 - Who is generating data, how to get it
 - Who needs data, how to send it
 - Employees, External entities, applications
- ◆ **Data Quality**
 - Insuring accuracy, consistency, timeliness of data
- ◆ **Metadata**
 - Data elements and their meaning
 - Information = Data + Context

Functional View



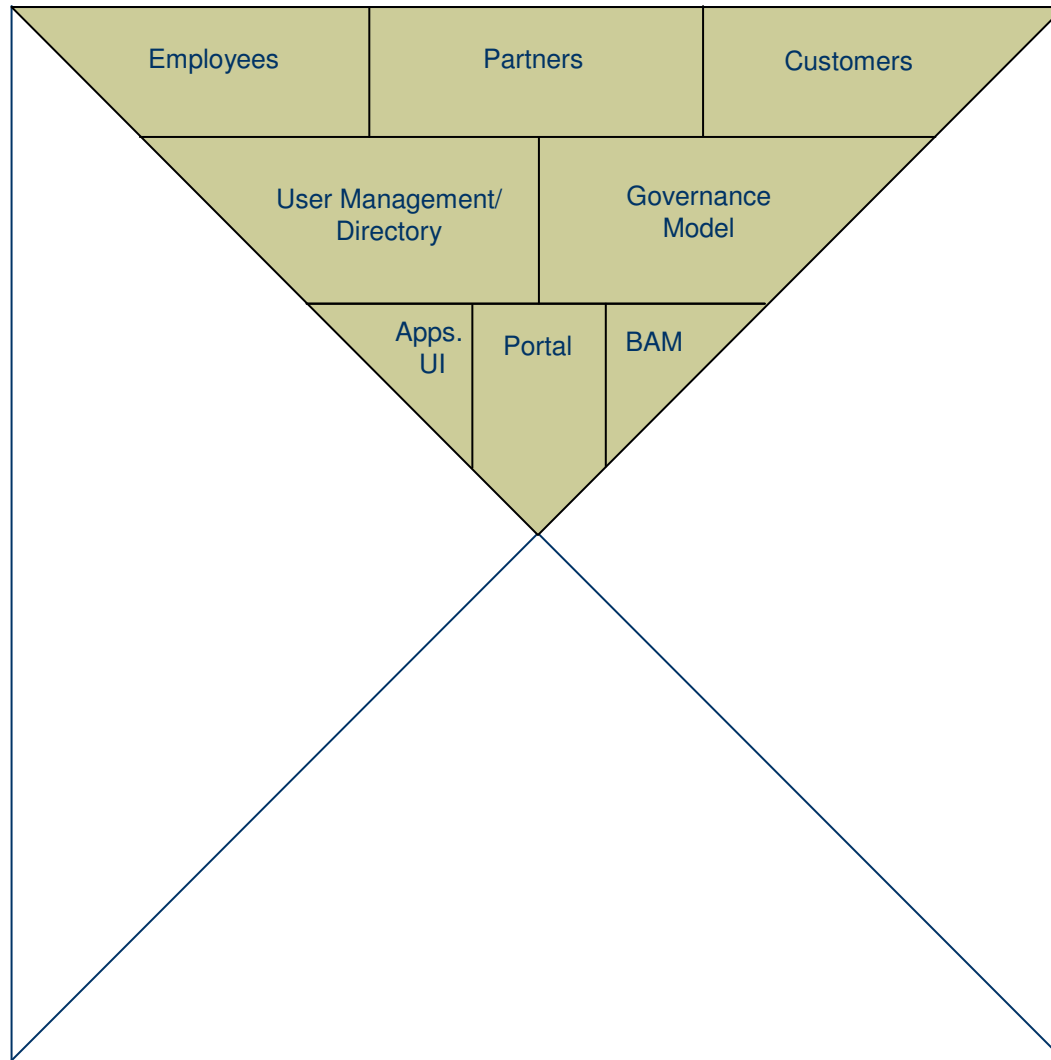
- ◆ Each of the areas in the diagram represent specific capabilities that are required for IT to successfully complete IT projects to meet business goals
- ◆ These combine together to define a set of requirements for any IT project
- ◆ By not explicitly addressing any one of these areas leads to
 - ➔ Gaps in the overall business and technical requirements
 - ➔ Finished projects not fully meeting original business needs
 - ➔ High risk of delays and failure of project





