Six Habitual Architecture Mistakes

and How to Avoid Them

Eddie Sayer, Ecosystem Architect, Teradata

TERADATA ANALYTICS UNIVERSE 2018

Habitual Architecture Mistakes

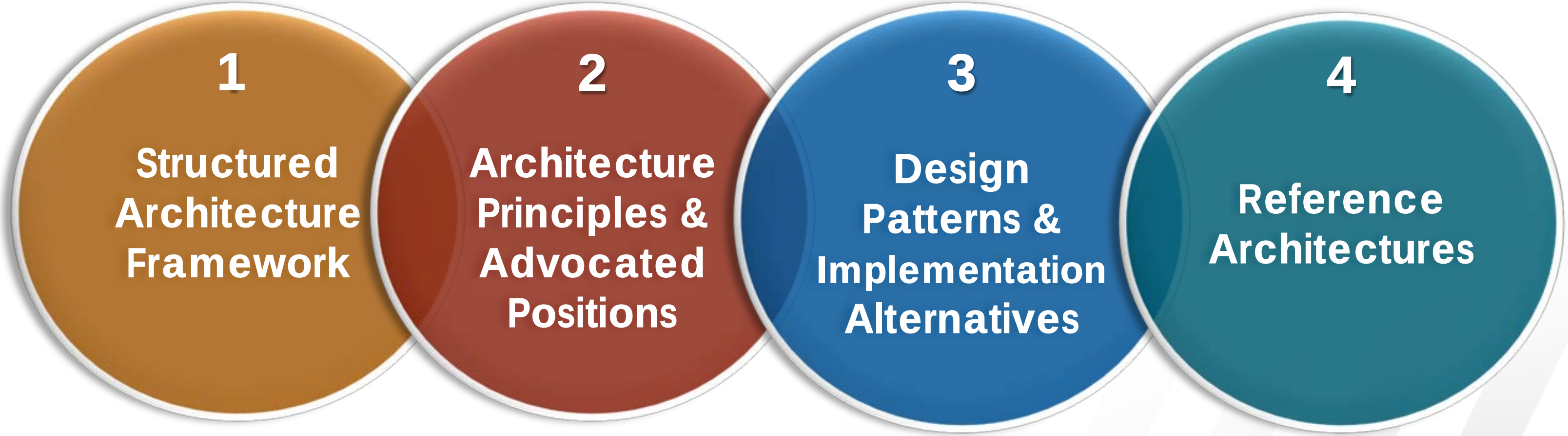
Six Habitual Architecture Mistakes

Does Not Work		Best Practice
Technology focused	1	Business driven
Accidental evolution	2	Methodical development
Art vs. Science	3	Art *and* Science
Contentious debates	4	Principle-based decisions
Reinvention of ideas	5	Reusable patterns
Pretty pictures	6	Documented blueprints

Avoiding the Mistakes

Architecture Components

Four architecture components to avoid the mistakes:



