

DAMA NJ
August 2006

Putting the “I” Back into IT Architecture
A Data-centric Approach to Enterprise Architecture

Jane Carbone

www.infomajic.com

What is an infomajic?

- **Data & enterprise architecture consulting firm – whose goal is knowledge transfer**
- **Developed experience-based EA methodology**
(IT Architecture Toolkit, Prentice-Hall PTR, 2004)
- **Deliver architecture training, coaching and professional services**
- **Registered NJ Small Business, Certified WBE**



732.580.9878

info@infomajic.com

Agenda

- **Enterprise Architecture (EA) Overview – EA *Toolkit* methodology**
- **Business Framework**
- **Architecture Framework**
 - **Architecture Principles**
 - **Architecture Models**
 - **Architecture Inventory**
 - **Architecture Standards**
- **Framework for Implementation**
- **Architecture & Analysis/Design**

Putting the "I" Back in IT EA

Data-centric Approach

Approach : infomajic *Toolkit* for Enterprise Architecture
(Formerly known as "Toolkit for Enterprise *Information Architecture*")

- **Definition of enterprise architecture:**
 - The set of plans that describes how *all* parts of the IT infrastructure need to behave to support the *enterprise needs* and goals. It includes all the *data* required to run the enterprise and the *functions, technology and people* that create, access, use or transform that data into information and ultimately, knowledge for the business.

- **Purpose of developing target enterprise architecture:**
 - To align the IT infrastructure *with the organization*, in a way that best promotes the organization's goals, while maximizing the benefit of IT dollars spent.

Value of EA

Examples

- **Failures - *Without* EA**
 - **Product conglomerate Data Center Consolidation**
 - **Financial services Data Stewardship Program**
 - **Major European Billing Project**
 - **Major US Billing Project**

Value of EA

Examples

- **Successes - *With* EA**
 - Canadian agency overhaul
 - Financial services – the data architecture that grew
 - From technical architecture to “Integration Architecture”
 - Customer service infrastructure

Value of EA

When...

- Most organizations have a *big* embedded IT infrastructure
- The infrastructure is too big or too tightly coupled or too complex to replace

...Why bother with enterprise architecture?

- It will provide *principles* for future IT decision-making
- It will provide a *roadmap* for how and where replacement infrastructure is desirable
- It will provide *requirements* for purchased applications
- It will provide development *direction* for new projects or undertakings
- It will provide a set of *standards* to guide reuse, consolidation and legacy retirement

Will it require lifetime employment for you to develop a target enterprise architecture?

NO!

EA *Toolkit* Foundation

Critical Success Factors

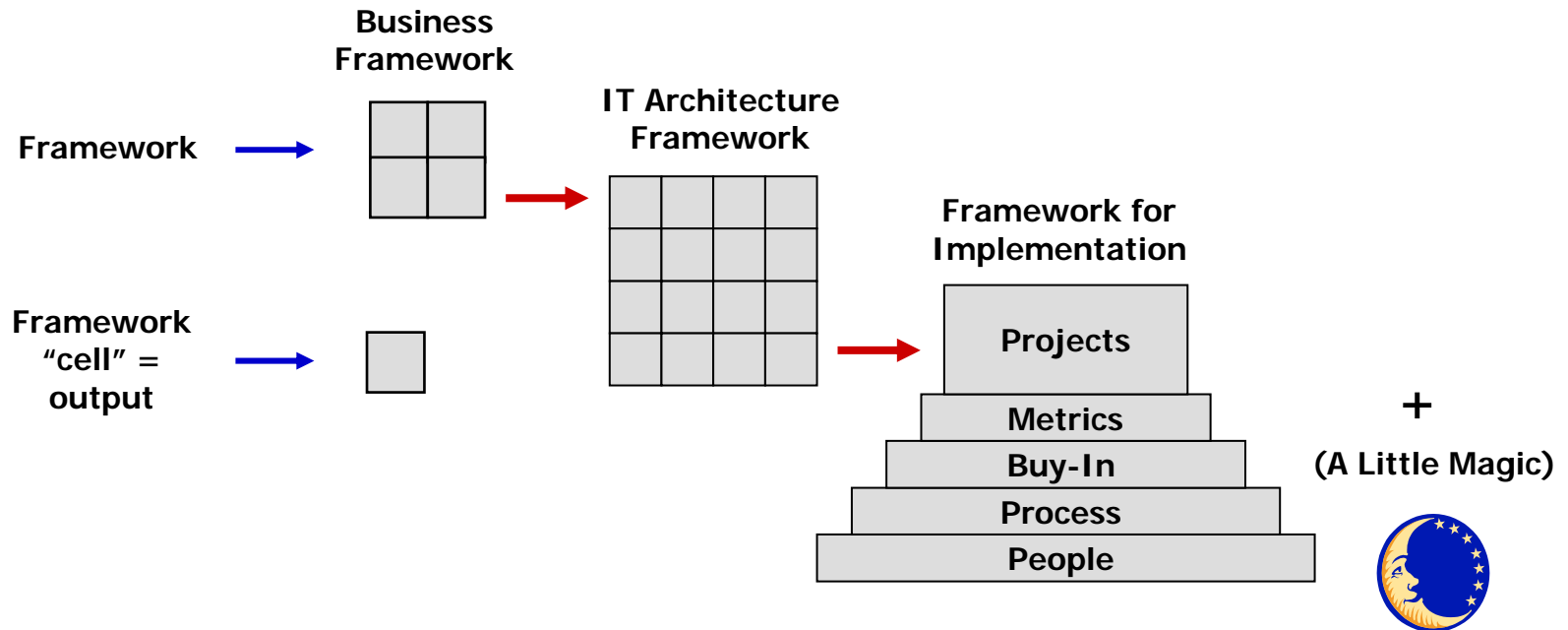
- A clear understanding of where the business is going vs. where it is
- An architecture scope that is realistic
- An action plan that includes the translation of the architecture to a small set of well-scoped, business-oriented projects*
- On-going linkage of the architecture to the business*
- Integrating architecture outputs with each other*
- Processes that support the architects and the deployment of the architecture
- Maintaining the linkage of the architecture with other IT outputs *

* **Audit trail/traceability**

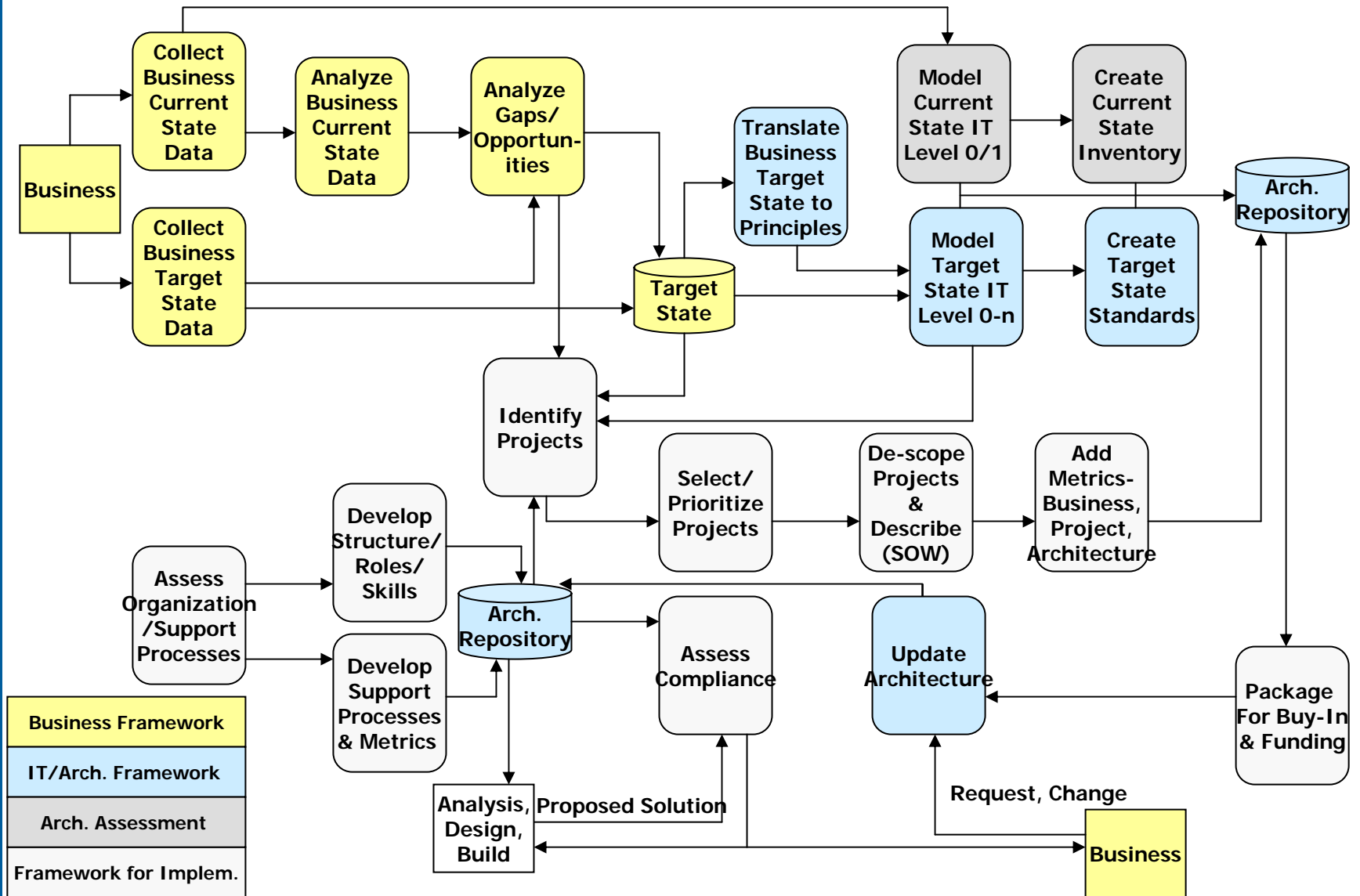
EA Toolkit Overview

We developed the *Toolkit* as a practical response to many architecture "lessons."

