

Metodologias de modelagem em empresas (frameworks)

(Material retirado da página www.numa.gov.br)

Serão rapidamente descritos os seguintes *frameworks* de modelagem: **ISO, CEN ENV 40003, CIMOSA, IDEFX/SADT e ARIS.**

Existem ainda muitos outros importantes *frameworks* para modelagem de empresas. O modelo conhecido como GRAI/GIM foi desenvolvido na universidade de Bordeaux como um método utilizado para analisar e sistemas de automação da manufatura e utiliza o método MERISE para o sistema de informação e o método IDEF0 para o sistema físico. O PERA (Purdue Enterprise Reference Architecture) foi desenvolvido na Universidade de Purdue em 1989, como base para o trabalho do grupo de CIM liderado pelo Prof. Williams, do Laboratório de Controle Industrial Aplicado. Outro *framework* é o Generalized Enterprise Reference Architecture and Methodology (GERAM) desenvolvido na Estônia em 1990, que teve por finalidade gerar um *framework* de referência macro para integrar conceitos de outros *frameworks* gerando um padrão que pudesse vir a ser mundialmente aceito. O resultado foi uma combinação do CIMOSA, GIM e PERA. Outros de menor importância, são IMPACS, baseado no GIM, e o Integrated Enterprise Modelling (IEM), baseado no conceito de orientação à objetos.

ISO:

Metodologia desenvolvida pelo International Standard Organization (ISO). O *framework*, denominado ISO Reference Model consiste de:

Um modelo de contexto que identifica as macro-funções da empresa (produção, finanças, marketing, etc.) e define o relacionamento entre elas. Este modelo é baseado no modelo de plantas de manufatura desenvolvido pelo então National Bureau Standard dos Estados Unidos, hoje denominado NIST (National Institute of Standards and Technology);

Um modelo do chão de fábrica (Shop Floor Production Model – SFPM), o qual representa um modelo de atividade genérica (GAM – Generic Activity Model) que descreve as atividades e fluxos (de materiais, informação e recursos) entre as atividades.

IDEFX/SADT:

O *framework* de modelagem IDEFX (integration definition for function modeling), ou seja, a família de modelos IDEF, é a ferramenta de modelagem de empresas mais utilizada na prática graças a sua enorme simplicidade, chegando mesmo a tornar-se um padrão para este fim. Já o SADT (structured analysis and design technique) não chega a ser uma arquitetura, e poderia ser melhor caracterizado como um formalismo de modelagem. Ambos são apresentados juntos, pois tem origens e vidas tão próximas que seus nomes são até mesmo fundidos na prática.

O SADT foi desenvolvido por Douglas T. Ross no MIT no final dos anos 60, a partir de um projeto de uma linguagem estruturada para programação de máquinas-ferramenta no MIT. Este formalismo trazia algumas características revolucionárias que auxiliaram sobremaneira a descrição e desenvolvimento dos sistemas de software complexos que começavam a aparecer. São elas a decomposição funcional e o fato de serem baseados numa representação esquemática simples. Estas características disseminaram a aplicação deste formalismo e também o seu desenvolvimento em direção a abordagens estruturadas completas destinadas a diferentes aplicações, principalmente para a análise de sistemas.

Já o IDEF foi desenvolvido durante o projeto ICAM (Integrated Computer Aided Manufacturing) realizado pela Força Aérea Americana na década de 80. Esta arquitetura é composta de três diferentes técnicas:

- 1 - IDEF0: usada para produzir o modelo de funções e é baseada no formalismo utilizado pelo SADT. Um modelo de funções é uma representação estruturada das funções, atividades e processos da empresa.
- 2 - IDEF1: usada para produzir o modelo de informações. Baseada numa versão inicial do modelo entidade-relacionamento.
- 3 - IDEF2: usada para produzir o modelo dinâmico. O modelo dinâmico representa a variação das características do sistema ao longo do tempo.

O SADT e IDEF0 são baseados num diagrama conhecido como “ativigrama”. Este diagrama é composto por “caixas” que representam as atividades. Estas caixas são ligadas por linhas e dispostas visando a formar uma ordem de condução das atividades. As linhas que chegam e saem na lateral das caixas representam inputs e outputs de informação. As que chegam no topo são controles e embaixo mecanismos. Com mais algumas poucas regras além das aqui apresentadas, tem-se todo o formalismo necessário para descrever estes modelos.

