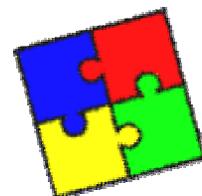
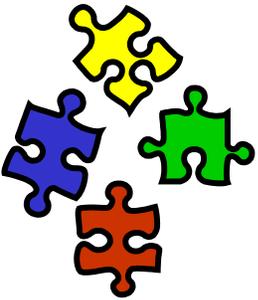


Visão Conceitual

Resumo da Representação

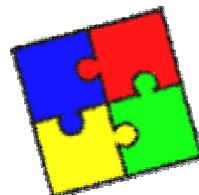
Artefato	Representação
Configuração Conceitual	Diagrama de Classes UML
Protocolo de Porta ou Papel	Diagrama de Statechart UML ou diagrama de Sequência
Comportamento do Componente ou do Conector	Diagrama de Statechart UML ou descrição em ling. natural
Interações entre Componentes	Diagrama de Sequência UML

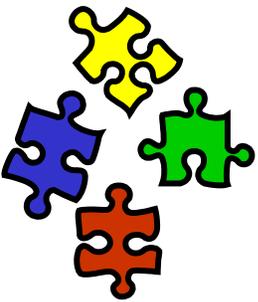




Visão de Módulo

- Componentes e Conectores que formam a visão conceitual são mapeados em subsistemas e módulos
- Diferença da Visão Conceitual para a Visão de Módulo
 - Cada uma torna explícito diferentes aspectos
 - Na visão conceitual os relacionamentos funcionais devem estar explícitos
 - A visão de módulo torna explícito como a funcionalidade é mapeada para a implementação

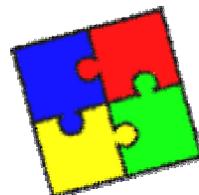


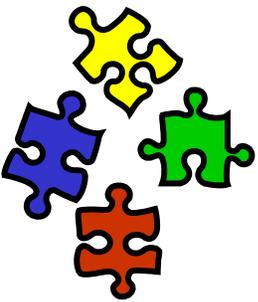


Relacionamento

Visão Conceitual x Visão de Módulo

- A Visão Conceitual e a Visão de Módulo são baseados em dois diferentes modelos:
 - Visão Conceitual: componentes implementam a funcionalidade da aplicação e interagem via conectores.
 - Visão de Módulo: toda a funcionalidade da aplicação e a comunicação devem ser mapeada em módulos

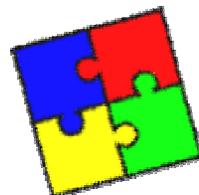


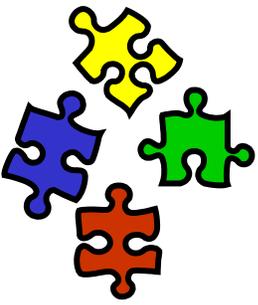


Relacionamento

Visão Conceitual x Visão de Módulo

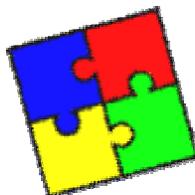
- Visão Conceitual
 - Dois componentes comunicam-se através de um conector com semântica *call/return*
- Visão de Módulo
 - Representa em módulos os elementos da implementação que poderá ter duas variações:
 - Os dois componentes na mesma máquina: o conector implementa uma chamada de procedimento local
 - Os dois componentes em máquinas diferentes: o conector implementa chamada remota de procedimento (RPC)

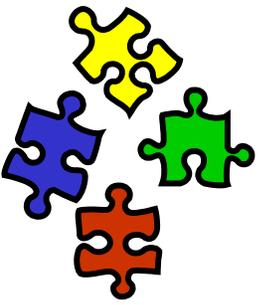




Visão de Módulo

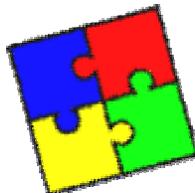
- Definir módulos que possuem interface com os serviços oferecidos (*provided*) e requisitados (*required*)
- Módulos interagem invocando serviços declarados na sua interface *required*
- Módulos não possuem implementação associada