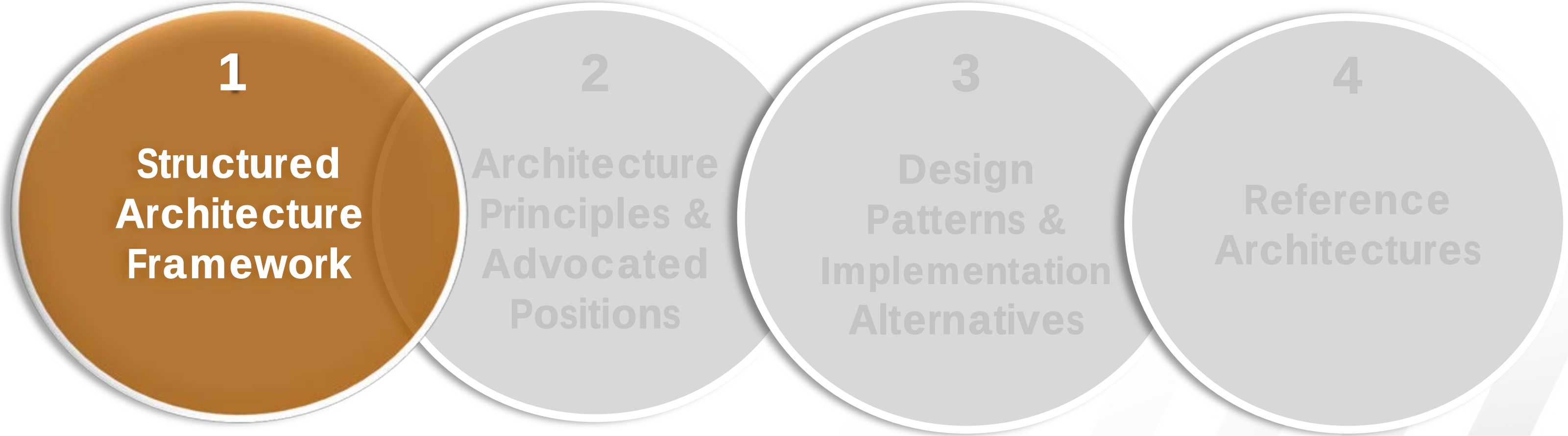
1
Structured Architecture Framework

2
Architecture Principles & Advocated Positions

3
Design Patterns & Implementation Alternatives

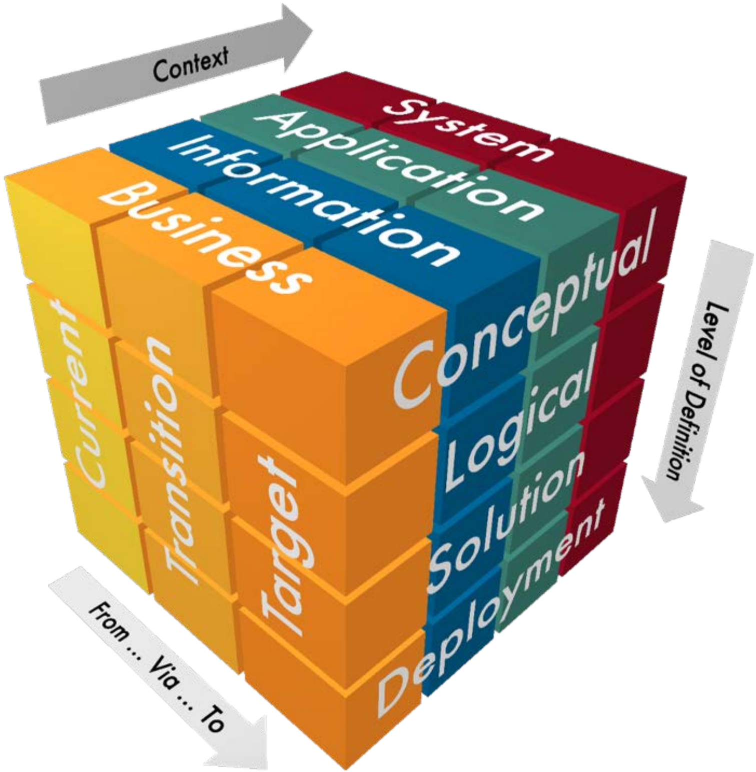
4
Reference Architectures

Architecture Components


1
Structured Architecture Framework
2
Architecture Principles & Advocated Positions
3
Design Patterns & Implementation Alternatives
4
Reference Architectures