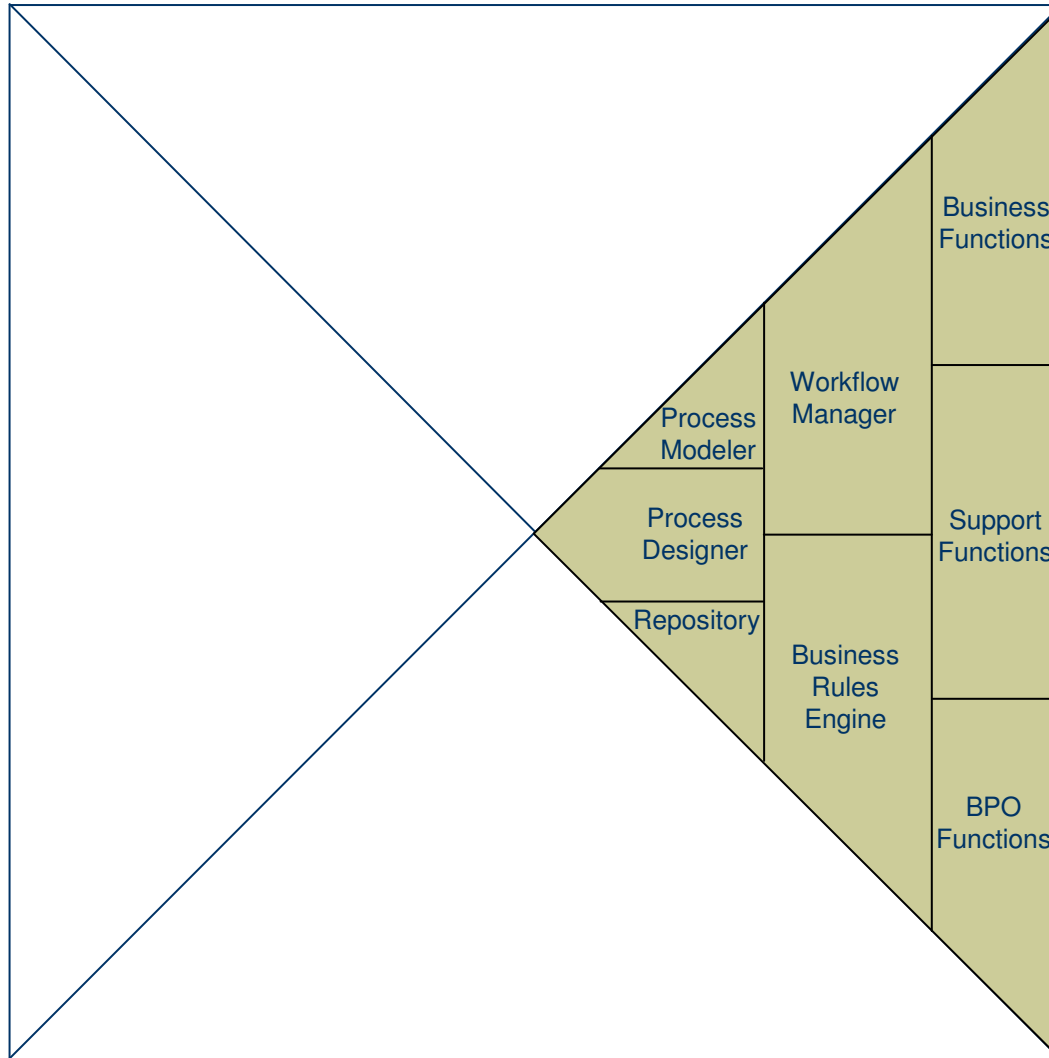
Technology View

Technology view: People



- ◆ **User Management**
 - Directory of users and managers, login parameters, security and permissions
 - CRM app for contact management
 - Examples: LDAP, MS Active Directory, Salesforce.com
- ◆ **Governance Model**
 - Sarbanes-Oxley compliance and controls
 - Performance Management
 - Project Management tools and methodology
 - Tracking of timeline and deliverables
- ◆ **Portal Management**
 - Creating a wizard-driven, Forms-based portal
 - Managing changes and enhancements
- ◆ **Business Activity Monitoring (BAM)**
 - Creation of periodic or on-demand reports
 - Real-time dashboards
- ◆ **Applications interface for users**