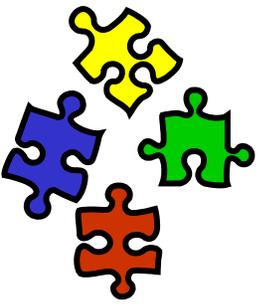




Visão de Módulo

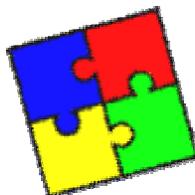
- Não define uma configuração
 - Define os módulos e seus relacionamentos mas não como eles são combinados em um produto particular (as visões conceitual e de execução define a configuração)
- Resultados da Visão de Módulo:
 - Módulos, Subsistemas e Camadas

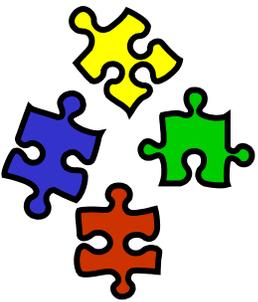




Visão de Módulo Subsistema

- Usualmente corresponde ao componente conceitual de mais alto nível (aquele que é decomposto em outros componentes e conectores)
- Contém outros subsistemas e módulos

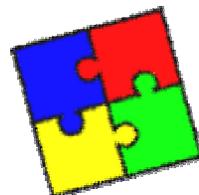


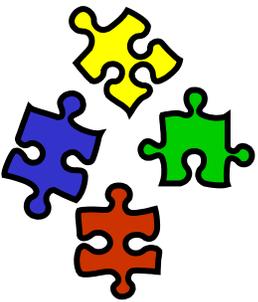


Visão de Módulo

Módulo

- Pode corresponder a um único elemento conceitual (componente, porta, conector ou papéis) ou a um conjunto deles.
- Podem ser decompostos em outros módulos
- Encapsula dados e operações para prover um serviço

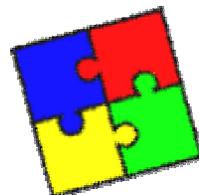


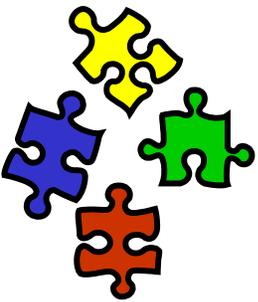


Visão de Módulo

Módulo

- Serviços:
 - Os serviços oferecidos por um módulo são definidos pelas interfaces que ele oferece (provided)
 - Os serviços que um módulo precisa de outro módulo para realizar suas funções são os serviços requisitados e são definidos pelas interfaces que ele solicita (required)
- Interação com outros apenas através das interfaces