Structured Architecture Framework

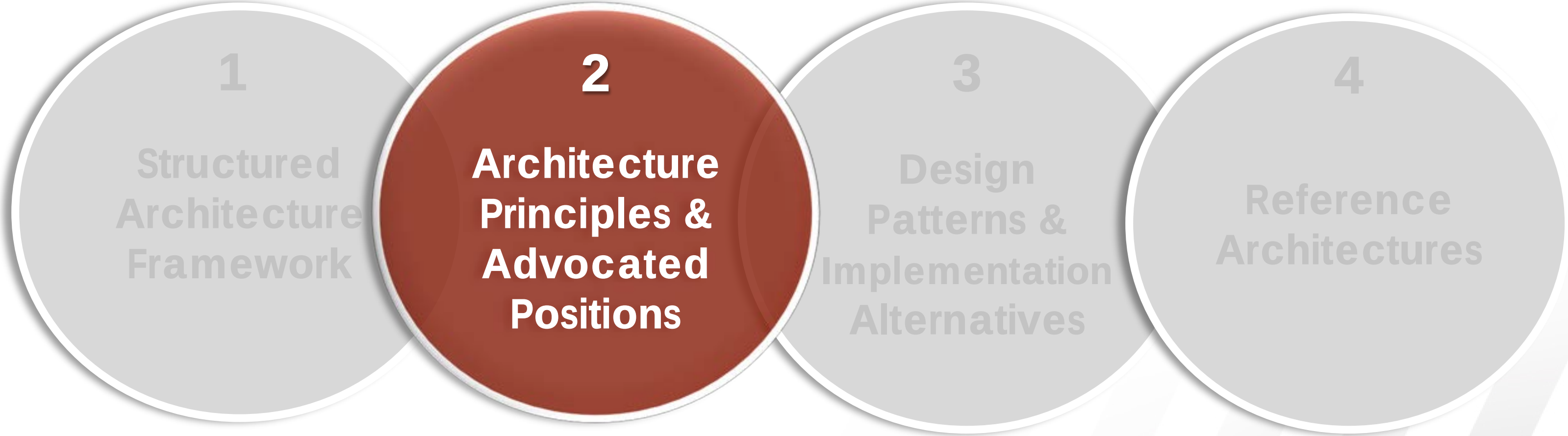
TERADATA ANALYTICS UNIVERSE



- Holistic **structure** with implied order of process

- Development of **systems is the final stage** in the approach

Architecture Components


1
Structured Architecture Framework
2
Architecture Principles & Advocated Positions
3
Design Patterns & Implementation Alternatives
4
Reference Architectures

Architecture Principles

Principles
Simplicity
Abstraction
Isolation
Standards
Scalability
Extensibility
Supportability
Integrity

The foundation for making architecture decisions

Designs and variations discussed early-on

Usable Documentation

Principles

- ▶ **Simplicity**
- Abstraction
- Isolation
- Standards
- Scalability
- Extensibility
- Supportability
- Integrity

Statement

Where more than one option exists for the design to deliver a business requirement, then, if there is no identifiable reason to choose the more complex option, use the simpler option

Rationale

Simpler architecture = simpler systems; Development, implementation, and maintenance of simpler systems is less expensive

Benefits

Systems based on this principle should be smaller, easier to build, maintain and govern throughout their life-cycle

Advocated Positions

Experiences and proven **best practices** evolve over time into advocated positions

Strongly held views based on what has, and has not, worked previously

Example Advocated Positions

Advocated Position	Short Description
Touch it, take it (extract all columns)	• Load all the source table into tier 1, not just the requested columns so requests for additional data do not require ETL script changes
Collect metadata	• The architecture should support a metadata driven approach
Include Acquisition / Staging layer in architecture	• All data architectures will include an acquisition layer
No production reporting from the Data Lab	• Data in the lab is not governed; operational reporting can introduce risks to the business (i.e., data quality, regulation, security)
Integrated logical and physical data models	• Logical and core physical data models are integrated data models
Enforce referential integrity	• Enforce RI as data is loaded so exceptions are flagged for data stewards
Prioritize data access over data loading	• Speed of user access should be more important than speed of loading; semantic model optimizations often drive access improvements, but require joins and views
Full copy of source data objects in acquisition area	• Full copy of source data objects in acquisition area