Definition: the *Toolkit* is an experience-based, simplified set of three frameworks and the methods for constructing and implementing enterprise architecture *outputs* in business time



Toolkit Process Flow



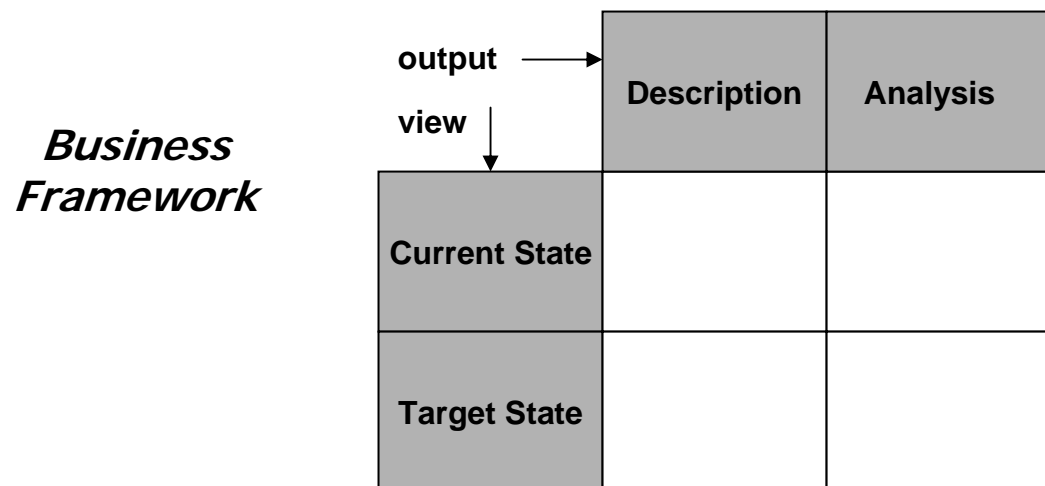
Business Framework

Overview

Definition of Business Framework: Structure used to collect and organize key information about the enterprise that will *drive* architecture

- Rows describe the point in time of the view or “business state”
- Columns describe outputs

Value – Clear and succinct capture and organization of business information for direct input to target architecture; Audit trail from the business needs to architecture content



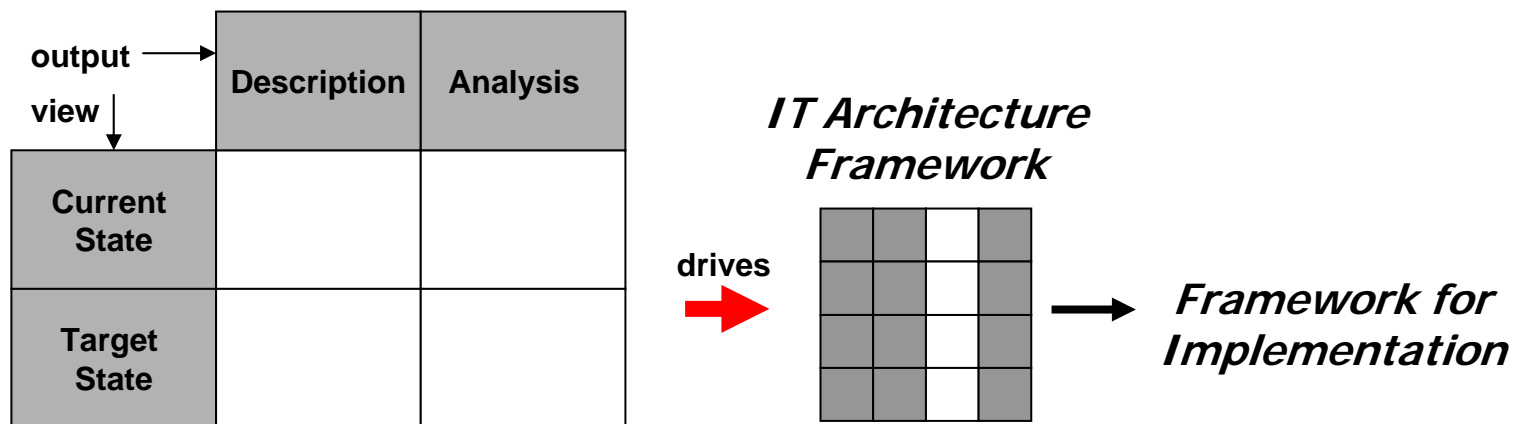
Business Framework *Overview*

Rows or "Business state"

- Current state—the enterprise as it operates today
- Target state—the desired future state of the enterprise

Columns or outputs

- Description - Basic, critical information – "data collection on a diet"
- Analysis - Extraction of key drivers & organization of data



Business Framework

Data Collection

- **Collect the most visible and accepted business strategy documents that capture business purpose, e.g.,**
 - **Mission/Goals/Values Statements**
 - **Strategic Business Plan**
 - **Key Business Initiatives/Major Programs**

E.g., "CDCo Statement of Goals for Y2006

- 1. Learn more about our customers to better market to, retain and satisfy them.**
 - 2. Increase the growth rate (and revenue) of music sales and maintain the growth rate in publications.**
 - 3. Leverage the combined product lines.**
 - 4. Eliminate unnecessary expenses**
 - 5. Become more agile and resilient to change.**
 - 6. Invest in employee growth and satisfaction**
 - 7. Leverage technology to improve business operations"**
- **Interview business leaders**

Business Framework

Data Collection

The goals of document collection and interviews are to identify:

- Key constraints, e.g., Regulatory, Shareholder considerations
 - Major forces that impact the organization, e.g., competition, market forces
 - Significant “culture” issues or changes in business direction, e.g., Mergers, Consolidations, Process Re-engineering
- Synthesize the data you have collected, e.g., put together a brief portrait of the organization—“Key Facts—5 Ws” – E.g.,

What: *CDCo*, Inc. is a publicly held, stable retail music chain that sells CDs, DVDs and still maintains an inventory of records, tapes, posters and sheet music for collectors. *CDCo* sold almost 30 Million items in 2004, with revenue of close to \$1.5 Billion

When: In December of 2005, the board of *CDCo* voted to expand into the book/magazine product line and acquired Bookseller, Inc. Bookseller sales have increased dramatically via user-friendly web site.

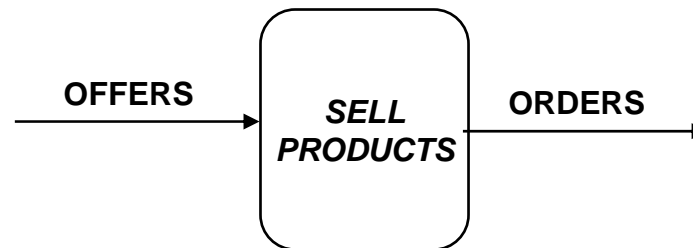
Business Framework

Follow the Data...

In the Beginning...there were *data and functions...*

Process flows - very simple diagrams used as description/analysis tool to capture major business processes and the data on which they operate.

Representations from *Gane & Sarson*, Data Flow Diagrams for Data Flow and Function:



Tool to identify *business process* problems – leads to capture of business gaps/opportunities and critical information

Business Framework

Putting It Together - Example

- **BF outputs (Gaps & Opportunities)** are used as direct input to the creation of target architecture outputs

	Description	Analysis
Current State	<p>Describe the current state of the business— E.g., “5 W’s,” Key Facts</p> <p>“ (CDCo)...has problems accepting on-line payments.”</p>	<p>Analyze the current state E.g., Use Assessment Indicators (Risks, Strengths, Growth/expense reduction, etc.)</p> <p>“Customers unhappy with on-line pay”</p>
Desired (Target) State	<p>Describe the target state of the business—E.g., Reverse responses to Assessment Indicators</p> <p>“Customers are happy with on-line pay”</p>	<p>Analyze & Compare= Gaps/Opportunities in Current to Target State</p> <p>“Repair on-line payment.”</p>

•**Gap:** the problem or difference between the current and desired states of the business

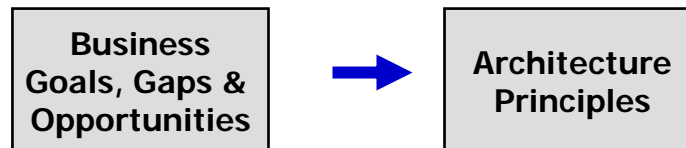
•**Opportunity:** positive statement of action to resolve gap

Architecture Framework

Business Framework Segue

Translating Business Framework *Outputs* into Architecture *Inputs*

- *If Architecture* = The set of plans that describes how all parts of the IT infrastructure need to behave to *support the enterprise needs* and goals
- *And its purpose is "To align the IT infrastructure with the organization, in a way that best promotes the organization's goals, while maximizing the benefit of IT dollars spent"*
- *And we synthesized the business data collection* into a set of architecture drivers (e.g. Business Opportunities)
- *Then* the architecture process begins with the translation of business drivers into IT Principles that support the business




Architecture Framework


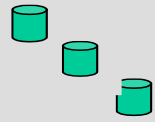
Overview

Architecture Framework: structure used to define IT infrastructure plan content (subjects) and output inter-relationships, driven by the business

IT "architecture" consists of all of the outputs for each of the subjects

Business Framework Gaps & Opportunities



	Principles	Models	Inventory	Standards
Data	"Data is a corporate asset."			"A Customer describes..."
Function				
Platform				
People				

Architecture Framework

Overview

The framework rows describe primary infrastructure components:

- **Data: Key facts about the enterprise**
- **Functions: Key operations necessary to run the enterprise**
- **Platform: Key technologies used to enable the data, functions & people**
- **People : Consumers/providers/users of the Data**

The framework columns describe the architecture outputs for each component:

- **Principles**
- **Models**
- **Inventory**
- **Standards**

Architecture Principles

Overview

Definition of Architecture Principles: The abstraction of the business target state to form guidelines for IT decision-making.

Overview of Guidelines

- 1. Translate a business target statement, gap or opportunity to a statement of intent for IT. E.g.,**
 - Goal—"1. Learn more about our customers to better market to, retain and satisfy them."
 - Principle—"Our (customer) information is a core asset"
- 2. To make sure IT decisions can be tested against principles, make a positive assertive statement. E.g.,**
 - "We will identify and share core data."
- 3. Keep it short and sweet (i.e., no more than a sentence each.) A set of principles should be able to be contained in one page.**
- 4. The set of principles ought to include all the "rows" in the framework, e.g., Data Principle**
 - "Data is owned by the corporation and not by any individual person or group"

Architecture Principles

Example

CDCo IT Principles

1. Information about our core business must be complete, accurate, timely and secure.
2. Employee training is a non-negotiable investment.
3. Functions will be flexible and reusable.
4. Technology will be selected based on architecture fit, capability and vendor support.
5. Core information will be easily available to all departments.
6. IT investments will require business case analyses.
7. When solving new problems, we will leverage existing infrastructure.
8. We will buy before build.

Architecture Models

Overview

Architecture Model Definition: Graphically represents the business view of Data, Functions, Technology, People and the relationships and/or connections between them. An architecture model is a key tool for communicating direction both with business and IT organizations.

