Avancier Methods (AM)
Data architecture diagrams

It is illegal to copy, share or show this document
(or other document published at http://avancier.co.uk)
without the written permission of the copyright holder
DATA ARCHITECTURE

Motivations and constraints
- Aims
  - Goals
  - Objectives
  - Requirements
- Directives
  - Principles
  - Policies
  - Rules
- Compliance
  - Regulations
  - Standards (SIB)
  - Design Patterns (RM)
- Management
  - Time
  - Budget
  - Resources

Passive structures
Things that are acted in or on

Behaviors
Things happening over time that access or change the state of business systems

Logical active structures
Specifications of things that act

Physical active structures
Things that act

Logical Data Component

Business Data Entity

Logical Data Entity

Data entity/Business function Matrix

Conceptual data Diagram

Logical Diagram

Data dissemination Diagram

Data entity/Data component catalogue

Function

Processes
TOGAF says: Data Entity/Business Function Matrix

- To depict the relationship between data entities and business functions within the enterprise.
- Business functions are supported by business services with explicitly defined boundaries and will be supported and realized by business processes.
- The mapping enables the following to take place:
  - Assign ownership of data entities to organizations
  - Understand the data and information exchange requirements business services
  - Support the gap analysis and determine whether any data entities are missing and need to be created
  - Define application of origin, application of record, and application of reference for data entities
  - Enable development of data governance programs across the enterprise (establish data steward, develop data standards pertinent to the business function, etc.)
One of the most traditional EA artifacts

- Read Function for Process, and note clustering on “Create”
ArchiMate???
TOGAF says: Conceptual Data Diagram

- The key purpose is to depict the relationships between critical data entities within the enterprise - to address the concerns of business stakeholders. (TOGAF)

- Poor match in ArchiMate
- An information structure view – business/conceptual level
- (Aargh! Aggregates and compositions instead of properly named associations!)

Difficult to draw a data model for a whole enterprise
A catalogue of core business data entities is more practical
TOGAF says: Data Entity/Data Component Catalog

- to identify and maintain a list of all the data use across the enterprise, including data entities and also the data components where data entities are stored.
- supports the definition and application of information management and data governance policies and also encourages effective data sharing and re-use.
ArchiMate???
TOGAF says: Data Dissemination Diagram

- ... to show the relationship between data entity, business service, and application components.
- It shows how the logical entities are to be physically realized by application components.
- It allows effective sizing to be carried out and the IT footprint to be refined.
- Moreover, by assigning business value to data, an indication of the business criticality of application components can be gained.
- It may show data replication and application ownership of the master reference for data...
- It can show two copies and the master-copy relationship between them.
- It can include services; that is, services encapsulate data and they reside in an application, or services that reside on an application and access data encapsulated within the application.

<table>
<thead>
<tr>
<th>Data entity</th>
<th>Application</th>
<th>CRM</th>
<th>ERP</th>
<th>Billing</th>
<th>Data warehouse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>Master</td>
<td>Copy</td>
<td>Copy</td>
<td>Copy</td>
<td>Copy</td>
</tr>
<tr>
<td>Order</td>
<td>Master (1)</td>
<td>Copy</td>
<td>Master (2)</td>
<td>Copy</td>
<td></td>
</tr>
<tr>
<td>Invoice</td>
<td>Master</td>
<td>Copy</td>
<td>Master</td>
<td>Copy</td>
<td></td>
</tr>
</tbody>
</table>

(1) until Order Closed
(2) after Order Closed.
Partial match in ArchiMate

▶ Hand made
TOGAF says: Logical data diagram

- ... to show logical views of the relationships between critical data entities within the enterprise - to address the concerns of Application developers and Database designers. (TOGAF)

- Poor match in ArchiMate
- An information structure view – logical level
- (Aargh! Aggregates and compositions
- instead of properly named associations!)
In models of persistent business entities, the passage of time tends to turn

- Subtypes into roles
- Aggregations into associations
- 1-1 associations into 1-N
- 1-N associations into N-to-N with link entities

See for example this Salesforce.com model
Salesforce.com has a structural model of a typical style.

Two dozen 1-to-N associations
One 1-to-0 or 1 association
TOG says: Data security diagram

- to depict which actor (person, organization, or system) can access which enterprise data.
- can be shown in a matrix between objects or as a mapping.