Usable Documentation

Acquisition Layer Positions

▶ **Include an Acquisition/Staging Layer**

Touch It, Take It

Full Copy of Source Data Objects Within Acquisition Area

Statement

All data architectures will include an acquisition layer

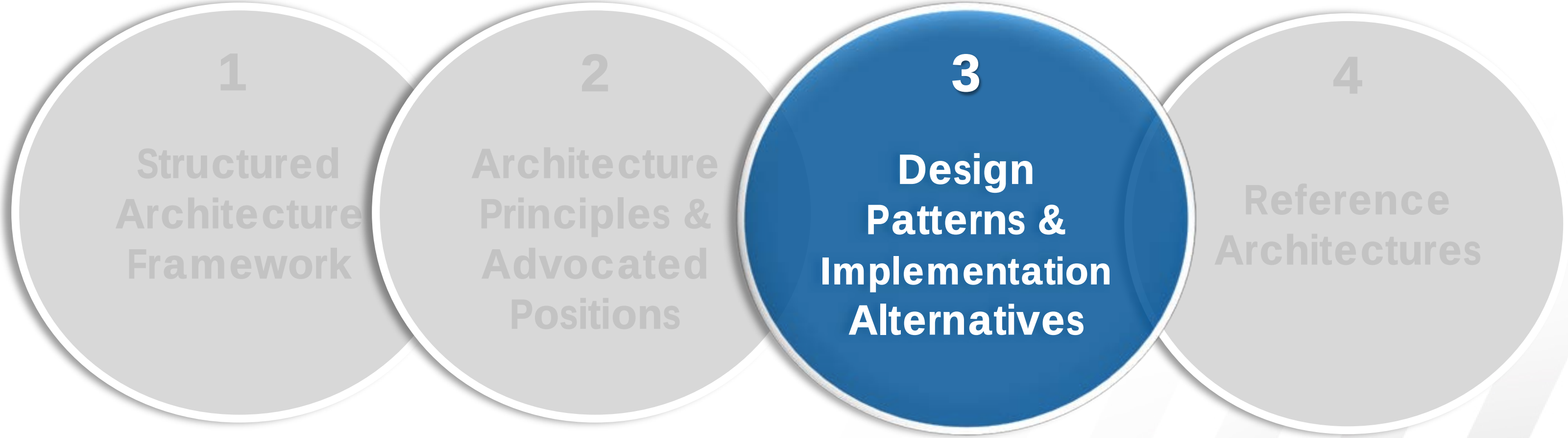
Rationale

This layer serves primarily an audit function, in that it must contain the source data as it was at the point of extraction, supporting the traceability and auditability positions; users will not be given access to this layer for production usage

Benefits

Must be persistent to fulfill its audit and reconciliation functions; data conditioning/data profiling permissible in this area since they are non-destructive; transformation is not permitted as it is [potentially] destructive

Architecture Components



1

Structured Architecture Framework

2

Architecture Principles & Advocated Positions

3

Design Patterns & Implementation Alternatives

4

Reference Architectures

DPs and IAs

Design Patterns

- Logical **architectural options** with pros/cons and decision criteria
- Enables architects to have a **starting point** for evaluating **architectural choices**

Implementation Alternatives

- **Physical counterparts** to design patterns
- Defines **instantiation options** for materializing an architecture

Avoid Reinvention

Simple Example

Design Problem – road junctions

- Design Patterns
  - > Cross-roads
  - > Cloverleaf intersection
  - > Roundabout

- Implementation Alternatives for **Cross-roads** Design

Pavement Markings

Pedestrian Signals

Traffic Flow

- Traffic lights
- Four-way stop
- Two-way stop (minor road gives way)

Architecture Examples

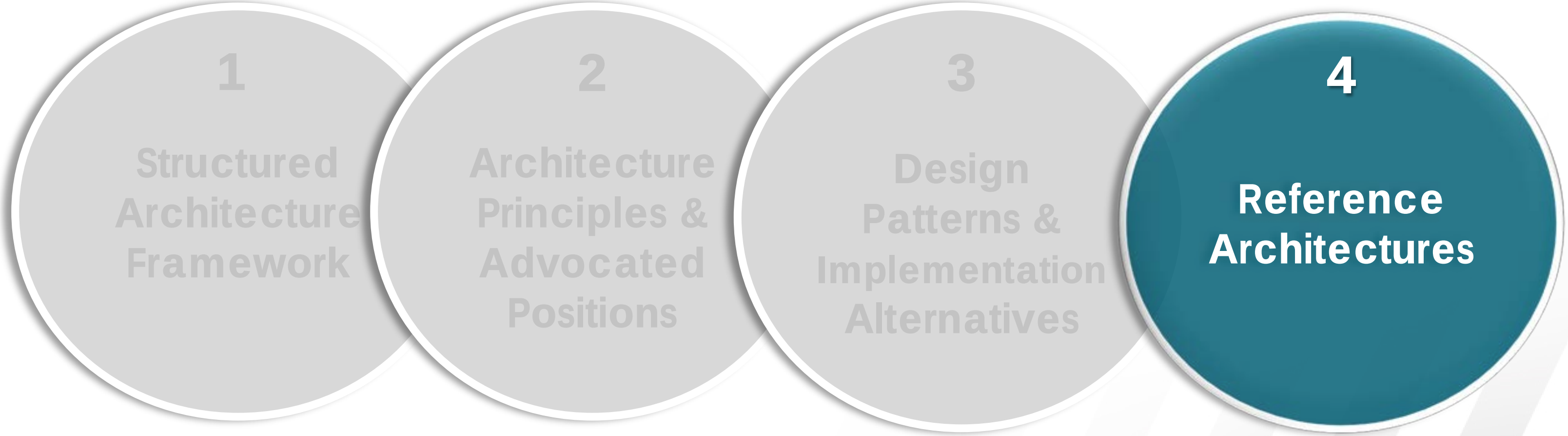
Design Patterns

- Data Population
- Reference Data Management
- Data Exception Handling
- Data Modeling
- Hierarchy Management
- Access Layer Design
- Data Quality
- **Information Architecture**