Por sua simplicidade e facilidade de uso (existência de ferramentas computacionais para auxílio da modelagem) o SADT/IDEX é uma das principais ferramentas de modelagem de empresas. Também traz consigo grandes desvantagens. Segundo VERNADAT(1996), sua semântica é imprecisa, o comportamento dinâmico do sistema não é claro, não manipula fluxos mas simplesmente dependências entre as atividades, a modelagem de informações é limitada e existe uma baixa consistência devida a não integração entre os diferentes modelos. Já CANTAMESSA & PAOLUCCI (1998) aludem principalmente a necessidade de grande cuidado no desenvolvimento do modelo, alta possibilidade de desvios e grande tempo gasto na modelagem devido à necessidade excessiva de revisões para garantir a consistência do modelo.

Assim, VERNADAT(1996) sugere que esta arquitetura seria mais apropriada para modelos pequenos e restritos às visões de maior nível de abstração, portanto, com menor complexidade. Já CANTAMESSA & PAOLUCCI(1998) apresentam propostas de diversos autores para a alteração nesta arquitetura visando a superação destes obstáculos e propõem um novo caminho complementando o formalismo IDEF0/SADT com outros modelos.

CEN ENV 40003:

Este *framework* foi desenvolvido pelo Comitê Europeu de Padronização (European Committee for Standardization) dentro do grupo de trabalho Arquitetura de Sistemas CIM. Sua proposta visa padronizar as atividades na área de modelagem de empresas, dando suporte a Manufatura Integrada por Computador.

Este *framework* define três dimensões e procura padronizar os modelos existentes dentro delas. As dimensões são:

1. Dimensão da generalização: representa a generalização das entidades descritas num elevado nível de abstração. Parte-se de um nível genérico, contendo os elementos básicos como componentes, restrições, regras, termos, serviços e funções a serem modeladas para um nível particular, que especifica os elementos para a empresa específica que está sendo modelada.
2. Dimensão dos Modelos: especifica o nível de abstração dos modelos. É dividido em três etapas. A primeira é o nível de requisitos onde são definidas as operações da empresa e as terminologias, informações e recursos, sem qualquer referência às opções de implementação ou decisões tomadas. A segunda é o nível de projeto, onde são definidas como são realizadas as operações na empresa, as suas entidades de informações, recursos e estrutura organizacional usados para atingir os requisitos especificados na etapa anterior. Por fim, a etapa das especificações que descreve as formas e ou as regras para serem utilizadas na execução das operações definidas.
3. Dimensão das Visões: define o ponto de vista que o modelo descreve a organização. A primeira é a visão das funções, que permite a descrição hierárquica de todas as funções da empresa. O segundo é a visão da informação, que permite uma descrição estruturada dos objetos de informação e dados da empresa. Em seguida a visão dos recursos que descreve o conjunto de recursos necessários para a execução das operações da empresa. Por fim, a visão da organização, que descreve a estrutura organizacional.

CIMOSA:

O CIMOSA, European Open System Architecture for CIM, foi desenvolvido de 1986 até 1994. É um *framework* completo, que compreende tanto a modelagem propriamente dita como a metodologia de implantação CIM. É composto de uma definição geral do escopo e implantação de CIM, guias para a implantação e um *framework* que define termos e padrões para modelagem e implantação.

O *framework* de modelagem consiste de um conjunto de níveis e abstrações parecido com o do ENV descrito anteriormente. Ele separa claramente a denominada arquitetura de referência da arquitetura particular. A primeira refere-se a uma descrição genérica que serve a um conjunto geral de empresas e corresponde aos dois primeiros níveis de instanciação. A arquitetura particular, por sua vez descreve uma empresa específica.

ARIS (ARchitecture for Integrated Systems):

ARIS é a sigla para ARchitecture of Integrated Information Systems e foi desenvolvida pelo Prof. Scheer na Alemanha entre 1992 e 1994, sendo um *framework* de modelagem que enfatiza os aspectos organizacionais da empresa e aqueles ligados à engenharia de software. No ARIS são definidos quatro visões e três níveis de modelagem.

As visões são do ARIS são:

1. Visão Funcional: permite construir modelos que definem de maneira hierárquica todas as funções da empresa começando das mais macro e decompondo-as até o nível de detalhe que permitem especificar funções de programação específicas dentro de aplicativos de software;
2. Visão dos Dados: é utilizada para definir os modelos de dados partindo das definições das informações mais complexas (relatórios ou conjunto de informações) passando pelo modelo de dados e seus relacionamentos, a definição de esquemas e definição da própria base de dados;

3. Visão Organizacional: permite especificar e detalhar a estrutura organizacional da empresa desde a definição das divisões e unidades de negócios, a estrutura de cargos e seus ocupantes, até a estrutura física com os equipamentos, com ênfase especial à estrutura de informática na medida que há métodos específicos para a modelagem da rede de computadores da empresa;

4. Visão de Controle: é a visão que permite relacionar as três visões anteriores. Nesta visão há métodos de modelagem específicos para definir a relação entre funções e dados; funções e organização; organização e dados e, principalmente, capazes de integrar as três funções utilizando-se principalmente do conceito de evento.