Fundamental guidelines* for architecture modeling -

1. Select and define standard “parts”
2. Use a standard set of representations
3. Set a scope
4. Determine level of detail
5. Define state
6. Define environment

* Detailed guidelines describe how to translate business gaps/opportunities to architecture outputs

Architecture Models

Fundamental Guidelines

1. **Select and define a set of standard architecture parts - the *types* of IT infrastructure parts to be addressed in your architecture models.**

Example parts and definitions we use:

- **Architecture service: Communication requirement between two or more architecture components, e.g., translation, messaging, transport**
- **Business function: Key operation necessary to run the business, minimally a verb and a noun, e.g., "Target Customers"**
- **Datastore: Key facts about the business stored for operational use**
- **Reference Data: Small set of relatively static data shared across many business functions, often consisting of codes (e.g., account codes), abbreviations (e.g., state codes), "type" or "model" data (e.g., product/service type)**

Architecture Models

Fundamental Guidelines

2. Use a standard set of icons to represent standard parts. Standardize any notations intended to convey additional meaning.

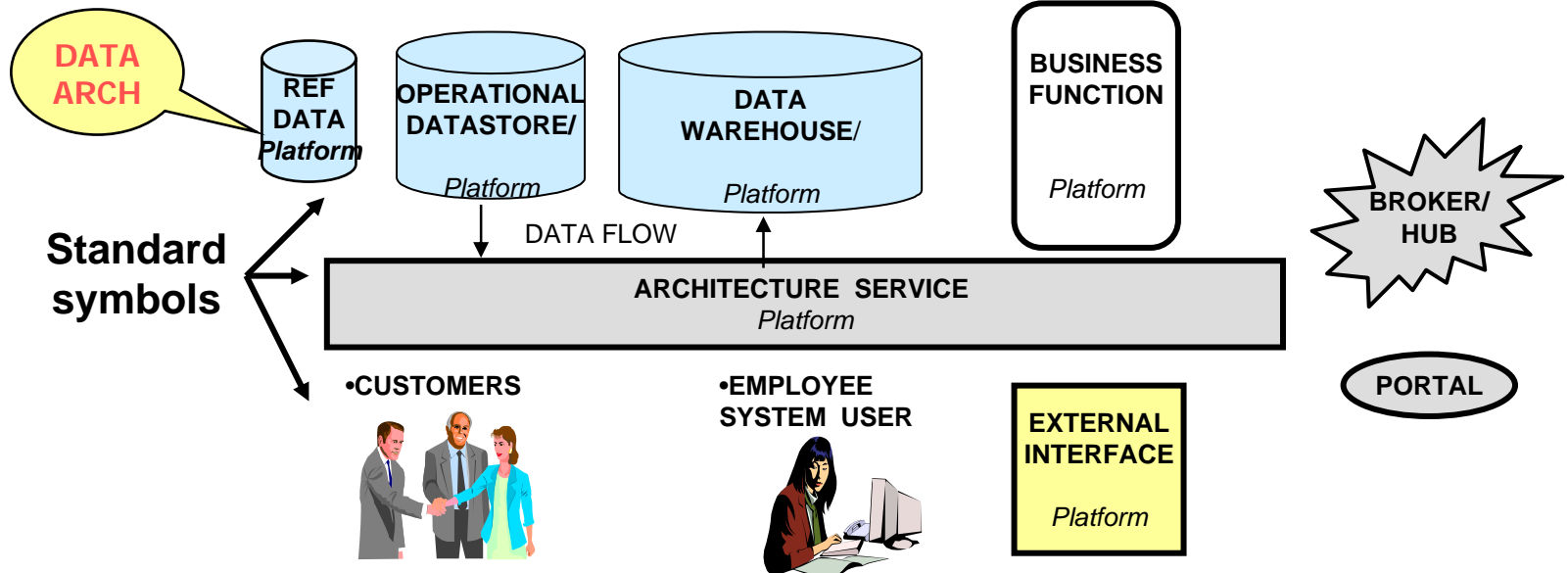
Toolkit standard architecture representation

Standard notations →

EXISTING or
MECHANIZED

PLANNED or
MANUAL

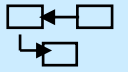

DUPLICATE



Architecture Models

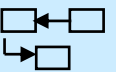



Fundamental Guidelines

3. Set a model scope - what content is combined, included in, excluded from an individual architecture model, e.g.,

	Principles	Models	Inventory	Standards
Data	"Data is a corporate asset."			"A Customer describes..."
Function				
Platform				
People/Process				

...The content of a cell

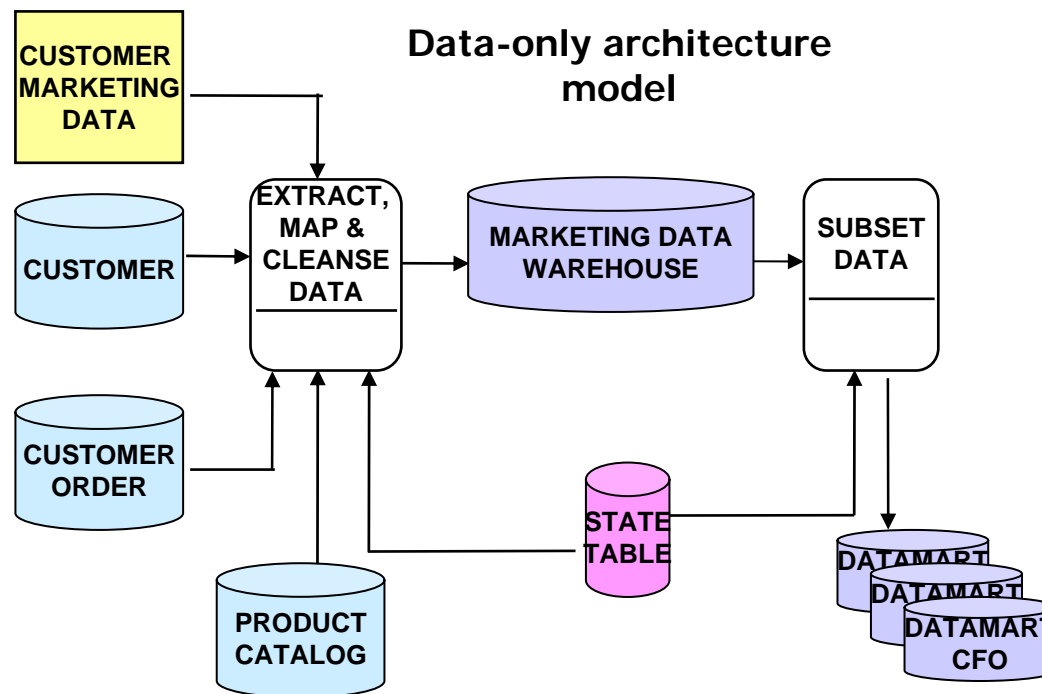
...The content of a column

	Principles	Models	Inventory	Standards
Data	"Data is a corporate asset."			"A Customer describes..."
Function				
Platform		MSx		
People/Process				

Architecture Models

Example

- **Scope: We recommend integrated models—where data, function, technology & people are included in *each* model***
 - Supports broadest capture of relationships across components & provides a holistic view for decomposition
 - Vs.



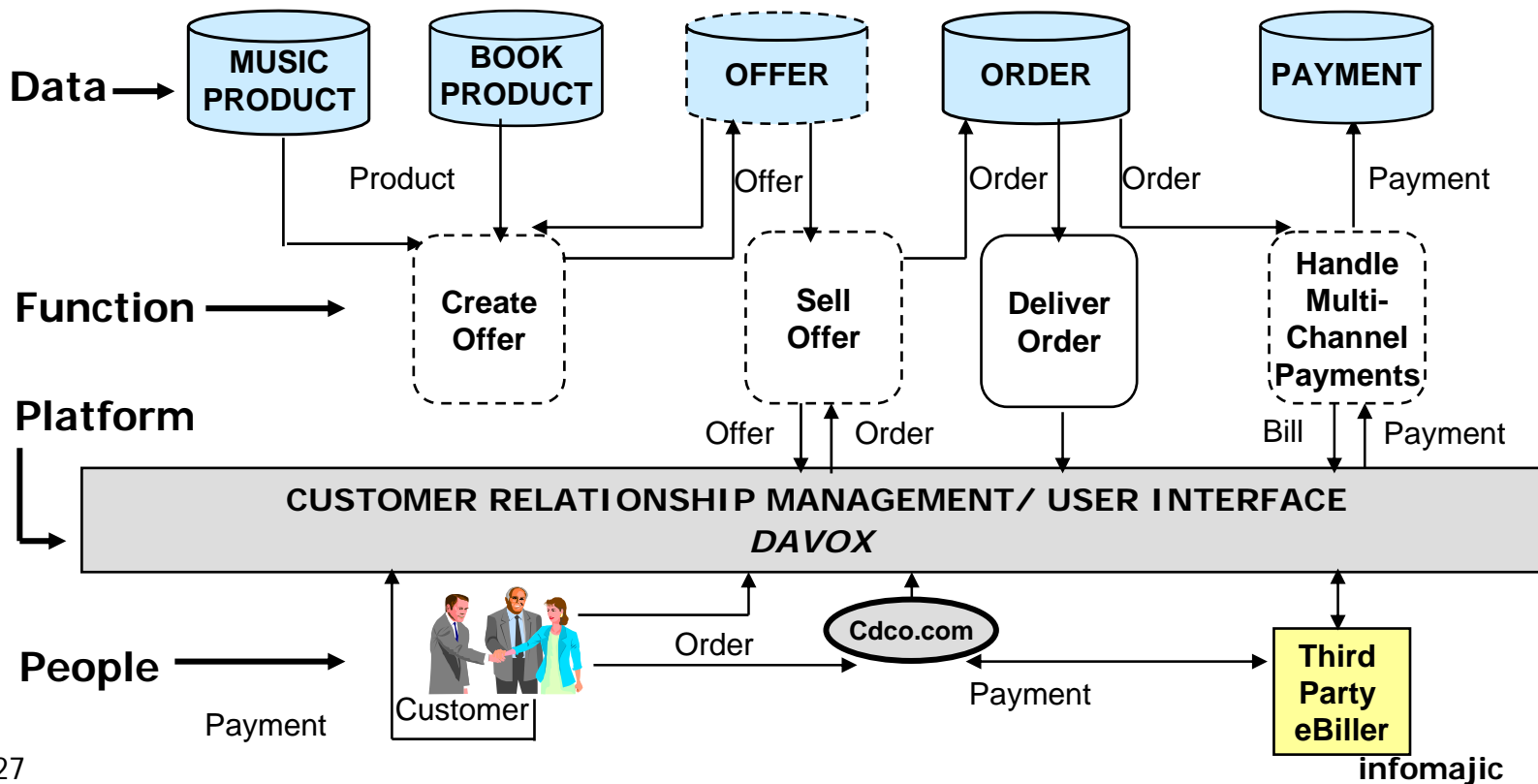
* Different than Zachman *atomic* models – see FAQs