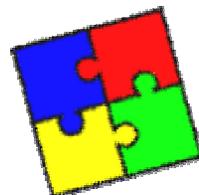


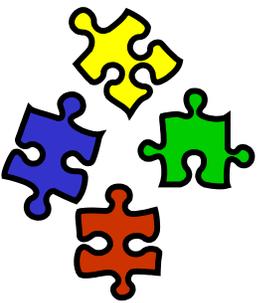


Visão de Módulo

Camadas

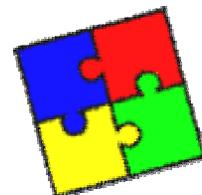
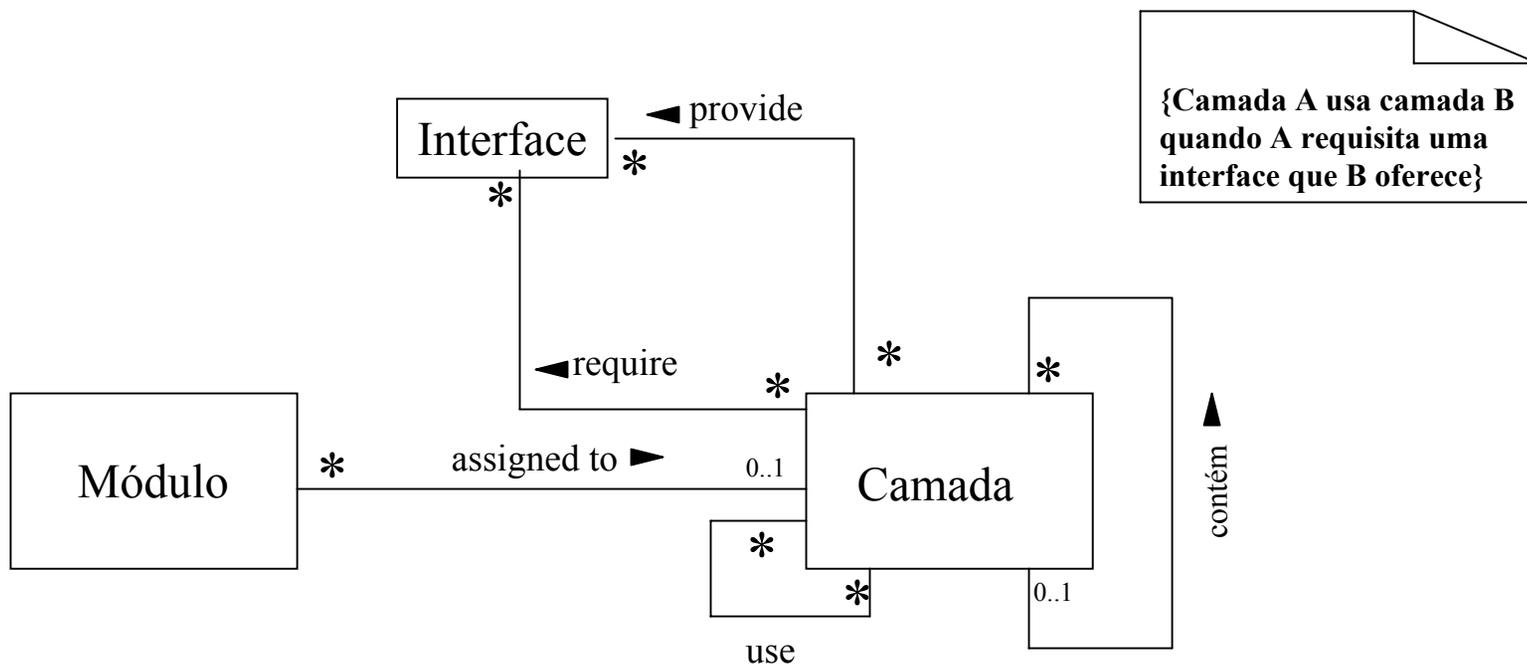
- Camadas organizam módulos em uma hierarquia
- Quando um módulo é associado a uma camada, ele pode usar outros módulos desta camada
- Quando um módulo necessita usar serviços de um módulo de outra camada, a interface solicita e requisitada pelos módulos devem ser também requisitas e solicitadas pelas camadas.
- Camadas podem ter subcamadas

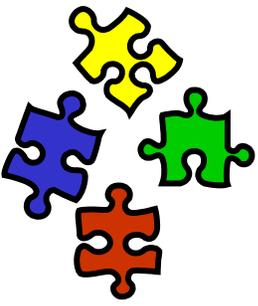




Visão de Módulo

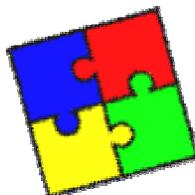
Meta-modelo para Camadas

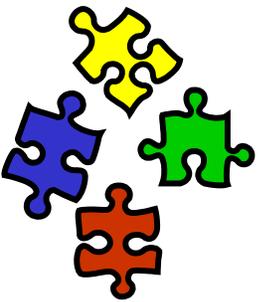




Visão de Módulo Camadas

- Utilidade
 - Reduzir complexidade
 - Permitir reuso atribuindo serviços comuns a camadas de serviços da aplicação
 - Oferecer independência entre partes do sistema evitando que a mudança em uma parte afete o sistema inteiro





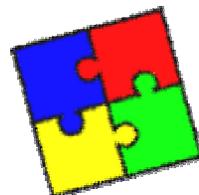
Visão de Módulo

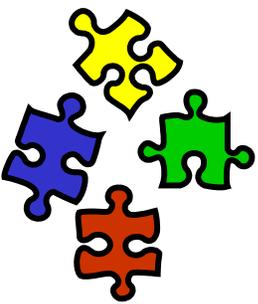
Camadas

- Como definir Camadas
 - Top-down: determine as camadas baseadas na experiência. Atribua os módulos às camadas. (As camadas servem de guia para definição dos módulos)
 - Bottom-up: determine os módulos, suas responsabilidades e dependências. Em seguida determine as camadas.