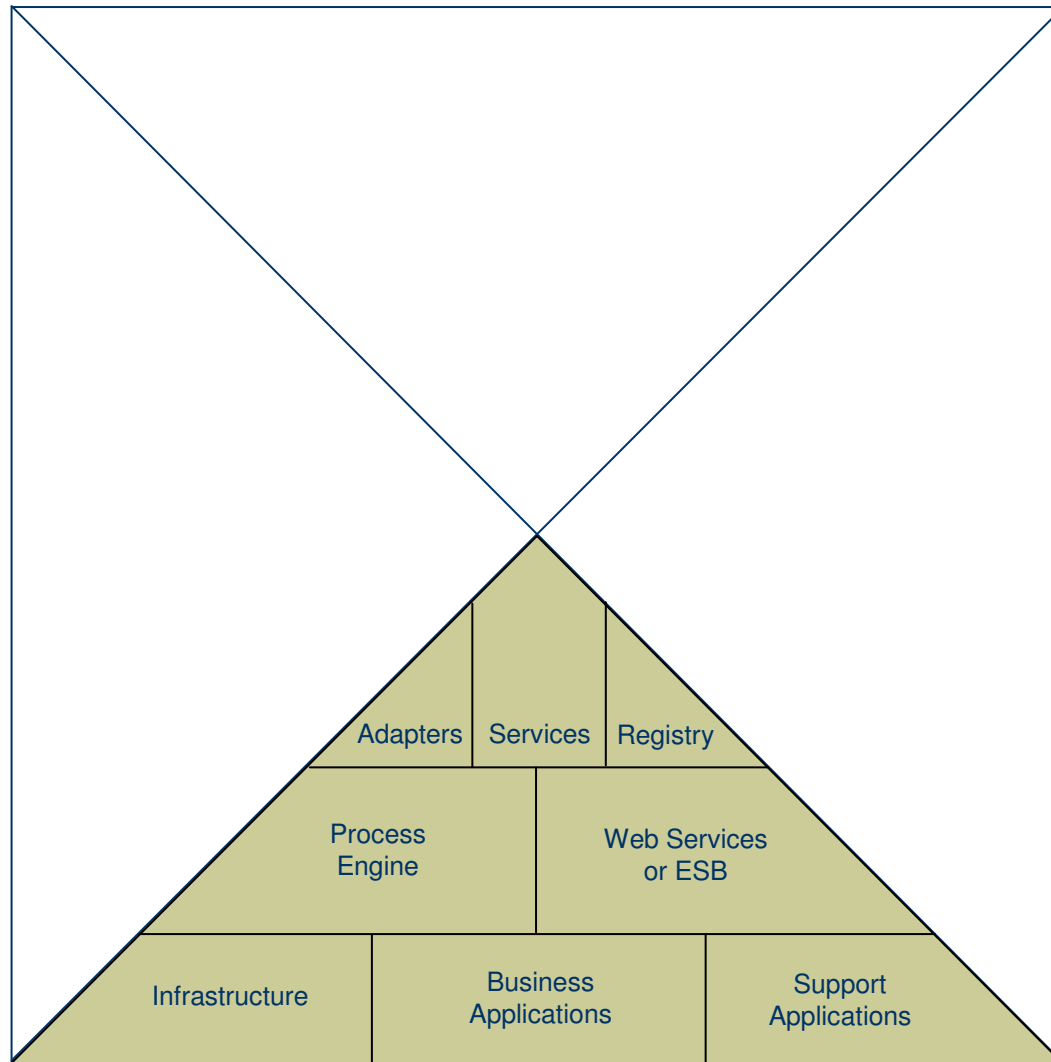
Technology view: Processes



- ◆ **Process Designer**
 - Graphical interface to document processes
- ◆ **Process Modeler**
 - Simulate process parameters and rules to create most efficient and optimal processes
- ◆ **Process Repository**
 - Store and manage processes and max. reuse
- ◆ **Workflow Manager**
 - Create and manage human workflow tasks
- ◆ **Business Rules Engine (BRE)**
 - Specify rules and create rule sets
 - Manage and store rules to maximize reuse