Architecture Models

Example

Slice of CDCo model using

- Standard parts
- Standard representation
- All rows ("Models" Column) are in scope - "integrated"



Architecture Models

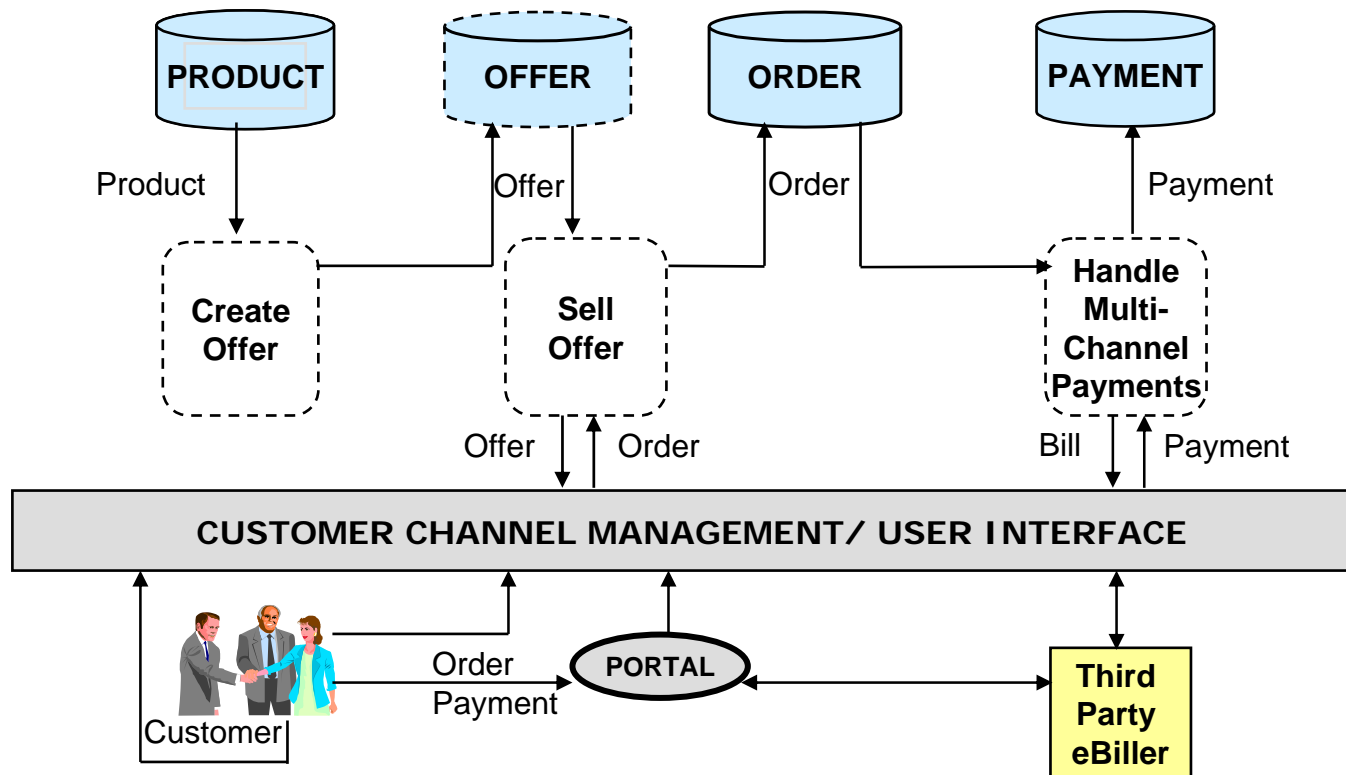
Fundamental Guidelines

4. Determine appropriate level of detail for communicating with audience, e.g.
 - Level 0: Conceptual one-page model relating infrastructure components...used to communicate concepts and elicit feedback (business & IT leaders)
 - Level 1: More detailed, specific view of all or part of Level 0 view...used to communicate direction (business/systems analysts)
 - Level n : Most detailed level of architecture model...used to communicate decisions that require compliance (designers & developers)
 - Presentation Level: Further abstraction of Level 0 model. It may include only components which require funding to implement (decision-makers)

Architecture Models

Fundamental Guidelines

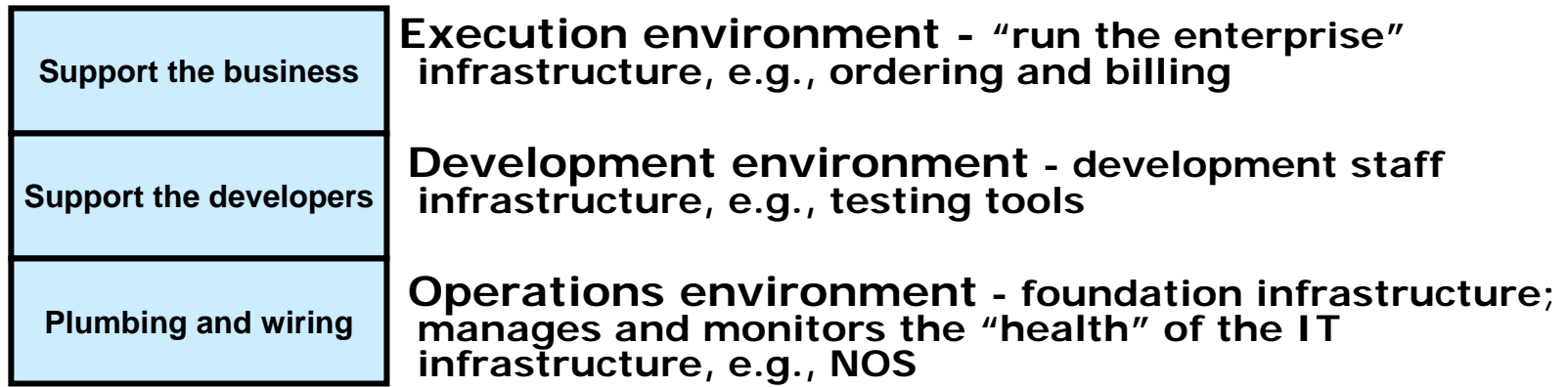
Example—*CDCo* Presentation-Level Model



Architecture Models

Fundamental Guidelines

5. Define the environment. There are many layers of IT architecture, e.g.,



- **Hint: Don't mix and match** - We recommend modeling each environment separately.
- **Quiz: The model in the previous slide is ??? environment only"**

Architecture Models

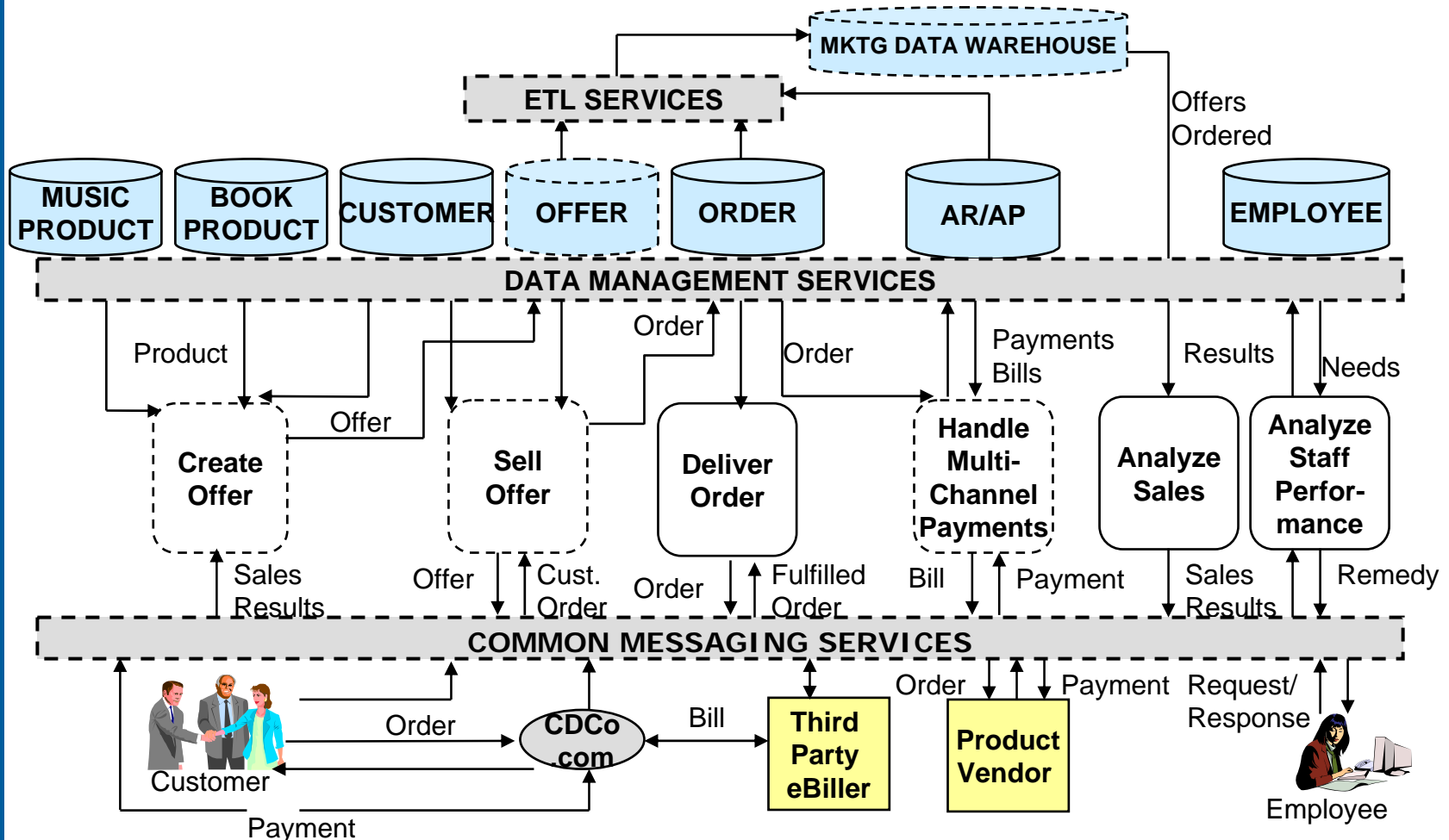
Fundamental Guidelines

6. **Define state.** The state specifies the view or point in time reflected in the model, e.g.,
 - **Current State**
“As is” view—very useful for demonstrating why re-architecture is necessary
 - **Target State**
Desired future state view—“if the world were perfect” perspective
 - **Business plan/budget horizon**
End-of-budget-cycle view—focused view, very useful to high-light projects requiring/receiving funding
 - **Migration Plan**
Interim or migration view—primarily for internal architecture use