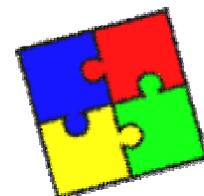
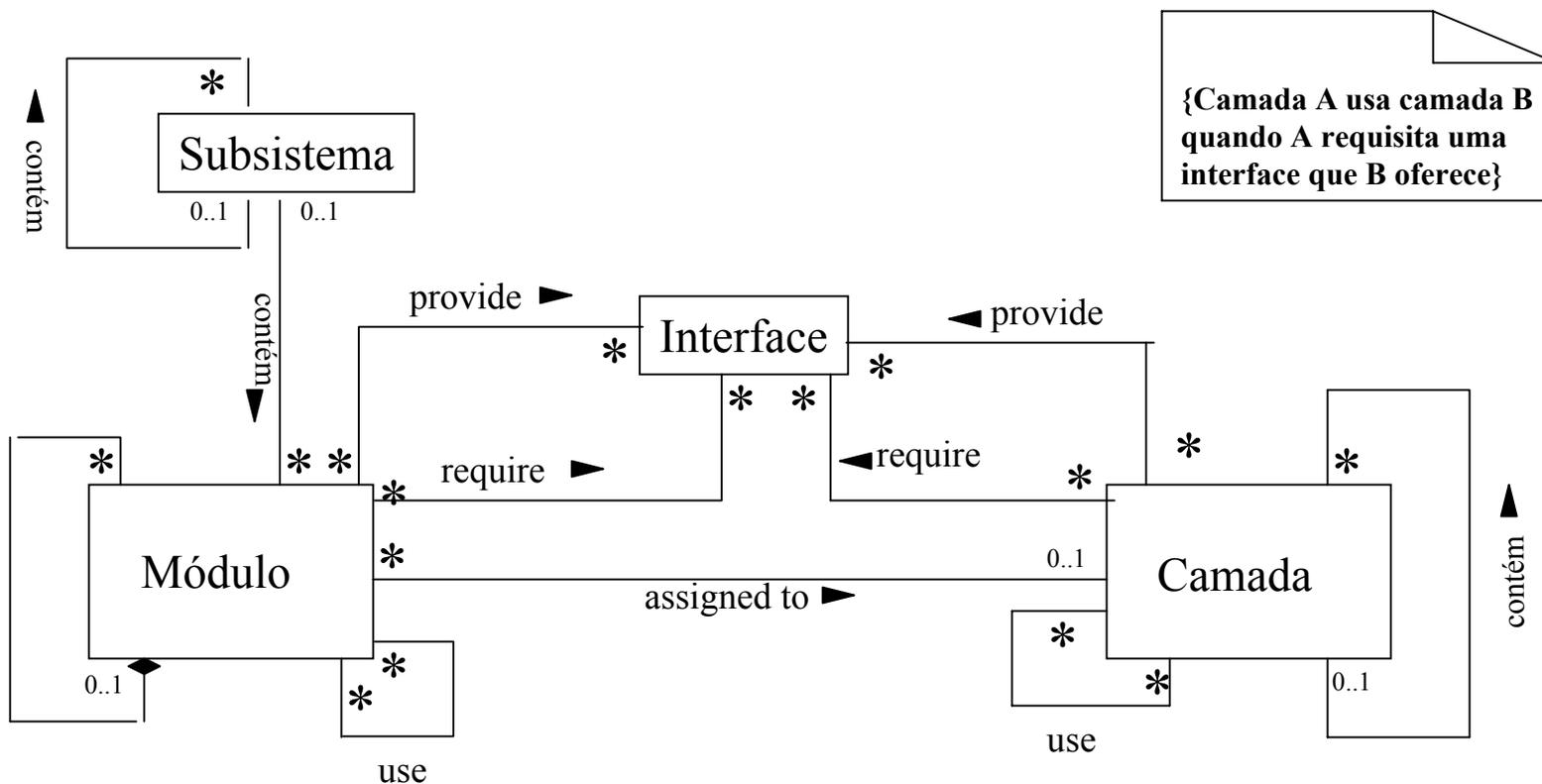
Estratégia Top-Down + Bottom-Up: divida em camadas genéricas (aplicação, interface com usuário, serviços do sistema).