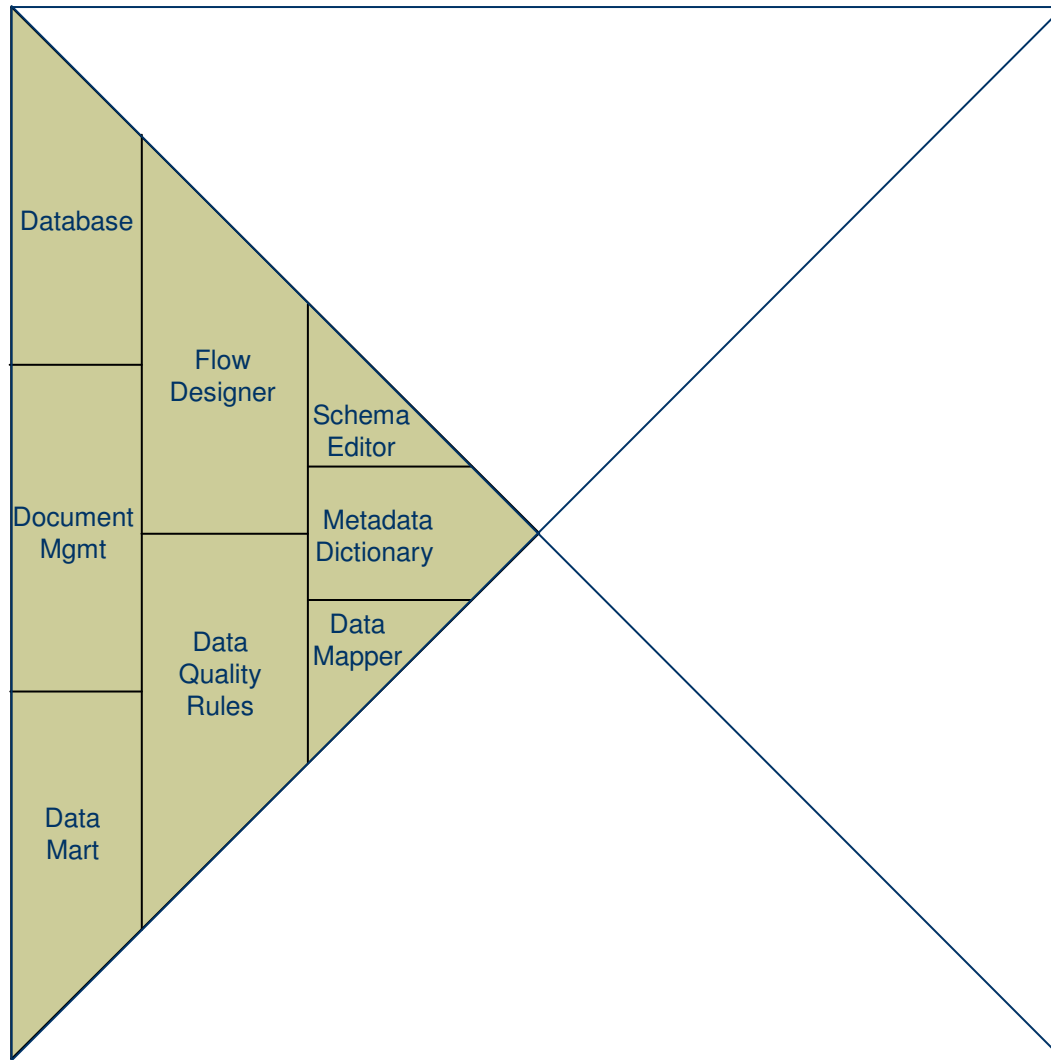
Technology view: Systems



- ◆ **Infrastructure**
 - LAN Network, Internet connectivity, Email
 - Data centers, Hosting locations
 - Security (Firewalls), Backups & Recovery
 - Systems Management
- ◆ **Applications**
 - Legacy, Proprietary systems
 - Packaged applications (CRM, ERP, Accounting)
 - Business specific applications
- ◆ **Web Services or ESB**
- ◆ **Process Engine**
- ◆ **Services Repository**
- ◆ **Services**
 - → Functional components: Discrete IT modules that perform a specific function using underlying applications
 - E.g.: Get customer account number, Generate AL3 download file, Download email attachment



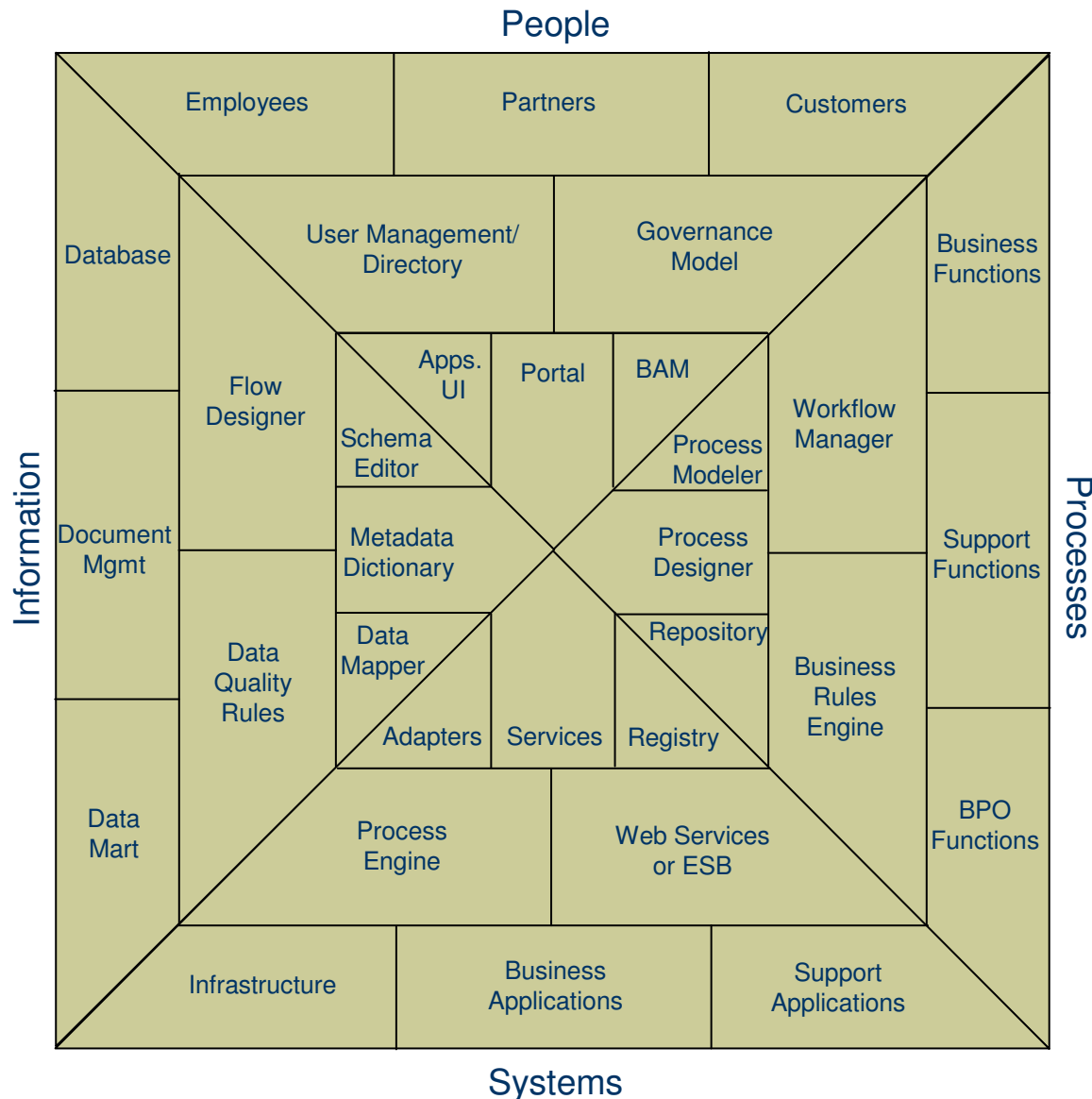
Technology view: Information



- ◆ Databases
- ◆ Document or Content Management (ECM)
- ◆ Data-marts and warehouse
- ◆ Flow Designer
- ◆ Data Quality Rules
- ◆ Metadata Dictionary
- ◆ Schema Editor
- ◆ Data Mapper