Architecture Models

Putting It Together

Example—*CDCo* Level 0 Execution Environment Target State Model



Architecture Inventory

Overview

- **Role of Architecture Inventory**
 - *Inventory* for current state architecture assessment
 - *Standards* for development of target state architecture
- **Definition:**
 - **Inventory is the listing of all the key IT resources of the organization—data, applications, platform/technology and people assets—and their key attributes.**
- **Fundamental Guidelines - Key Points:**
 - **Inventory Scope - Create a separate Inventory for each framework column**
 - **Inventory State - Current state**
 - **Key attributes – Unique by row but ALWAYS capture costs**

Architecture Inventory

Example

Part of *CDCo* Current State Execution Environment Data Inventory

Data Store Name	Data Type/Key Entities	Acronym	Technology	Use	Annual Cost
Customer Data Warehouse	Customer, Customer Acct, Customer Sales	CDW	UDB	Decision Support, Operational	\$2.5 M
Weekly Accounts	Customer, Customer Acct, Customer Sales	WKACT	Flat files	Store Vendor Feeds	\$10. M
Customer Collection Accounts	Customer, Customer Bill Acct, Customer Payments	CCA	Oracle	Operational	\$1.2 M
Customer Payment Fraud Data Mart	Customer, Customer Bill Acct, Customer Payment History	CFDM	Sybase?	Operational?	\$1. M
Campaign Performance	Offer, Product, Campaign Sales	CAMPS		Decision Support	\$.8 M
Marketing Data Mart	Offer, Offer Sales History	MDM	SQL	Operational?	\$1.5 M
Prospects	Prospect, Credit Score	PDB	INFORMIX	Store Vendor Feeds	\$4. M
Offers	Offer, Price, Product	OFFDB	ACCESS	Operational (Prototype)	\$.1 M TBD

Architecture Standards

Overview

- **Definition of an Architecture Standard:**
 - **For Data & Function: the agreed-to or corporate name and description**
 - **For Platform: the name/specification of the selected technology**
 - **For People: role/skill and description**
- **Value/purpose**
 - **Serve as translation of conceptual architecture to specific architecture**
 - **Provide detailed direction for development**
 - **Provide constraints and boundaries for governance decisions**
- **While risking being accused of HERESY, we recommend *the architect* set key standards**

Architecture Standards

Fundamental Guidelines

- Key considerations for setting standards
 1. Establish ownership - The architect together with the business partners should standardize names for critical sets of data and functions
 2. Define the state – target state
 3. Define the environment – begin with execution environment, then standardize the operations & development environments
 4. Define the level of detail, e.g., “model or unit”
 5. Source/Linkage to principles, models & inventory
 - Decomposition of data “mega-entities”
 - lowest level architecture model
 - highest level (conceptual) data model



- Decomposition of Inventory

Architecture Standards

Example

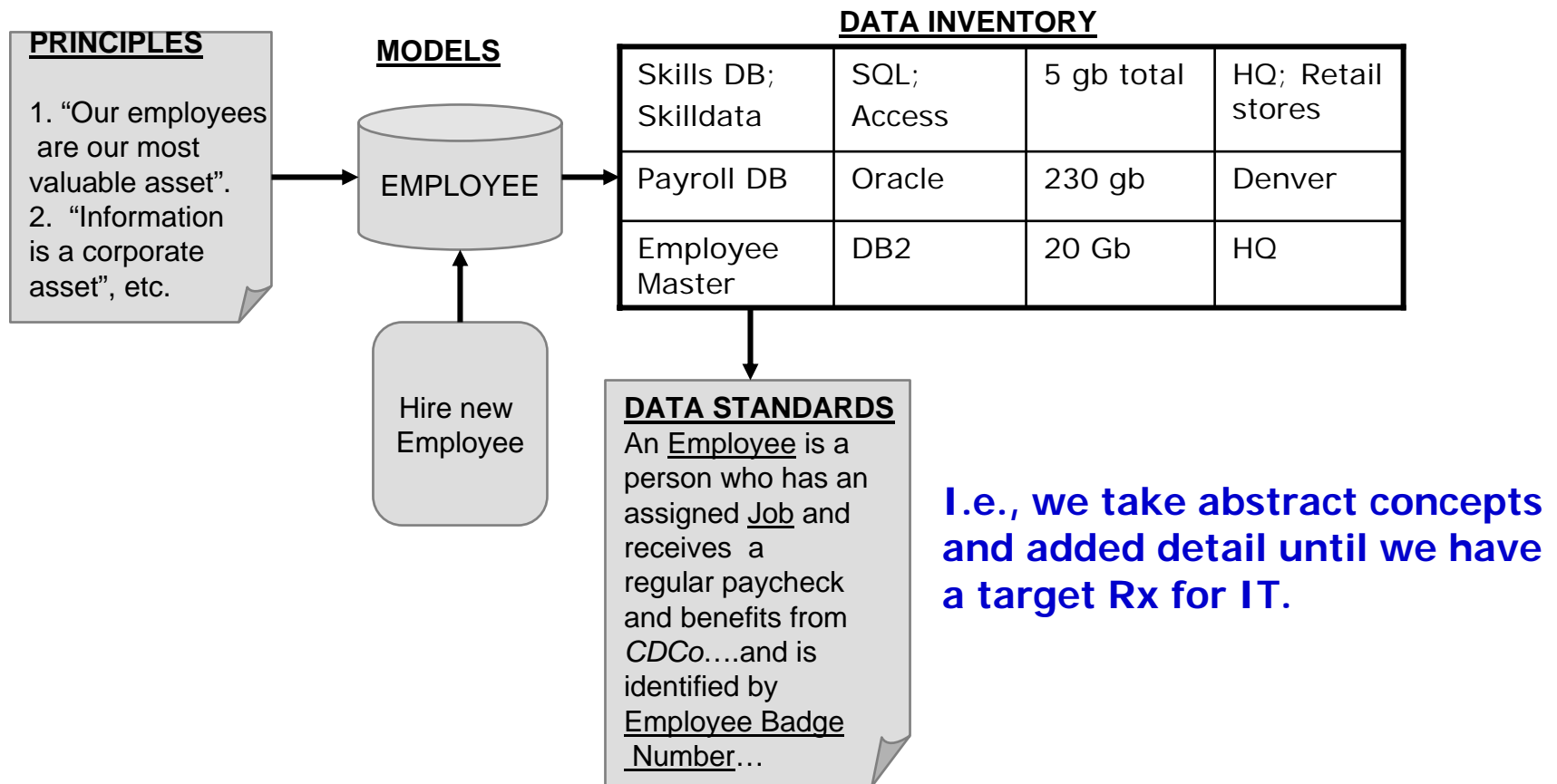
- **Characteristics of a data standard**
 - **Formal Name**
 - **Definition**
 - **Description**
 - **Rules (especially constraints on the creation, deletion, update & access of the data)**
 - **This example assumes that you identified “Employee” as a “mega-entity” (in architecture model or conceptual data model) & yes, this is “business” metadata...**

Name	Definition	Description	Rules
Employee (Identifier)	Unique description of any person hired by CDCo who receives a CDCo/Bookseller Paycheck for providing specific service on a regular full-time or part-time basis to CDCo.	Described by a twelve-position system-generated code assigned to an Employee. Valid Values: alpha-numeric Alias: Badge Number	HR adds (creates) a new Employee Id when the Employee is hired. Employees may retire or be terminated. Employee (Identifiers) may not be deleted.

The Architecture Framework

Putting It Together

- The business drives the architecture
- Architecture outputs are traceable to business drivers & *each other*



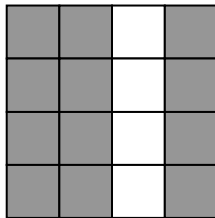
Framework for Implementation

Overview

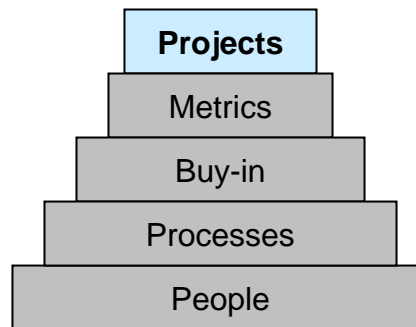
Definition of Framework for Implementation: structure used to define key components of an architecture action plan - minimum set of key areas that need to be addressed in order for target architecture to be actualized

Framework has only a single column – a single implementation plan related to the entire set of architecture outputs

Architecture Framework



Framework for Implementation



Architecture Projects

Overview

Not every project is an architecture project!

Projects that architects pick to begin the implementation of target OR those projects that architects can influence to be implemented consistent with the target architecture

Definition of Architecture Project:

- a discrete (scoped) set
- of (funded) IT tasks
- with a well defined *business* output
- for delivery by an agreed-to date, at an agreed-to cost
- for an identified enterprise sponsor/set of enterprise user(s)
- *that moves the IT infrastructure towards target.*

Architecture Projects

Guidelines

- **Guidelines for translating architecture to projects**

1. Identify potential projects

- **Translate Opportunities to Potential Projects**
- **Identify changes reflected in lowest level target state models**
 - **Examine your models for key changes (e.g., a new function, a new service)**
- **Follow the data - Data is foundational**
 - **Identify key sets of (new/changed) information that support multiple functions (e.g., Customer data)**

2. Select projects

3. Intentionally reduce scope of selected projects

Architecture Projects

Identify Potential Projects

1. Translate Opportunities to Potential Projects

***CDCo* Opportunities**

1. Opportunity: Sell combination offers of book and/or music products
2. Opportunity: Reduce legacy *CDCo* and Bookseller sales channels, systems and data
3. Opportunity: Reorganize *CDCo* & Bookseller IT groups to improve employee satisfaction and reduce redundancy
4. Opportunity: Repair on-line payment
5. Etc.