Implementation Alternatives

- Data Synchronization
- Logical-to-Physical Design
- Temporal Data
- Transaction Management
- Entity Matching & Harmonization
- Security
- Key Management
- **Platform Infrastructure**

Architecture Components



1

Structured Architecture Framework

2

Architecture Principles & Advocated Positions

3

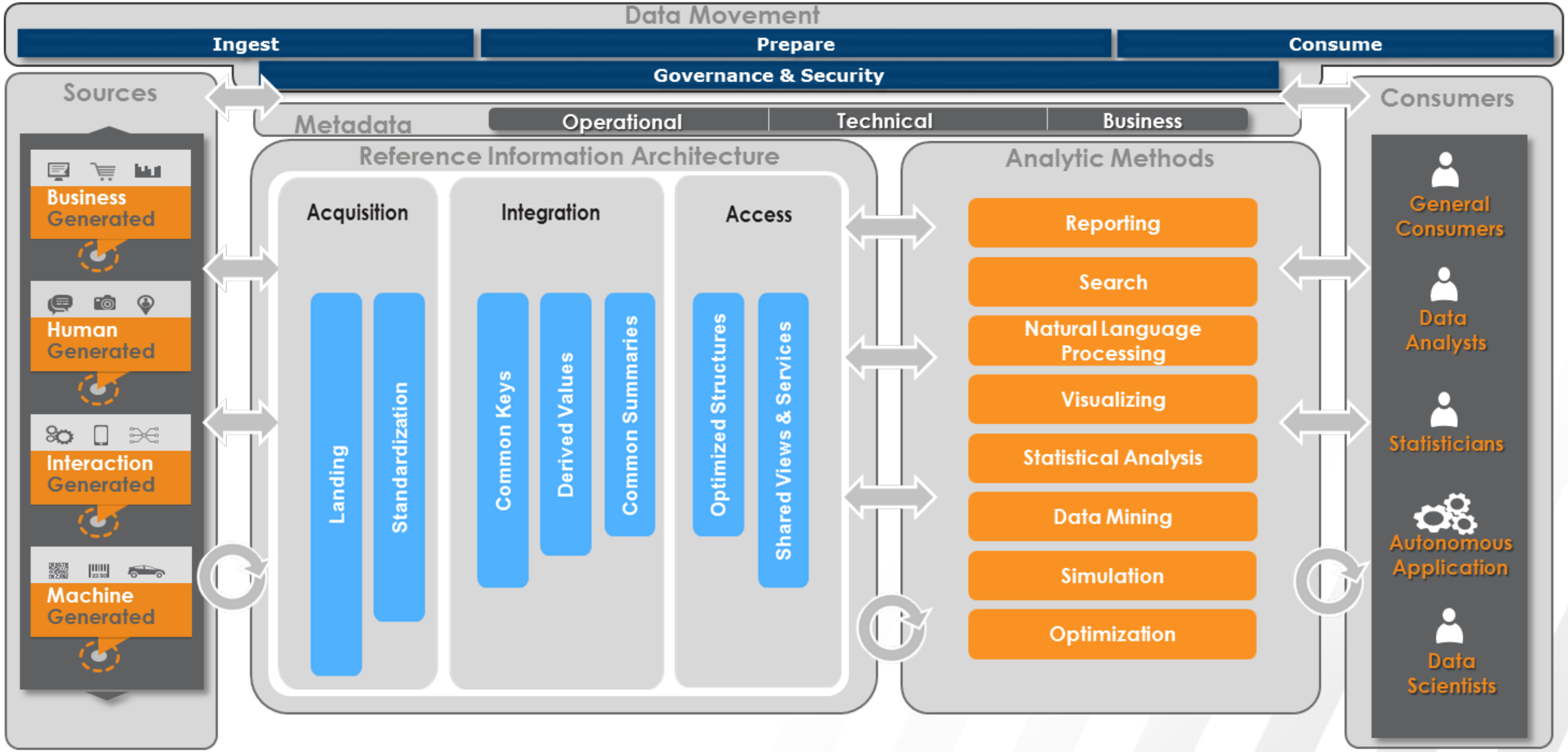
Design Patterns & Implementation Alternatives