Technology View



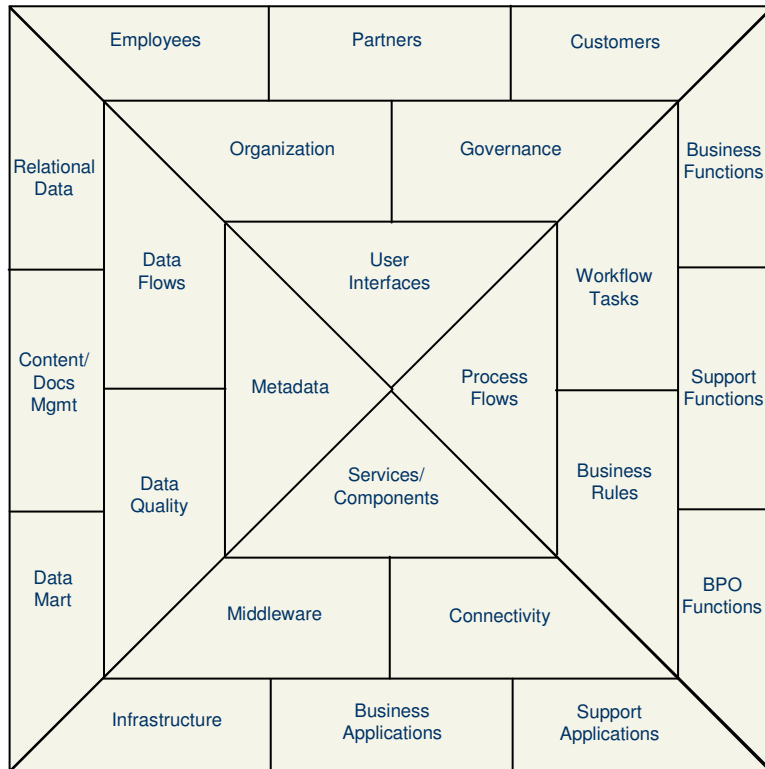
- ◆ This represents specific technology capabilities that are needed to address each of the areas in the diagram
- ◆ These are addressed by:
 - Skill sets
 - Internal IT staff
 - Internal Business Analyst staff
 - Consultants
 - Third-party IT products
 - Internally developed technology
- ◆ By not explicitly addressing any one of these areas leads to
 - ➔ Gaps in the design of IT Projects
 - ➔ Constant delays and high failure rates
 - ➔ Gaps manually addressed in ad-hoc manner
 - ➔ Leads to lot of custom code and manual work-arounds