***CDCo* Potential Projects**

1. Sell Offers

- Build Offer data store
- Link *CDCo*, Bookseller and other Products to Offers
- Build a function that supports sales of offers
- Provide customers and sales reps access to the function.

2. Integrate systems and data

- Integrate *CDCo* and Bookseller Customer data: Assign a cross-organization Customer Number to all customers; modify marketing, sales and ordering systems; consolidate customer data stores
- Integrate Employee data: Assign a common Badge Number to all *CDCo* and Bookseller Employees; modify HR systems; consolidate employee data stores, etc.

3. Reorganize IT

- Select *CDCo* or Bookseller applications for cross-organization use
- Design & deliver cross-training
- Consolidate web sites

4. Replace online pay

Architecture Projects

Selecting Projects

2. Guidelines for selecting projects

– Eliminate white elephants

- Estimate project size. Eliminate potential projects that are unrealistically expensive.

• Potential elephant – Project #2

Integrate *CDCo* and Bookseller Customer data:

Integrate *CDCo* and Bookseller Customer data:

Assign a cross-organization Customer Number to all customers; modify marketing, sales and ordering systems; consolidate customer data stores. Begin consolidation with *CDCo* and Bookseller Customer Card History data by linking new Customer Number to card history...

Architecture Projects

Selecting Projects

2. Guidelines for selecting projects

– Focus on a few

- **Identify Key Data Areas—Map identified Projects to Business Target State and Target Architecture Model Data Areas (mega-entities)**
- **Focus on projects with high frequency of intersection with business targets and data**

ARCHITECTURE DATA AREA ↴

BUSINESS TARGET ↴

	Offer	Order	Customer
1. Enhance & Leverage ebusiness	Project 1, 2, 6	Project 1,5	Project 1, 4, 6
2. Improve Knowledge of Customer Relationships	Project 2, 6	Project 1, 2, 5	Project 1, 2, 5, 6
n. Consolidate Financials	Project 7		

← IDENTIFIED PROJECT

Architecture Projects

Selecting Projects

2. Guidelines

- **Apply Viability Criteria to remaining set of projects**
 - Manageable in scope
 - Deliver benefit for each implementation
 - address enterprise gap; solve enterprise problem; actualize enterprise opportunity
 - Do not result in “throw away” solutions
 - Support flexibility and affordability
 - Allow for phased migration to target

Potential Projects

- **Viable?**
 - Build Combo Offers?
 - Integrate Customer data?
 - Consolidate web sites?
 - Replace Payment portal?

Architecture Projects

Reducing Project Scope

3. Apply the set of six data-focused strategies to selected projects to intentionally reduce project scope, e.g.,

4. Smaller Is Better

Yesterday: Implement all the instances of the target data

VS

Today: "Smaller is better"

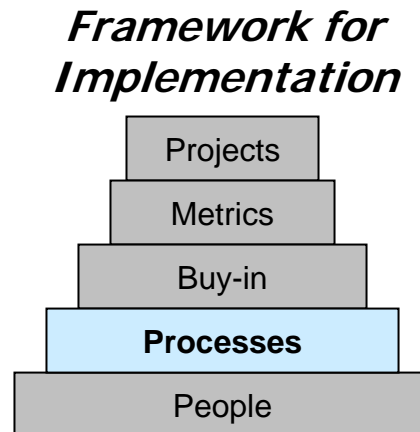
- Implement a *small subset of the target data population*; identify an exclusive subset of instances of same-entity data building a manageable, deliverable chunk

Implementation Framework

Architecture Processes

Definition: Process - the combination of people, equipment, raw materials, and work methods that produces a product or service*

Value – Putting related processes in place supports the syndication of architecture



* *The Memory Jogger*, AT&T. Lawrence, MA: G.O.A.L., 1985

Architecture Governance Process

Overview

Definition of Architecture Governance:

- **Sanctioned, formalized process**
- **Used by a recognized team**
- **On a regular basis**
- **To assess the extent to which infrastructure development complies with or deviates from architecture**
- **And recommend action**

Value – Ensures that architecture is integrated into the build process

Architecture Governance

Guidelines

Guidelines for developing an Architecture Governance Process:

- 1. Define a formal process, i.e.,**
 - Inputs**
 - Outputs**
 - Repeatable Steps**
- 2. Set up a review schedule**
- 3. Establish decision-making criteria**
- 4. Define outcomes - Include an exception mechanism**
- 5. Document outcomes**

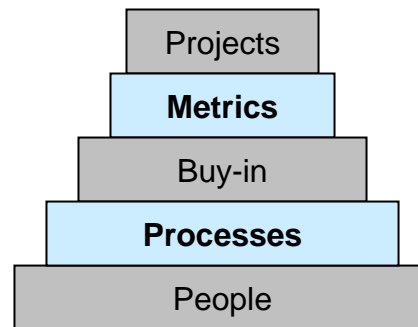
Implementation Framework

Architecture Measurements

Definition: the architecture objectives* and what constitutes successful achievement of them

Value – Demonstrates value of architecture to the business

Framework for Implementation



* (Objectives 101...An Objective is specific, measurable and time-bounded)

Architecture Measurements

Overview

What to Measure

- **Value - Business result**
 - Identify a quantitative measure of change in the enterprise results. E.g., Amount of Revenue, Number of Sales

Example

CDCo Project: Develop Combo Offers

Business results - "Increase amount of revenue"

Objective Measure: By 12/31/2006, Combo Offers sold via the web will result in \$5 Million in revenue.

Architecture Measurements

Overview

What to measure

- Effectiveness – Process results
 - Identify a quantitative change in the outputs of a process, e.g.,
 - Collect findings from architecture compliance reviews

Example - CDCo Governance Process:

Potential outcomes

- Overall design assessment (compliant, non-compliant, exception granted, modifications underway)
 - Findings: written comments identifying problems or suggestions for modifying the design
 - Action required: what follow up is necessary. E.g., exceptions require plans to migrate to compliant design within one year
- Measure if there is a change in outcomes.

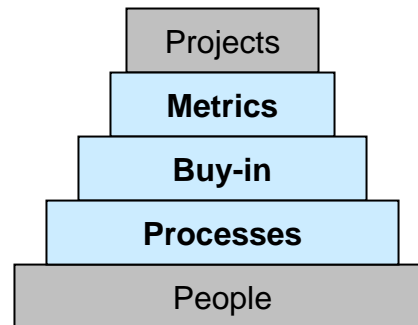
Implementation Framework

Architecture Concurrency

Definition: Ownership of the architecture by the organization demonstrated by actions

Value – Insurance (architecture “life expectancy”)

Framework for Implementation

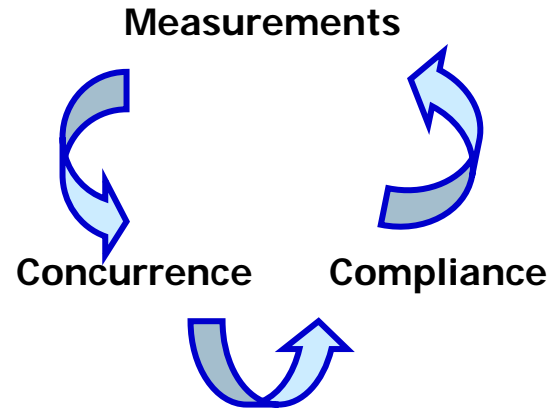


Architecture Concurrency

Overview

Seek support from

- **Business leadership**
- **IT leadership**
- **The CIO**
- **Development organization**



Architecture Concurrency

Overview

Purpose of seeking executive buy-in:

- To confirm architecture direction
- For review, selection and approval of key projects (*“Show me the money...”*)

Guidelines for gaining executive buy-in:

- Schedule regular architecture reviews with business department heads, the CIO, IT department heads and/or steering committee
- Report results
- Request feedback

Implementation Framework

HR Policies/People

Definition: All the formal procedures that affect how the architects and related personnel are treated by the organization

Value:

- Ensures vitality and effectiveness of the architecture team
- Promotes buy-in

HR Policies

Overview

Evaluate HR policies for the architecture team

- 1. Are there staffing criteria for architecture jobs?**
- 2. How do architecture positions operate (roles & relationships) in the organization?**
- 3. How are architects evaluated and compensated?**
- 4. Is there an architecture career path?**

People

Roles & Responsibilities

Definition - Formal role or job description that defines duties, outputs and interfaces

Value – Setting clear performance expectations promotes achievement of desired results

Key roles:

- 1. Information Steward**
- 2. Chief Architect**
- 3. Data Architect**
- 4. Application Architect**
- 5. Technology Architect**

Roles & Responsibilities

Key Architecture Roles

Example - Data Architect

The role of the data architect is to develop and translate conceptual architecture models to conceptual data models, to define key data standards, and to recommend standard data practices.

The data architect is a voting member of the architecture governance forum and reports to the chief architect.

S/he is responsible to deliver conceptual architecture and data models, data standards based on the target architecture and, for new projects, logical data models that translate the architecture to input for system and database design.

People

Architecture Organization Design

Definition: description of how architecture roles interoperate with each other and with roles external to architecture