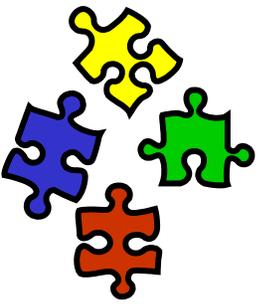
Quando definir os módulos refine o modelo em camadas adicionando camadas que tratem da funcionalidade específica da aplicação





Visão de Módulo Meta-Modelo

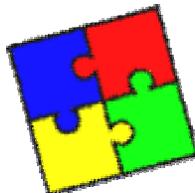


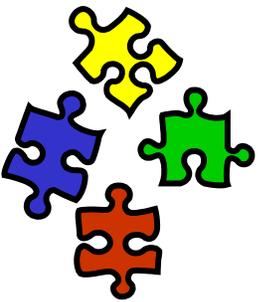


Visão de Módulo

Tarefas

- Mapear os elementos da visão conceitual em CAMADAS, SUBSISTEMAS e MÓDULOS
- Atribuir aos módulos uma responsabilidade e determinar sua composição e seus relacionamentos
- Pode ser necessário adicionar módulos de suporte que não tem correspondente na visão conceitual
- Descrever as interfaces para cada um dos módulos e camadas

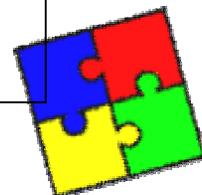


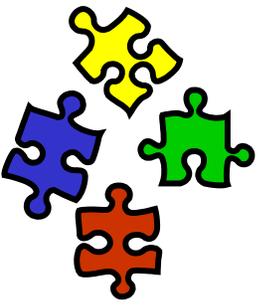


Visão de Módulo

Resumo da Representação

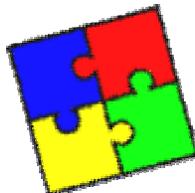
Artefato	Representação
Correspondência Conceitual-Módulo	Tabela
Subsistemas e Módulos	Diagrama de Classes UML
Dependências dos Módulos	Diagrama de Classes UML
Dependências de Camadas, Atribuição de Módulos a Camadas	Diagrama de Classes UML
Resumo das relações entre Módulos	Tabela

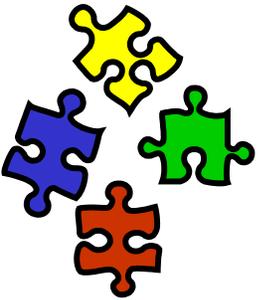




Visão de Execução

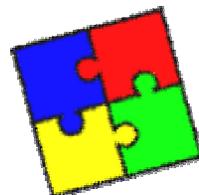
- Descreve a estrutura do sistema em termos de elementos da plataforma de execução (p.ex. tarefas do SO, processos, threads).
- Recebe como entrada o Modelo Conceitual e o Modelo de Módulos
- É necessário conhecer a plataforma de hardware (todos os componentes de hardware disponíveis) e a plataforma de software (os softwares que estão entre o hardware e o sistema projetado – SO, software de rede, middleware)

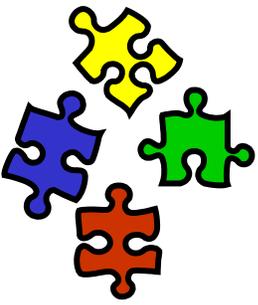




Visão de Execução

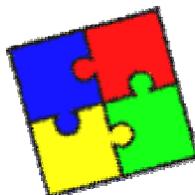
- Define:
 - Entidades de execução
 - Caminhos de Comunicação entre as entidades
 - Configuração de execução