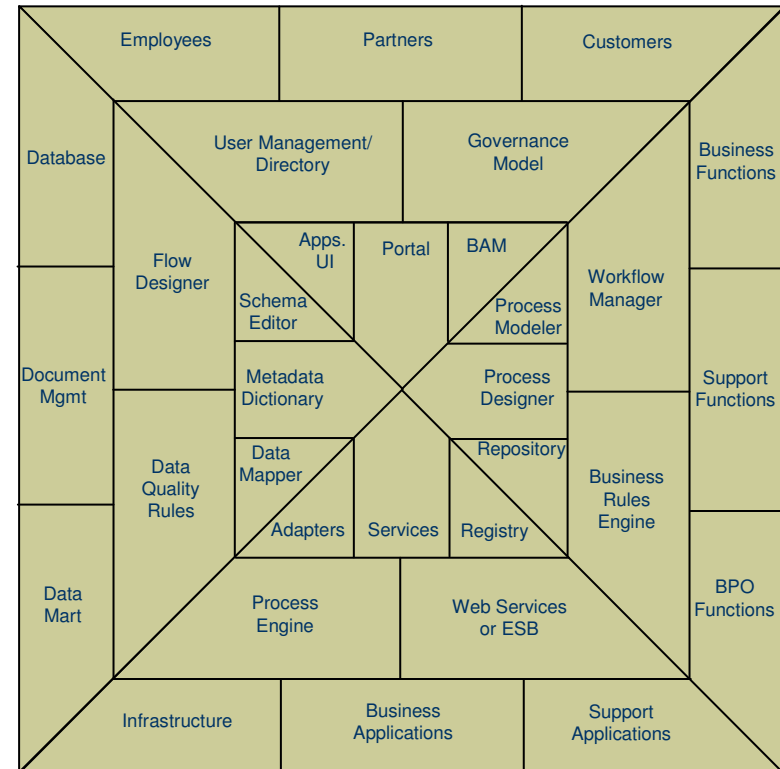
The two views complement each other



Functional View



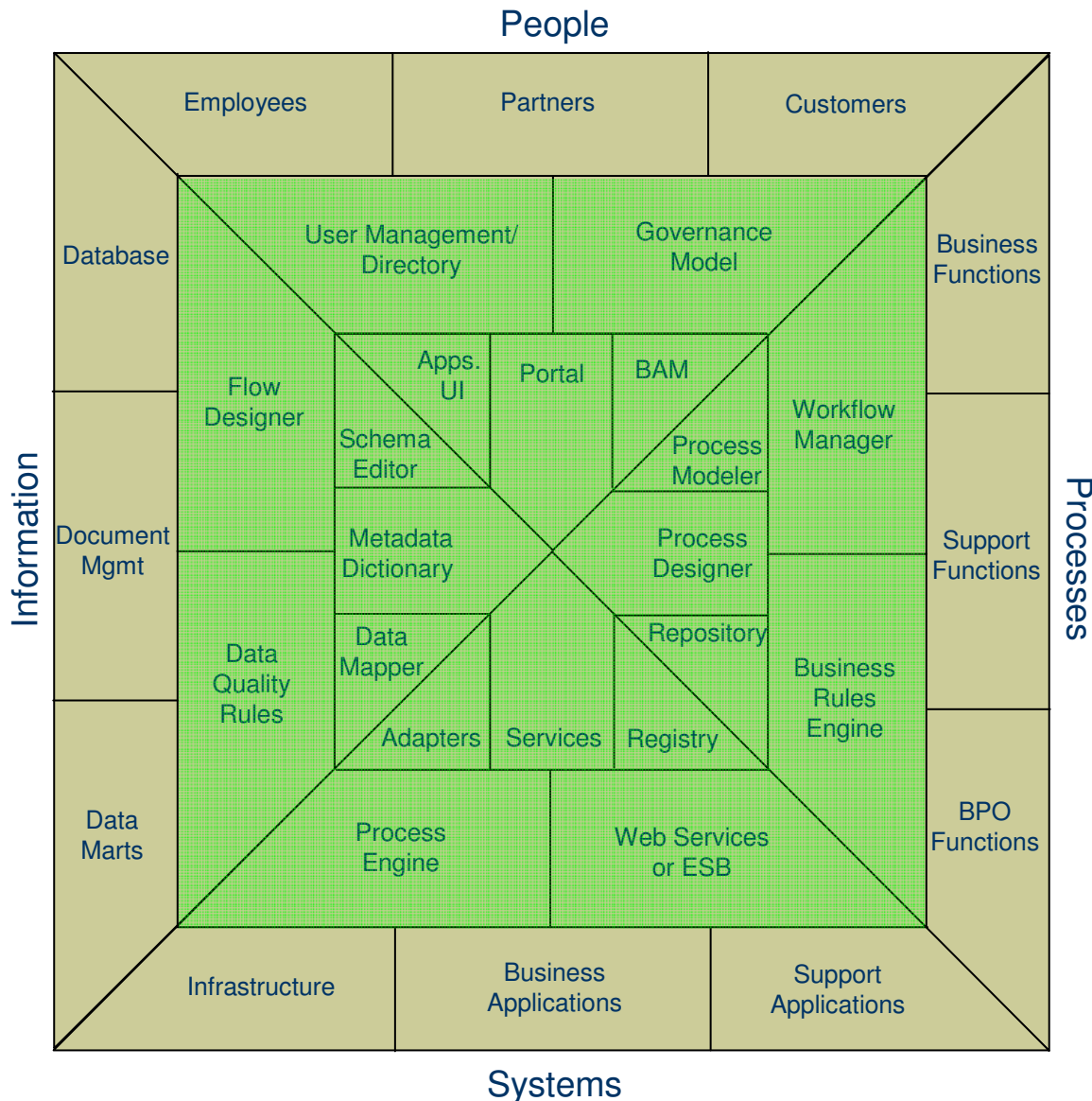
Technology View





How Adeptia supports this framework?

Adeptia: Technology platform for this Framework



- ◆ Adeptia offers technology features that correspond to green highlighted capabilities
- ➔ This makes it easier and faster to apply the common framework on enterprise-wide scale, on all or multiple IT initiatives
- ➔ Investing in Adeptia for any IT project supports an overall strategic vision

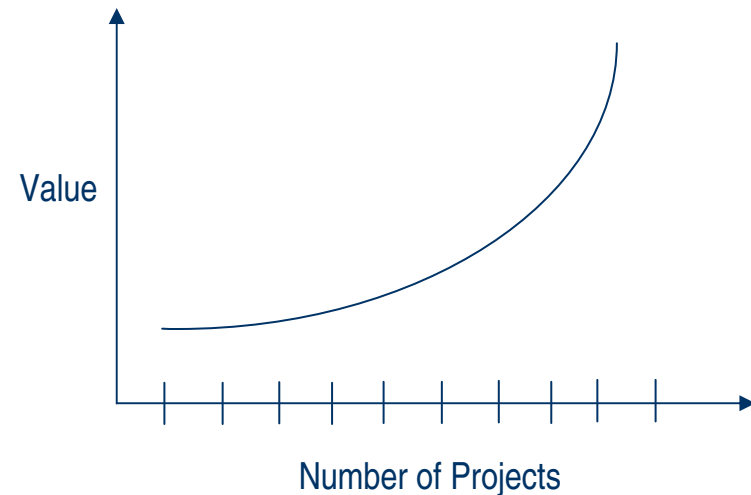


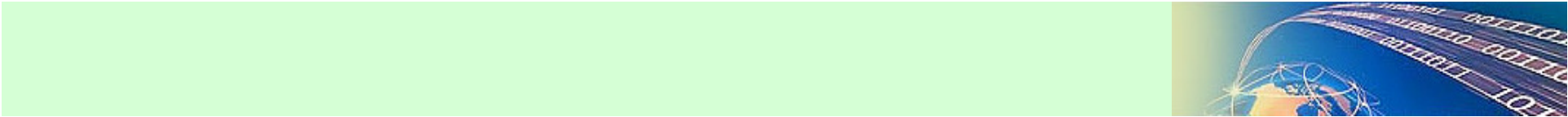
How to implement this framework?