Value - supports architecture effectiveness

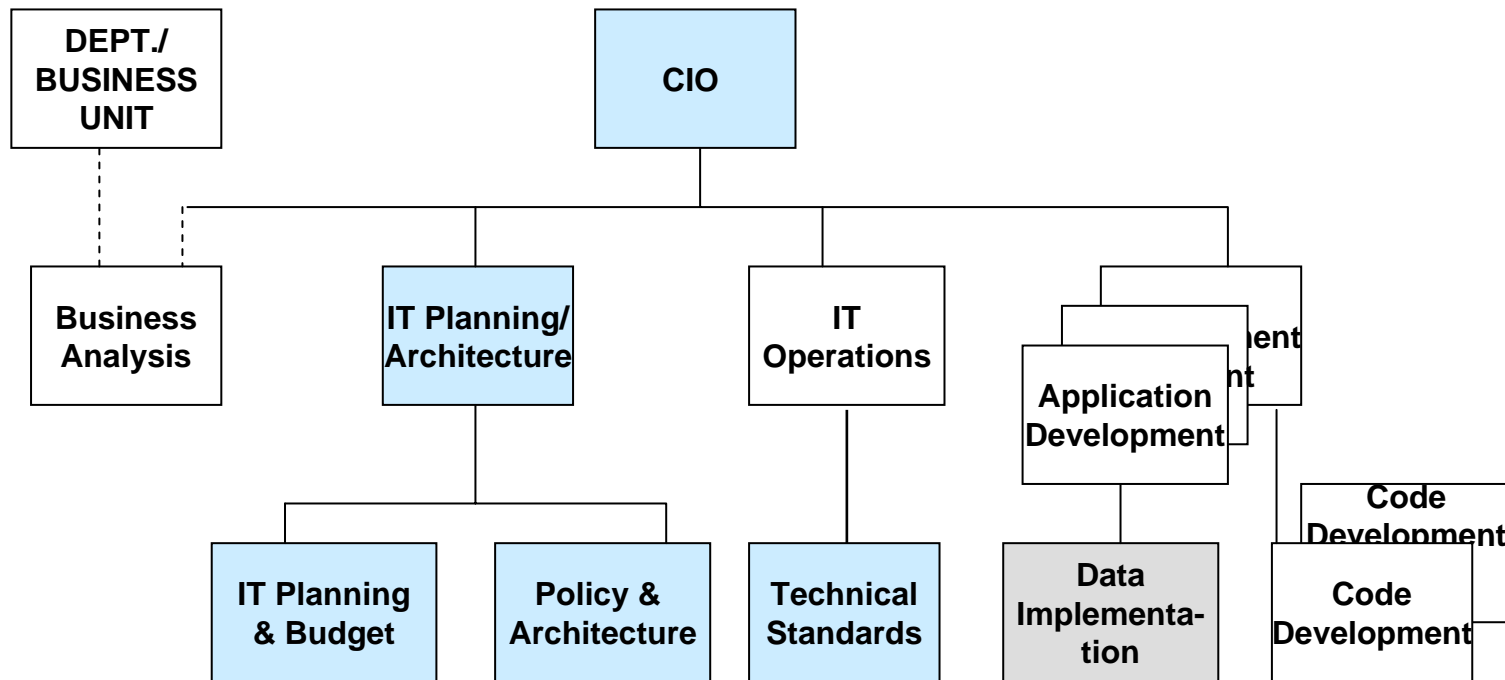
Caveats:

- **A good organization structure can support the implementation of architecture**
- **Organization structure changes can only address organization structure problems**
- **A less than optimum organization structure is not impossible to work around**

Organization Design

Example

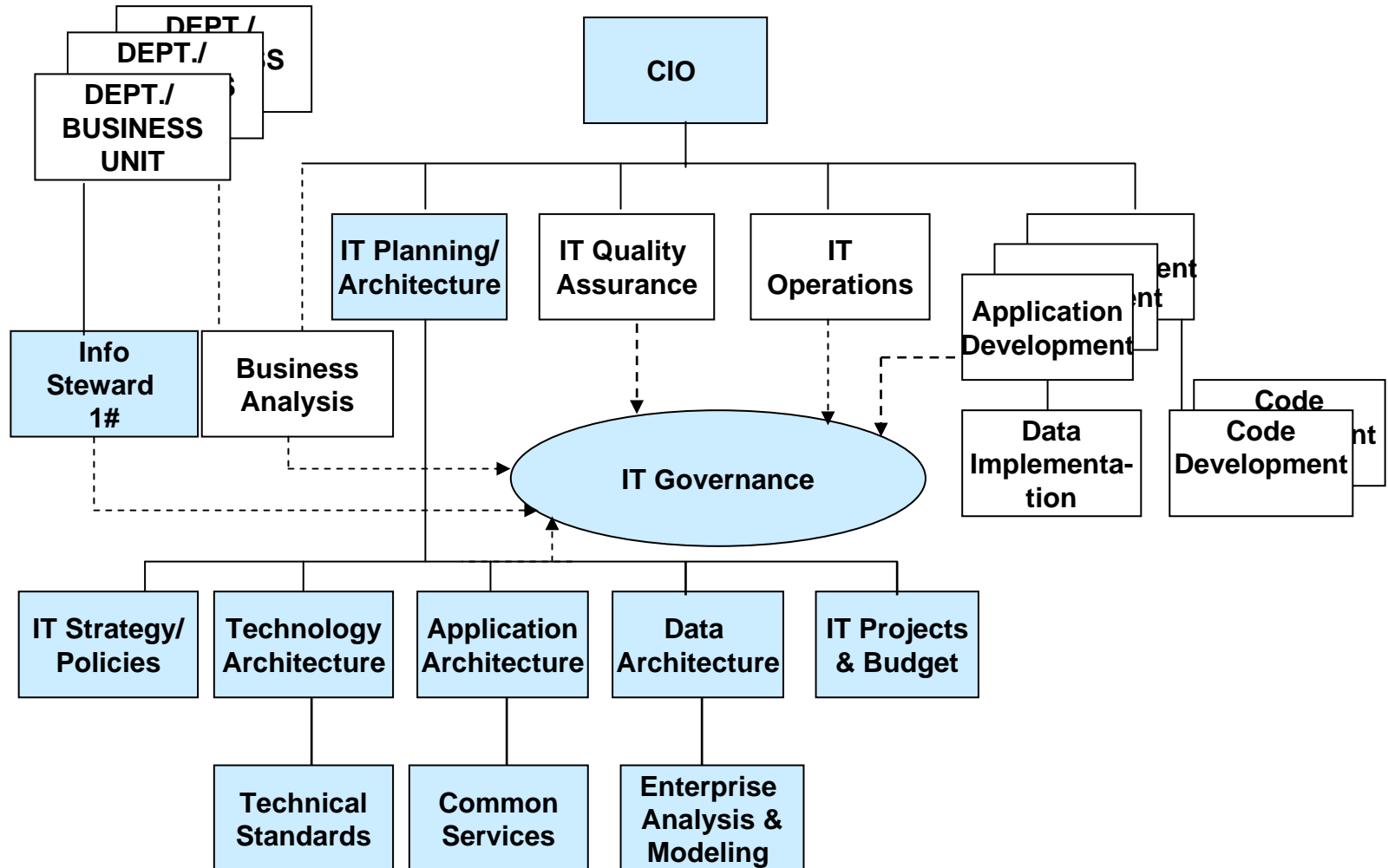
Example Organization Structure #1



Organization Design

Example

Example Organization Structure #2



A Final Word on Architecture *Processes*

Key Processes

- Architecture Development
- Architecture Governance
- Architecture Maintenance
- Related HR processes
 - Staffing
 - Evaluation/compensation
 - Career development
 - Organization design
- **IT processes**
 - **Integration into SDM**

IT Processes

Integrating Architecture into SDM

Include architecture in the Systems Development Methodology/Process

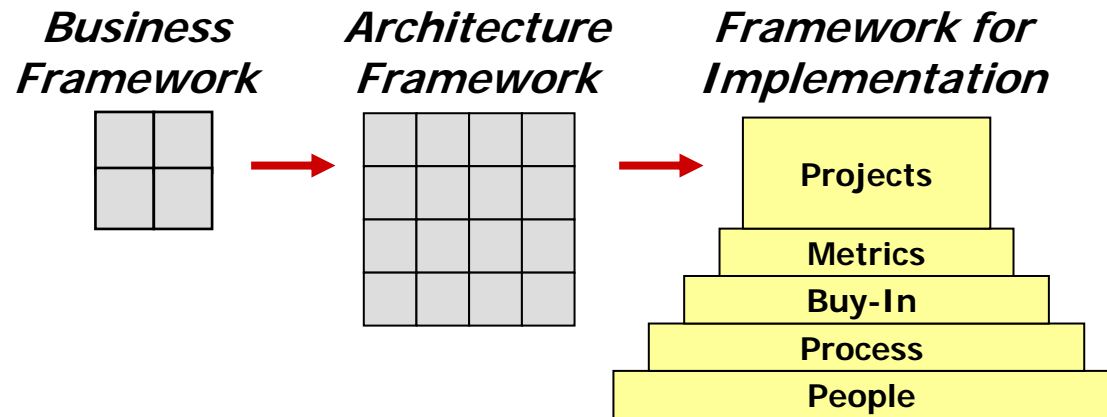
- **Identify and define the key artifacts architects and developers will exchange:**
 - **What outputs architects provide to analysts/designers/developers for direction?**
 - **What outputs analysts/designers/developers provide to architects for review?**
 - **At what point/phase in the development life-cycle do these exchanges to occur?**

- **Incorporate the “rules” in SDM**

Implementation Framework

Putting It Together

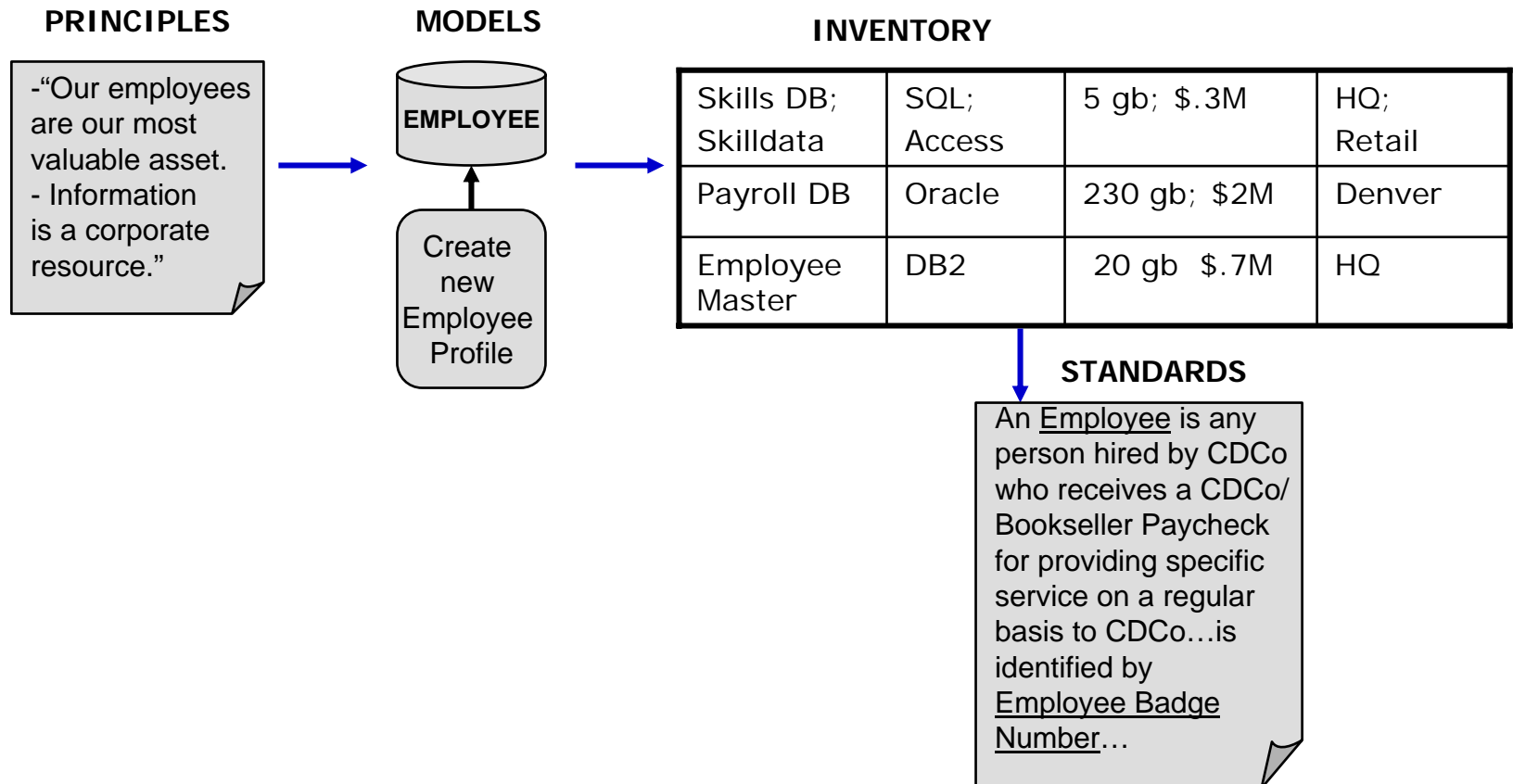
- Action plan for implementing architecture
 - Translate EA to small projects
 - Add metrics & “sell”
 - Put supporting processes – especially governance – in place to provide “clout”
 - Address the people concerns – HR/staffing, role & organization design