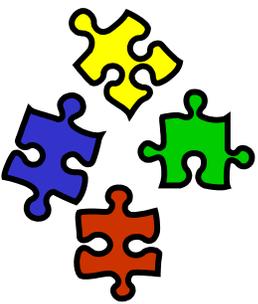




Visão de Execução

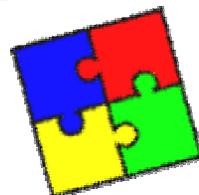
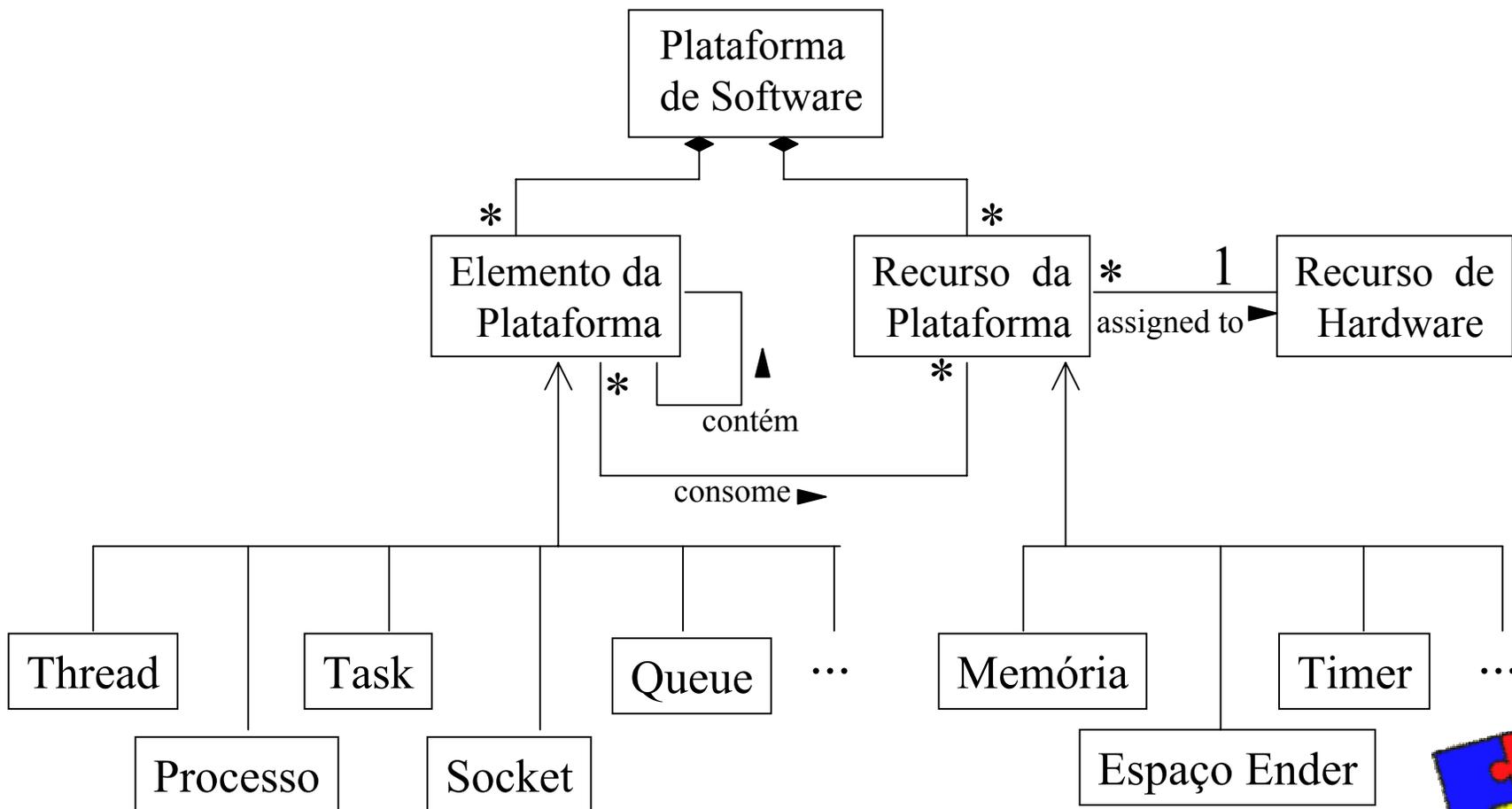
- Atividades Iniciais:
 - Liste os componentes de hardware do ambiente de execução
 - Liste a plataforma de software
 - Liste os elementos da plataforma que serão usados na visão de execução