Rolling out the Enterprise IT Architecture



- ◆ Enterprise IT Architecture is a VISION, not a single PROJECT
- ◆ Implementing it is not a one-time effort
 - ➔ Built an IT project at a time
- ◆ Apply the Enterprise Architecture on every new IT project
 - E.g. An Agent or Customer Portal, Data Warehouse, Executive Reporting or Dashboard, Accounting Reconciliation, Automating New Hire Process, Automating Claims or Underwriting
- ◆ Over time, the overall Enterprise Architecture will take shape
- ◆ The value of Enterprise IT Architecture is realized more in later projects





Example: Applying the framework to an actual initiative



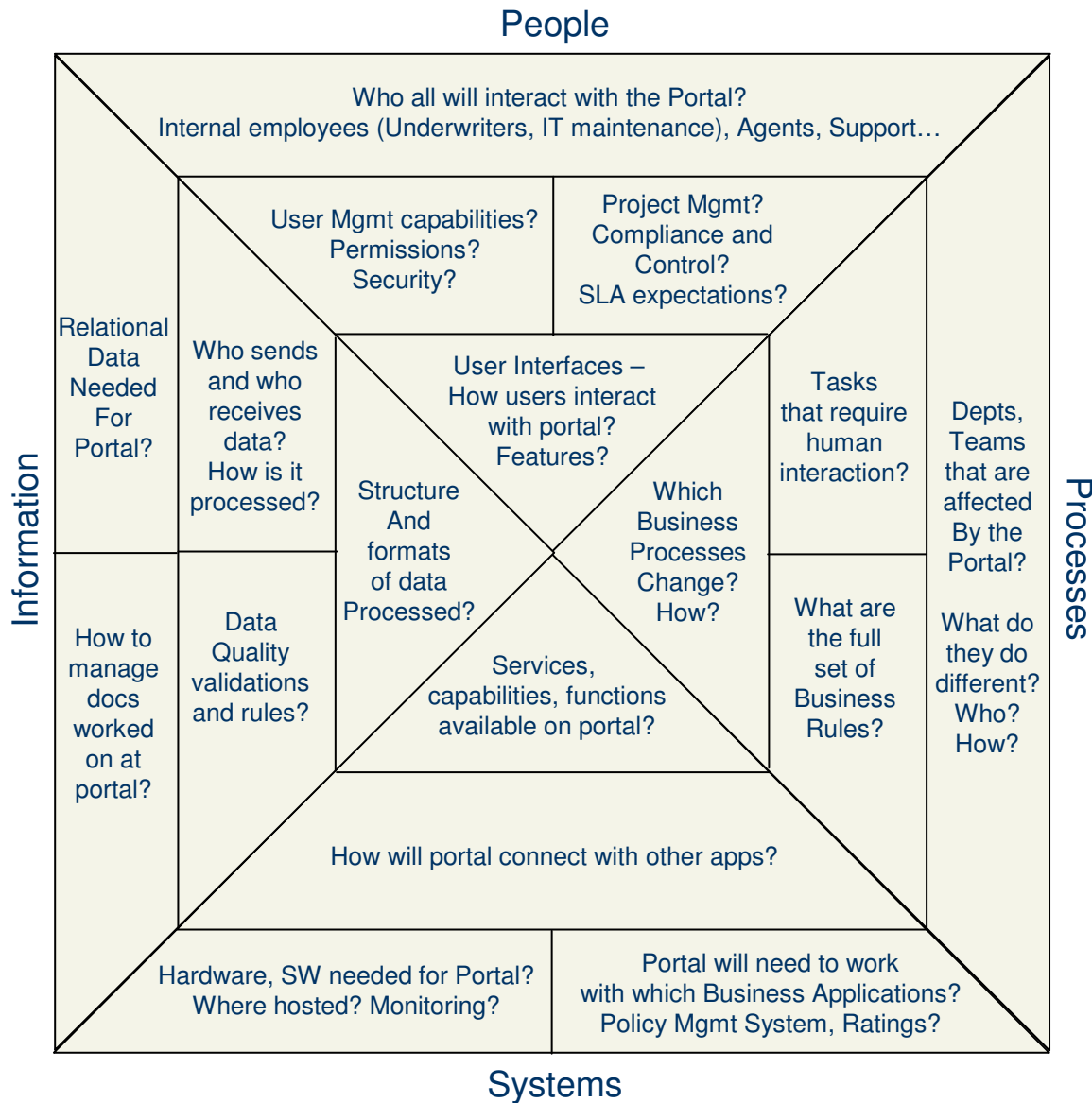
Example



- ◆ Example Initiative: An Extranet Portal
- ◆ Approach:
 - First: Apply the Functional View to the extranet portal project. This is done during the “Requirements” phase of the project.
 - Second: Apply the Technology View to the extranet portal project. This is done during the “Design” phase of the project.
 - Third: Create a Solution Design that is faithful to the Technology View (in terms of tools and SOA services to be used)
 - Fourth: Develop and Implement the project
 - Fifth: Test the solution vs. the Functional View
- ◆ This approach ensures success of the project and commitment to the strategic vision of Enterprise Architecture