Em cada uma das visões há níveis, que lembram os níveis do CIMOSA / CEN ENV 40003, que são: o da definição dos requisitos mais gerais dos termos do sistema; o nível de projeto, onde se define as formas de encontrar os requisitos e nível de implementação onde se define detalhadamente como o determinado elemento ocorre dentro da organização.

No nível da especificação dos requisitos e visão da organização são descritas as divisões do negócio da empresa (unidades, mini-fábricas, setores, etc.) e a hierarquia organizacional com cargos, responsabilidades e nomes de pessoas. Ainda na visão organização, porém no nível de projeto, é especificada a topologia da rede. No nível de implementação, ocorre a descrição da rede, modelando cada dispositivo de cada ponto, a partir de diversas características e também existe um método para descrever os detalhes do chão de fábrica, como máquinas e equipamentos de movimentação.

Na visão de funções o método Árvore de Funções (Function Tree) permite descrever a hierarquia de funções da empresa até um nível de atividades específicas. O Objective Diagram permite modelar os objetivos da empresa em termos gerais.

Os demais diagramas são específicos para algumas aplicações, o primeiro para o software de integração R/3 da SAP e o outro que descreve as tarefas ou funções da empresa conforme o modelo de integração de empresa Y-CIM proposto por Scheer, de uma forma altamente agregada e concisa.

No nível de projeto pode-se especificar, no Application System Type Diagram cada tipo de aplicação, por exemplo CAD, planilha, processador de texto, entre outros. E, no nível de implementação, pode-se especificar no Application System Diagram cada licença de cada aplicação da empresa, além de existirem outros diagramas onde é possível especificar funções de aplicações desenvolvidas pela própria empresa.

Na visão de dados existe uma grande quantidade de diagramas e notações possíveis, sendo o principal o MER clássico e o MER estendido. No nível de projeto, o Relation Diagram permite especificar o modelo relacional, sobre as quais a base de dados será construída.

No nível mais baixo, o de implementação, é possível especificar as tabelas. Na visão de controle, existe uma quantidade muito grande de técnicas específicas para relacionar diferentes visões. O diagrama mais importante é o EPC (Event Drive Process Chain) o qual introduz o conceito de evento como uma alteração significativa do status de um objeto. Com este conceito, pode-se especificar o fluxo de todas as funções intercaladas por eventos que resultam de sua ação e que as disparam.

Este diagrama é a espinha dorsal a partir da qual é possível relacionar todos os elementos, sua ocorrência (especificadas pelos eventos) e permite agregar quem é o responsável por fazê-la (visão organizacional) e as informações e dados necessários (visão dos dados). O eEPC é o EPC estendido com mais notações específicas e o Information Flow Diagram permite especificar o fluxo de dados. O Access Diagram mostra o fluxo de dados entre aplicações no nível de projeto, mostrando como os diferentes tipos de aplicações se comunicam. O Modelling System Interfaces mostra o relacionamento entre as Aplicações de Software, os Módulos de Software (partes de programas) e as Funções de Software, ou seja, integra os níveis de projeto e implementação da visão função.

No nível de implantação o Access Diagram pode ser utilizado para especificar a ligação entre dados, funções e organização no nível de implementação (ligando, respectivamente campos de tabelas com funções de programas e a alocação, ou sua localização física). Deve-se notar que estes são os métodos mais importantes, havendo a existência de muitos outros.

Para o auxílio na manipulação de todos estes métodos e visões foi desenvolvida uma ferramenta computacional pela empresa IDS, denominada ARIS Toolset, que tem interface gráfica para o sistema windows e disponibiliza uma grande quantidade de métodos para cada visão e nível do *framework* ARIS. Ela permite o desenvolvimento e gerenciamento de todos os modelos de maneira integrada. Desta forma, uma entidade tem uma existência única mesmo que com ocorrência em vários modelos ao mesmo tempo. Isto facilita o gerenciamento e aumenta a consistência tal que a edição da entidade em um modelo específico se reflete em todos os demais.