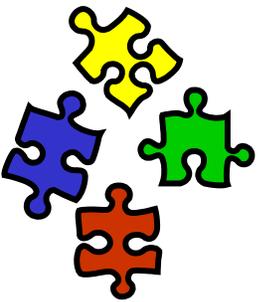




Visão de Execução

Meta-Modelo dos Elementos da Plataforma

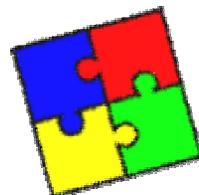


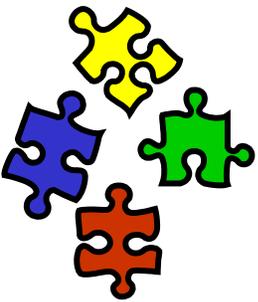


Visão de Execução

Entidades de Execução

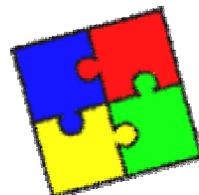
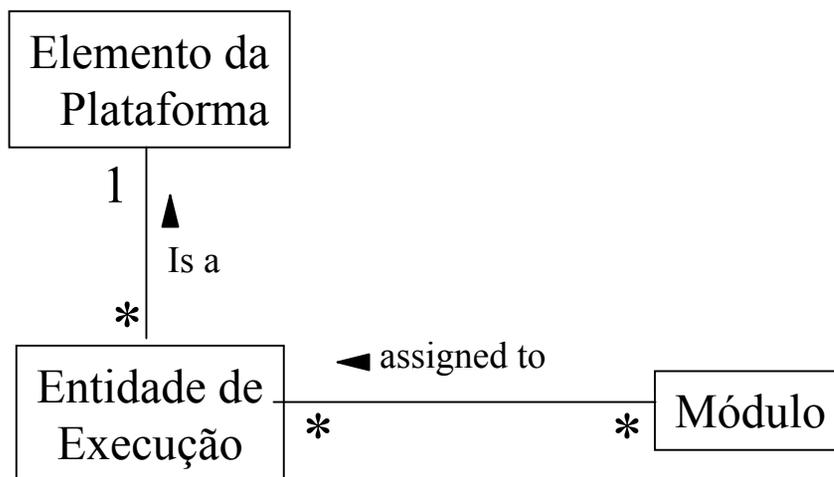
- Após identificar os elementos da plataforma de software deve-se decidir como mapear os componentes conceituais e módulos para os elementos da plataforma
 - Um ou mais módulos são representados por uma entidade de execução e um módulo pode ser atribuído a mais de uma entidade de execução
 - Podem existir entidades de execução como processos servidores que não tenham correspondência direta com módulos mas são necessários para dar suporte a outras entidades de execução.