Toolkit

Putting It Together

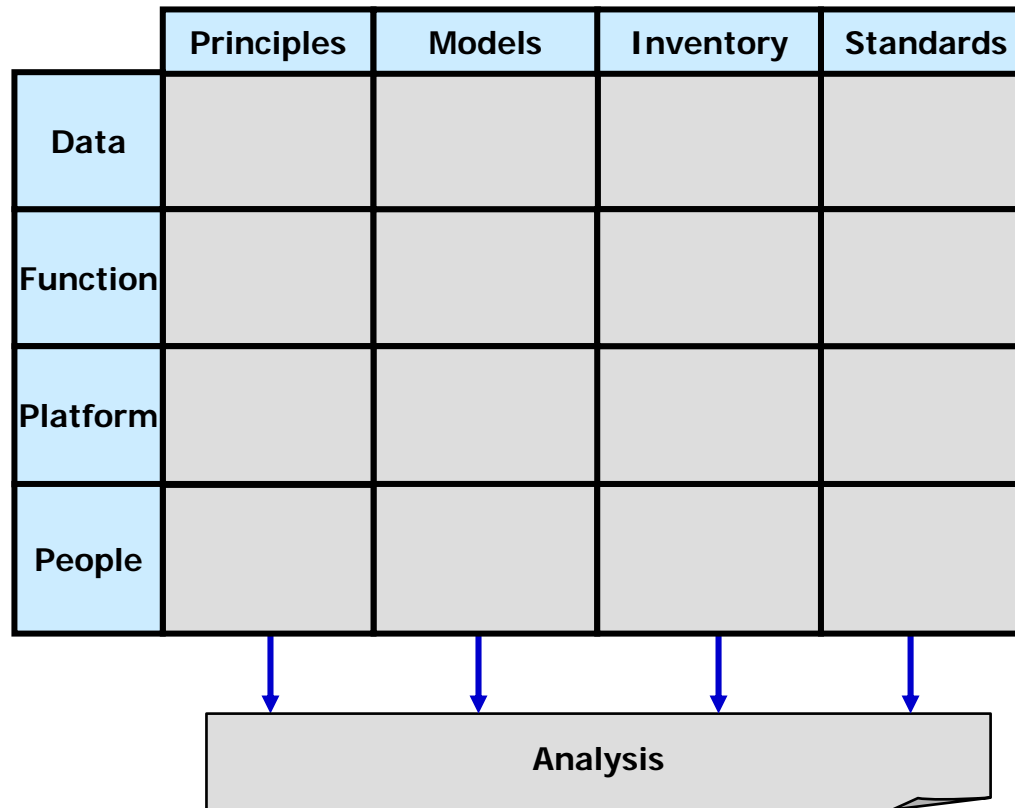
- *Frameworks* are related to each other...
- *Outputs* are related to each other...



EA, Analysis & Design

Putting It Together

- Outputs are related to each other...
- ...*And* to Analysis & Design



Architecture Model & *Conceptual Data Model*

- Relationship of architecture model & data model

Enterprise Architecture Model	Data Model
Business View of IT	IT View of Business
Strategic View	Logical View
Planning Phase	Analysis Phase
Relationship of Data to Business Functions, External Interfaces, Services, Technologies/ Strategies	Relationship of data to other data
Key Data Interactions, Capture and Storage	Data Structure

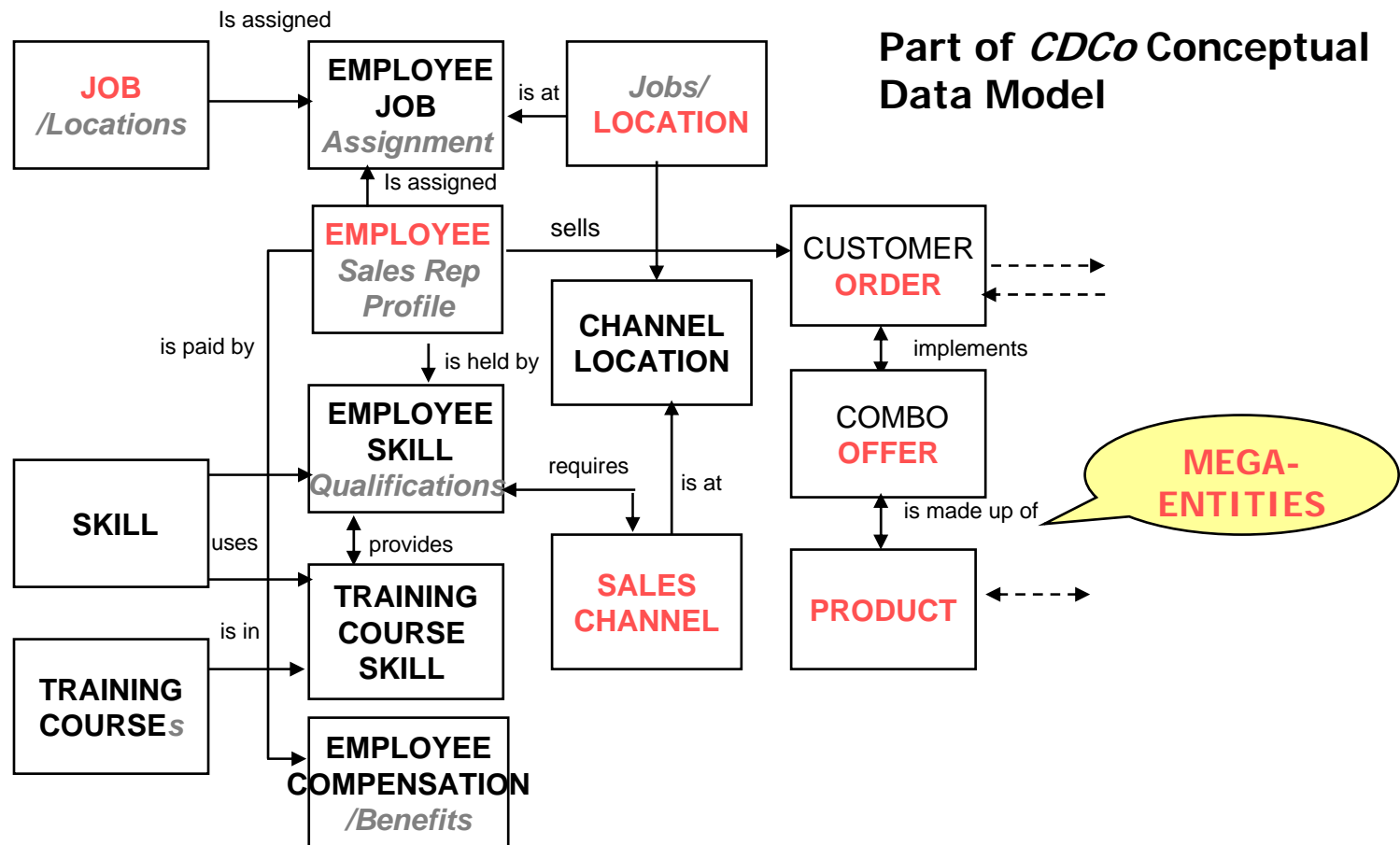
Art?

Science?

Framework Outputs

Drive Analysis & Design

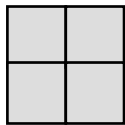
- The architecture model is created *before* the data model is created
- The data model is *based on* the architecture model



Last But Not Least...

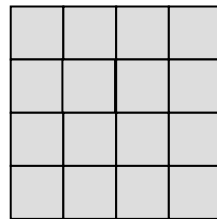
- Key takeaways - To develop a “good enough” architecture in months (not years):

Business Framework



- Tightly link the architecture to the business

IT Architecture Framework



- Use an approach that is disciplined (repeatable) and traceable back to the business

Framework for Implementation



- Develop an action plan to support implementation



- Architecture outputs are integrated with each other *and* with other IT outputs
- Base analysis & design on the architecture

Appendix

“FAQ”

Comparison of *Toolkit* with Zachman Framework for Enterprise Architecture

- **John Zachman has done ground-breaking work in architecture thought and philosophy.**
 - Our frameworks are consistent with but simpler than the Zachman Framework.
• What we are really talking about here are practical differences in approach.
- **The Zachman Framework represents a philosophy**
- **Many required outputs the same**
 - (e.g., Some of “top left hand” Zachman cells are covered in the business framework.

Appendix

"FAQ"

Comparison of *Toolkit* with Zachman Framework for Enterprise Architecture

- **Key differences:**
 - **Design not in scope**
 - **Network part of platform**
 - You could add a separate row to AF if you desired
 - **Framework for Implementation**
 - We have found it to be extremely important to require that the architecture is considered complete only when accompanied by an action plan which includes addressing "soft" issues
 - **Methods for creating cell output**
 - In our experience, it is really important to define *how* outputs are created and to capture the linkages between outputs. This provides not only an audit trail – i.e., how this principle is linked back to the business drivers, but also integration across outputs – i.e., how this function is linked to this data on this platform...