<table>
<thead>
<tr>
<th>Role/actor</th>
<th>Data entity</th>
<th>Customer</th>
<th>Product</th>
<th>Invoice</th>
<th>Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td>HR manager</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Can read</td>
</tr>
<tr>
<td>Product manager</td>
<td></td>
<td></td>
<td>Can read</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Salesman</td>
<td>Can read</td>
<td>Can read</td>
<td>Can read</td>
<td>Can read</td>
<td></td>
</tr>
<tr>
<td>1st line support</td>
<td>Can read</td>
<td>Can read</td>
<td>Can read</td>
<td>Can read</td>
<td></td>
</tr>
<tr>
<td>Fulfilment agent</td>
<td>Can read</td>
<td></td>
<td>Can read</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- can be used to demonstrate compliance with data privacy laws and other applicable regulations (HIPAA, SOX, etc).
- should also consider any trust implications where an enterprise’s partners or other parties may have access to the company’s systems.
# Data quality scoring scheme

Score each data item/group/store thus

<table>
<thead>
<tr>
<th>Confidentiality</th>
<th>Integrity</th>
<th>Availability</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unauthorized use or disclosure</td>
<td>Data inaccuracy, incompleteness or unauthorized modification</td>
<td>Unavailable information</td>
<td></td>
</tr>
<tr>
<td>Severely impairs business operations, make a segment of the company unable to function or cause high monetary loss.</td>
<td>Causes failures of operations, revenue loss, wrong decisions to be made, loss in productivity or loss of customer confidence or market share.</td>
<td>Impairs business operations, affects customer service or makes it impossible to process revenues.</td>
<td>High</td>
</tr>
<tr>
<td>Does not severely affect operations or does not result in high monetary loss.</td>
<td>Makes it impossible to make some decisions, but the problem is not difficult to detect and correct, and does not severely impact business operations.</td>
<td>Causes productivity loss, but does not interrupt customer service or revenue generation.</td>
<td>Moderate</td>
</tr>
<tr>
<td>Does not affect operations or result in significant monetary loss.</td>
<td>Does not disable business operations, since alternative validations of the information make it possible to continue</td>
<td>Does not severely impact business operations.</td>
<td>Low</td>
</tr>
</tbody>
</table>
TOG says: Data Lifecycle diagram

an essential part of managing business data through its lifecycle from conception to disposal within the constraints of the business process.

Each change in state is represented on the diagram which may include the event or rules that trigger that change in state.
Data Lifecycle diagram v Process Flow diagram

Notice how one reflects each other

Data Lifecycle

Recorded → Received
Record

App Form Sent → Send
Send App Form

App Form Received
App Form Received

App Form Returned
Receive App Form

Send for Approval → Approve
Send for Approval

Approved

Live

Event

Process Flow

Record Enquiry → Send Application Form → Receive Application Form
[No errors] → Send For Approval
[Errors] → Return for Correction

Check Credit Rating → Check Nationality

Approve Application → Activate Account
TOG says: Data Migration diagram (not a great example!)

- to show the flow of data from the source to the target applications.
- provide a visual representation of the spread of sources/targets and serve as a tool for data auditing and establishing traceability.
- For example, just an overall layout of migration landscape or could go into individual application metadata element level of detail.
## Four notations

- Barker’s is good

<table>
<thead>
<tr>
<th>Notation</th>
<th>Information Engineering</th>
<th>Barker Notation</th>
<th>IDEF1X</th>
<th>UML</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Multiplicities:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Zero or one</td>
<td>0+</td>
<td>—</td>
<td>—</td>
<td>0..1</td>
</tr>
<tr>
<td>- One only</td>
<td>1</td>
<td>—</td>
<td>—</td>
<td>1</td>
</tr>
<tr>
<td>- Zero or more</td>
<td>0..*</td>
<td>—</td>
<td>—</td>
<td>0..*</td>
</tr>
<tr>
<td>- One or more</td>
<td>1..*</td>
<td>—</td>
<td>P</td>
<td>1..*</td>
</tr>
<tr>
<td>- Specific range</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>3.7</td>
</tr>
<tr>
<td><strong>Attributes:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Names</td>
<td>NA</td>
<td>Attribute Name: Type</td>
<td>attributeName: Type</td>
<td>NA</td>
</tr>
<tr>
<td>Primary key/unique identifier</td>
<td>NA</td>
<td># Attribute Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Foreign key</td>
<td>NA</td>
<td>NA</td>
<td>attribute-name (FK)</td>
<td>NA</td>
</tr>
<tr>
<td><strong>Associations:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Labels</td>
<td>Customer</td>
<td>owned by</td>
<td>Account</td>
<td>Customer</td>
</tr>
<tr>
<td>Entity roles</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Subtyping</td>
<td>Sub Type</td>
<td>is a</td>
<td>Super Type</td>
<td>Sub Type</td>
</tr>
<tr>
<td>Aggregation</td>
<td>Part</td>
<td>is part of</td>
<td>Whole</td>
<td>Part</td>
</tr>
<tr>
<td>Composition</td>
<td>Part</td>
<td>is part of</td>
<td>Whole</td>
<td>Part</td>
</tr>
<tr>
<td>Or Constraint</td>
<td>Person</td>
<td>Customer</td>
<td>Employee</td>
<td>N/A</td>
</tr>
<tr>
<td>Exclusive Or (XOR) Constraint</td>
<td>Item</td>
<td>Product Service</td>
<td>Item</td>
<td>Product Service</td>
</tr>
</tbody>
</table>
Logical Data diagram notation: Barker and IDEF styles