4

Reference Architectures

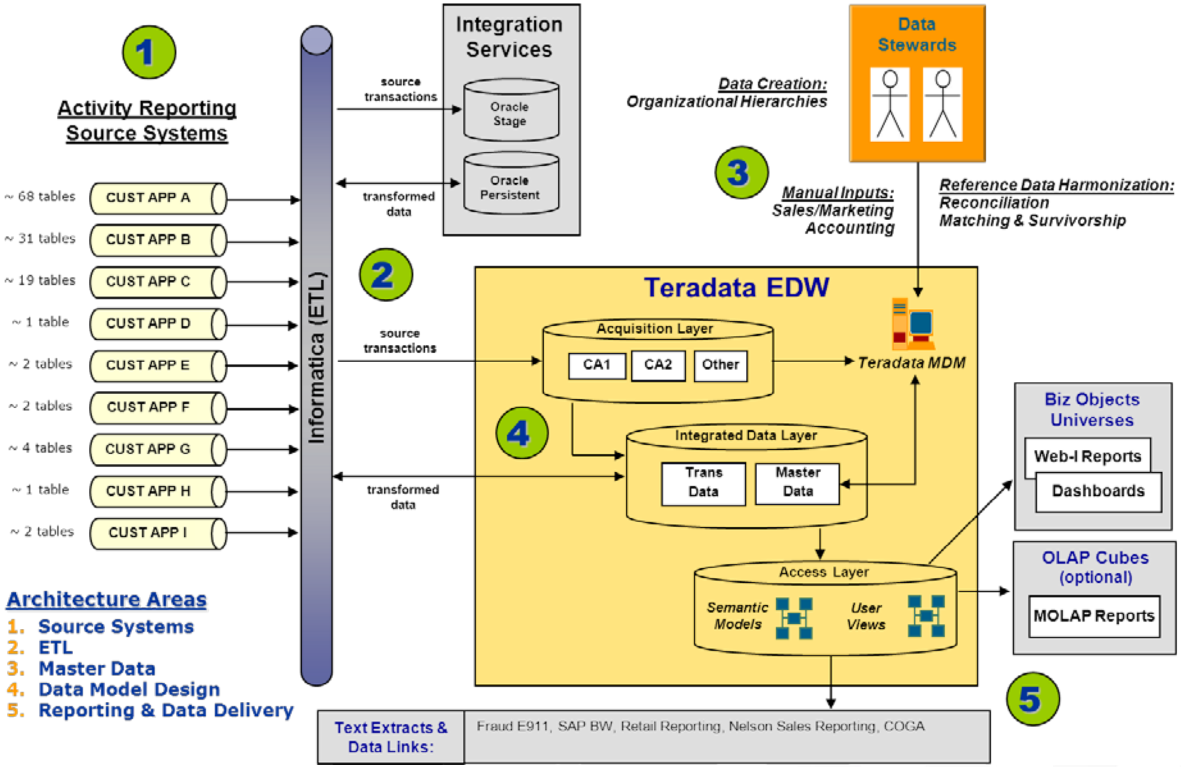
Example Reference Architecture

TERADATA
ANALYTICS UNIVERSE



Customized to Your Requirements

TERADATA
ANALYTICS UNIVERSE

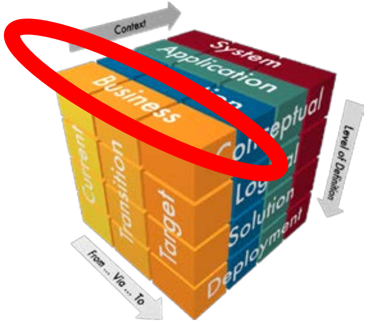


Architecture Areas
1. Source Systems
2. ETL
3. Master Data
4. Data Model Design
5. Reporting & Data Delivery

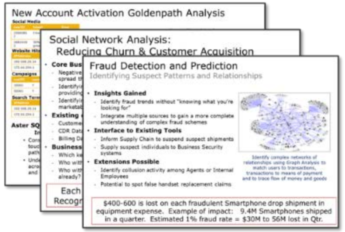
Other Reference Architectures

TERADATA
ANALYTICS UNIVERSE

Business



Discovery Use Cases



New Account Activation Goldenpath Analysis

Social Media

Website Hit

Campaigns

Search Term

Aster SQ

Social Network Analysis:
Reducing Churn & Customer Acquisition

- Core Bus
  - Negative spread t
  - Identifyi providing
  - Identifyi marketab
- Existing
  - Custome
  - CDR Data
  - Billing D
- Business
  - Which ke
  - Who with
  - Who with already?