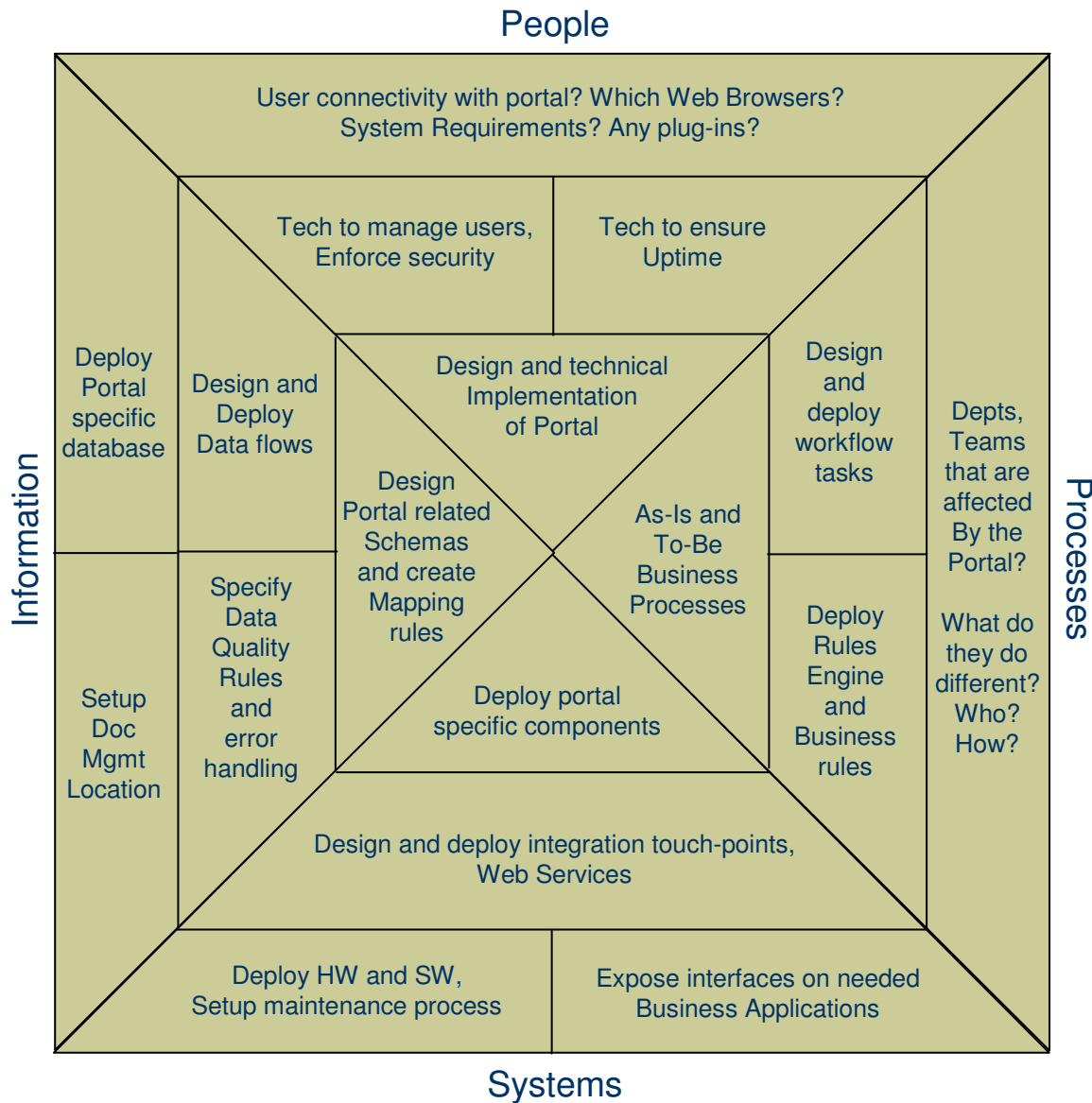
Example: Framework applied to Portal Project



Functional View



Example: Framework applied to Portal Project



Technology View



How does applying the framework ensure success?



Risks without the Framework

- ◆ Incomplete requirements
- ◆ Incomplete technical design
- ◆ End result unlikely to fully meet business goals
- ◆ Last minute crisis, custom-coding and work arounds to address gaps
- ◆ Does not support long-term IT vision
- ◆ Does not promote reuse

Benefits with the Framework

- ◆ Complete requirements and solution design
 - ◆ Project will meet business objectives
 - ◆ Ensures high degree of reuse
 - ◆ Overall strategic vision is supported
- Guarantees success