- **Barker style**
  - Woman
  - Child
  - Coach Trip
  - Passenger
  - Choir Performance
  - Singer Performance

  - Optional
  - Multiple

- **IDEF style**
  - Department
  - Project
  - Relationship Name
  - Entity Name

  - Mandatory Existence
  - Optional Existence

  - One
  - Many
  - Manages
Footnote: naive uses of aggregation

- Optional
- Multiple

Barker Entity Relationship Diagram

- Woman
- Child
- Coach Trip
- Passenger
- Choir Performance
- Singer Performance

- May bear
- Takes
- Needs
- Was born to

Archimate Information Structure

- Woman
- Child
- Coach Trip
- Passenger
- Choir Performance
- Singer Performance

- Is a group of

Not all one-to-many associations are aggregates
## Interface catalog

### A Data Flow Catalogue (Interface Catalogue in TOGAF)

<table>
<thead>
<tr>
<th>Functional attributes</th>
<th>Flow name</th>
<th>Enquiry</th>
<th>Response</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger</td>
<td></td>
<td>Enquiry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>Customer</td>
<td>Sales</td>
<td>Customer</td>
<td></td>
</tr>
<tr>
<td>Destination</td>
<td>Sales</td>
<td>Customer</td>
<td>Sales</td>
<td></td>
</tr>
<tr>
<td>Information</td>
<td>Unstructured</td>
<td>Unstructured</td>
<td></td>
<td>Order details (tbd)</td>
</tr>
<tr>
<td>Frequency</td>
<td>1,000/day</td>
<td>1,000/day</td>
<td>30/day</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td></td>
<td></td>
<td></td>
<td>500K</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Integrity</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Availability</td>
<td>24/7</td>
<td>09.00-18.00</td>
<td>24/7</td>
<td></td>
</tr>
<tr>
<td>Transport mechanisms</td>
<td>Technology</td>
<td>Telephone</td>
<td>Web</td>
<td>HTTPS</td>
</tr>
<tr>
<td>Protocol</td>
<td>HTTP</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Data Access Path (short-term process) diagram

- Which data entities are accessed by a process?
- Useful in analysing the efficiency of a process.
- Where an automated service accesses data in a data store, then the logic of the process can be shown in an access path diagram.
- A node in the diagram is a persistent entity.
- Lines between nodes show the path a process takes through the structure of persistent entities.
- (You might manipulate a UML interaction diagram to define an access path.)
Data Flow Structure (regular expression) diagram

- What is the content and hierarchical structure of a data flow or message?
- What data structure must be implemented in an XML or other data definition schema?
Message with iterated element

Name SELECT
   Single Name
Name OR
| Full name SEQUENCE
| First Name
| Middle Names ITERATION
|   Middle Name
| Middle Names END
| Last Name
| Full name END
Name END

xsd:choice
   xsd:element name="SingleName" type="Text" minOccurs="1" maxOccurs="1" /
   xsd:sequence
   | xsd:element name="FirstName" type="Text" minOccurs="1" maxOccurs="1" /
   | xsd:element name="MiddleName" type="Text" minOccurs="0" maxOccurs="unbounded"/
   | xsd:element name="LastName" type="Text" minOccurs="1" maxOccurs="1" /
   /xsd:sequence
/xsd:choice
Logical Data diagram – Relational style

- Entities
- Attributes &
- Relationships

- Usually defining the content of one persistent data store
- And designed to enable I/O data flows
Logical Data diagram: MODAF style

**Example – OV-7 Extract for Generic Stores System**

**Description:**
Describes the structure and relationships between operational data elements

**Alternative Views:**
UML Class Diagram
Logical Data diagram: ORM style

- This specialist notation expresses detailed business rules that constrain the values of data items/attributes.
- Example from Wikipedia