Each Recogn

Fraud Detection and Prediction
Identifying Suspect Patterns and Relationships

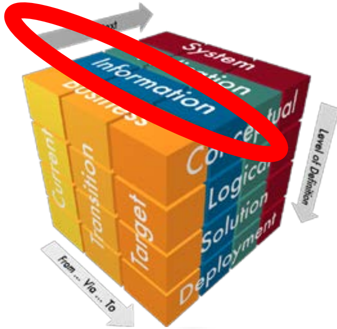
- Insights Gained
  - Identify fraud trends without "knowing what you're looking for"
  - Integrate multiple sources to gain a more complete understanding of complex fraud schemes
- Interface to Existing Tools
  - Inform Supply Chain to suspend suspect shipments
  - Supply suspect individuals to Business Security systems
- Extensions Possible
  - Identify collusion activity among Agents or Internal Employees
  - Potential to spot false handset replacement claims

Identify complex networks of relationships using Graph Analysis to match users to transactions, transactions to means of payment and to trace flow of money and goods

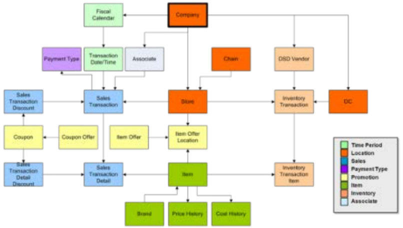
$400-600 is lost on each fraudulent Smartphone drop shipment in equipment expense. Example of impact: 9.4M Smartphones shipped in a quarter. Estimated 1% fraud rate = $30M to 56M lost in Qtr.