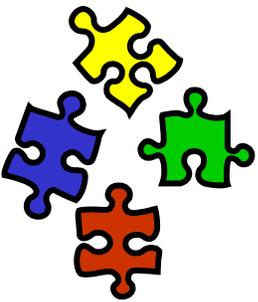




Visão de Execução

Meta-modelo para Entidades de Execução

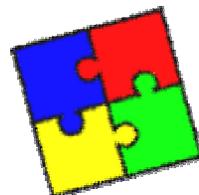


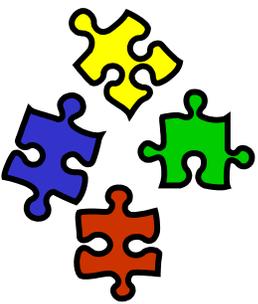


Visão de Execução

Caminho de Comunicação

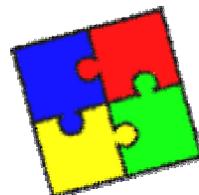
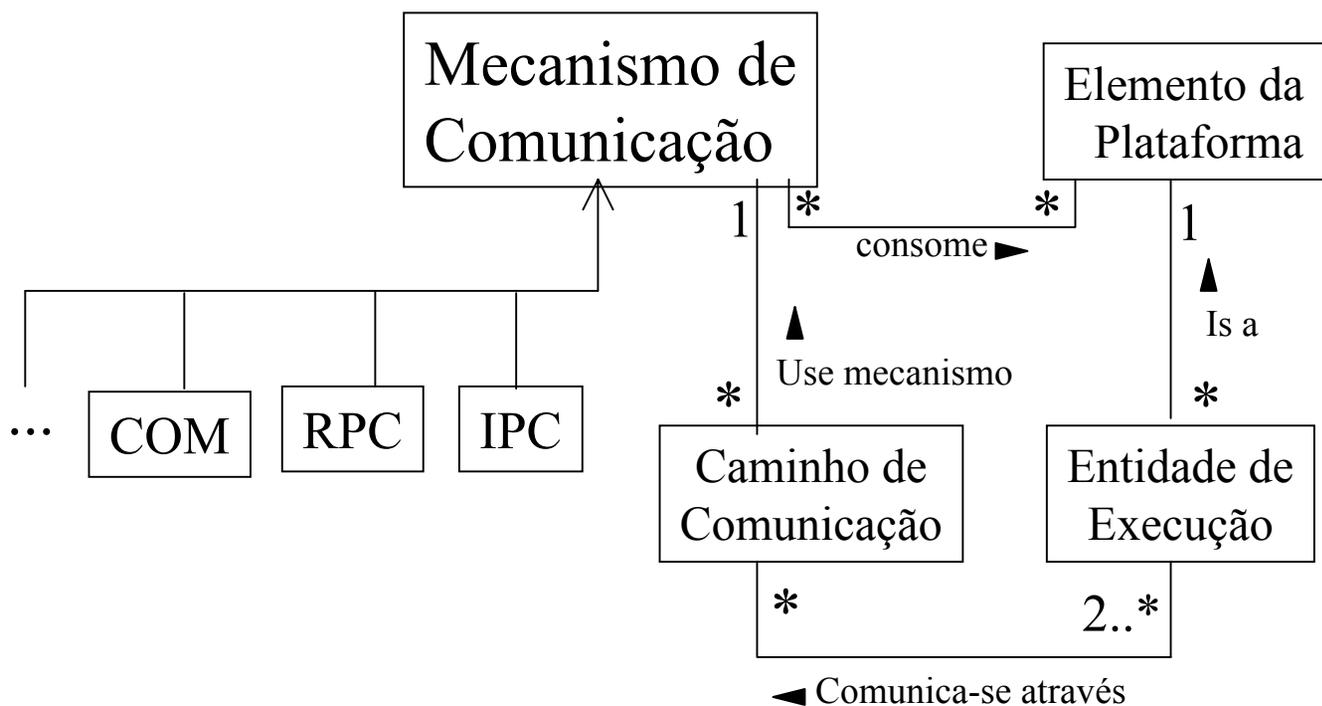
- Identificar os caminhos de comunicação entre as entidades de execução (mecanismos e recursos usados para comunicação)

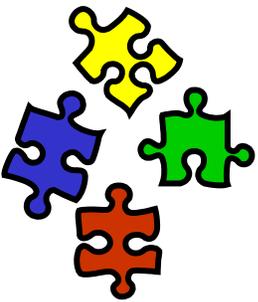




Visão de Execução

Meta-modelo para Comunicação

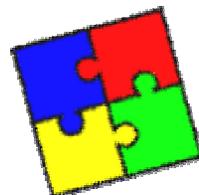


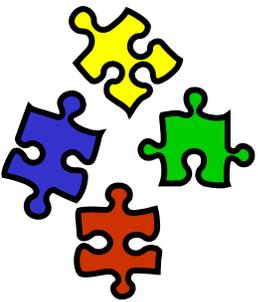


Visão de Execução

Configuração de Execução

- Descreve a topologia de execução do sistema caracterizando as instâncias das entidades de execução e como elas são interconectadas.
- Há distinção entre uma entidade de execução e suas correspondentes instâncias (exceto quando só há uma instância)

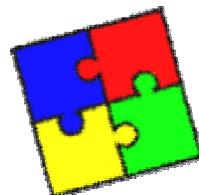


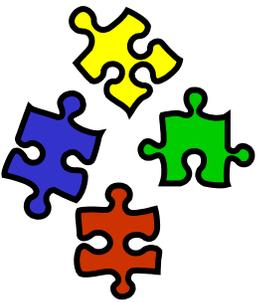


Visão de Execução

Configuração de Execução

- Deve-se determinar e descrever como a configuração muda ao longo do tempo e como estas mudanças são controladas.

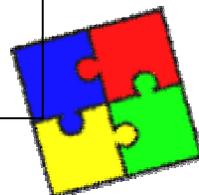


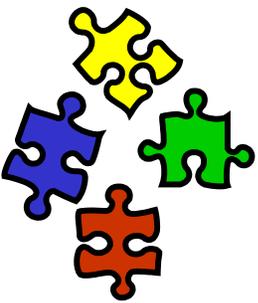


Visão de Execução

Resumo da Representação

Artefato	Representação
Configuração de Execução	Diagrama de Classes UML
Configuração de Execução mapeadas a dispositivos de hardware	Diagrama de Desenvolvimento UML
Comportamento dinâmico da configuração, ou transição entre configurações	Diagrama de Sequência UML
Descrição das entidades de execução (tipos de host, replicação, etc)	Diagrama de Classes UML ou Tabela
Protocolo de Comunicação	Diagrama de Sequência ou Statechart





Visão de Execução

Meta-modelo da Visão de Execução