Ideas for proven analytical value in each industry

Information

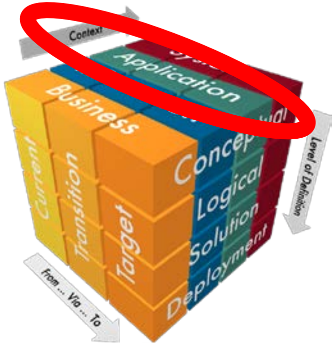


Logical Data Model

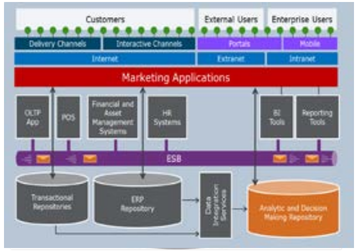


A starting point for data models in each industry

Application

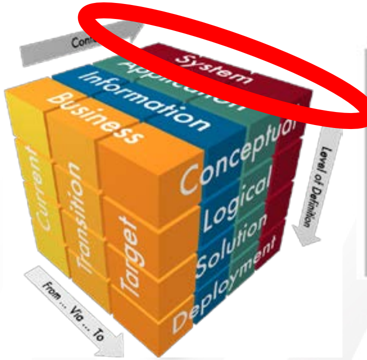


Application Suite

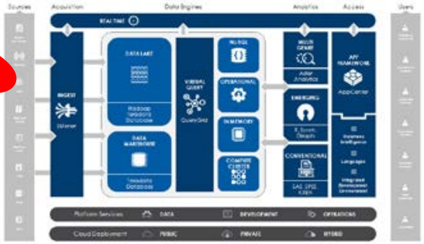


Identifies analytical applications to meet specific business needs

System

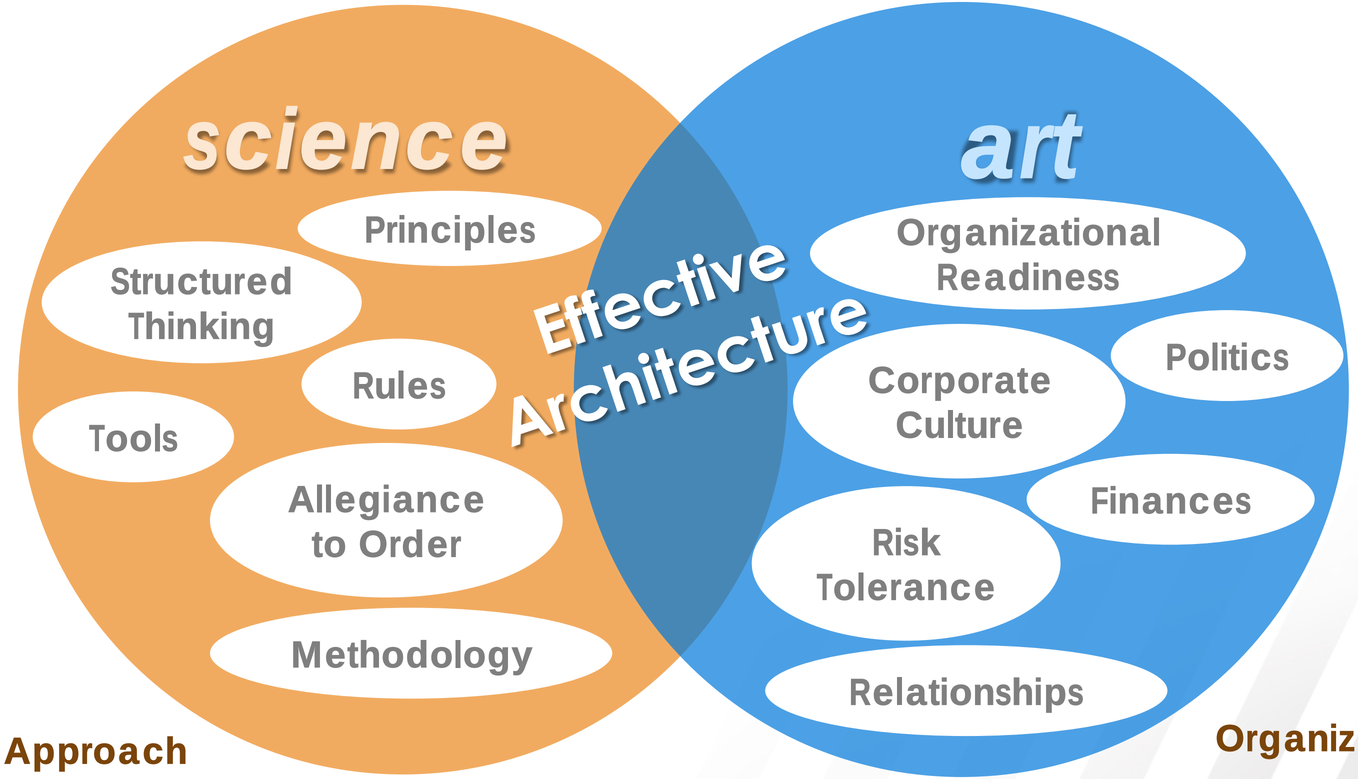


Unified Data Architecture



Identifies the appropriate role for each type of platform

Reminder: Art *and* Science



Architecture Approach

Organizational Filter

Summary

Recognize the Mistakes

Does Not Work		Best Practice
Technology focused	1	Business driven
Accidental evolution	2	Methodical development
Art vs. Science	3	Art *and* Science
Contentious debates	4	Principle-based decisions
Reinvention of ideas	5	Reusable patterns
Pretty pictures	6	Documented blueprints

Recommendations

- **Adopt a structured architecture framework**
  - Always drive from business needs
  - Methodically evolve using transitional states
- **Adhere to architecture principles and advocated positions**
  - Reduce contentious debate
  - Maintain record of deviations and design rationale
- **Develop a library of design patterns and implementation alternatives**
  - Avoid reinvention and inefficiencies
- **Use reference architectures as a starting point**
  - Shamelessly borrow, customize and document
  - Liberally reference as blueprints for construction
- **Balance art with science**
  - Recognize your environment is unique and one size does not fit all

Thank You!

Rate This Session # 1021

with the Teradata Analytics Universe Mobile App

Follow Me

LinkedIn @ **Eddie Sayer**

Questions/Comments

Email: **Eddie.Sayer@Teradata.com**