

nemo

ontology & conceptual
modeling research group



Banco de Dados – Análise de Requisitos

Vítor E. Silva Souza

[vitorsouza@inf.ufes.br]

<http://www.inf.ufes.br/~vitorsouza>

Departamento de Informática

Centro Tecnológico

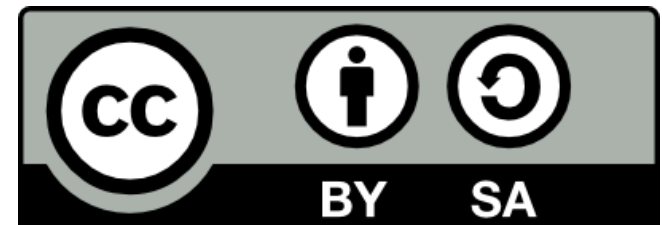
Universidade Federal do Espírito Santo

Licença para uso e distribuição

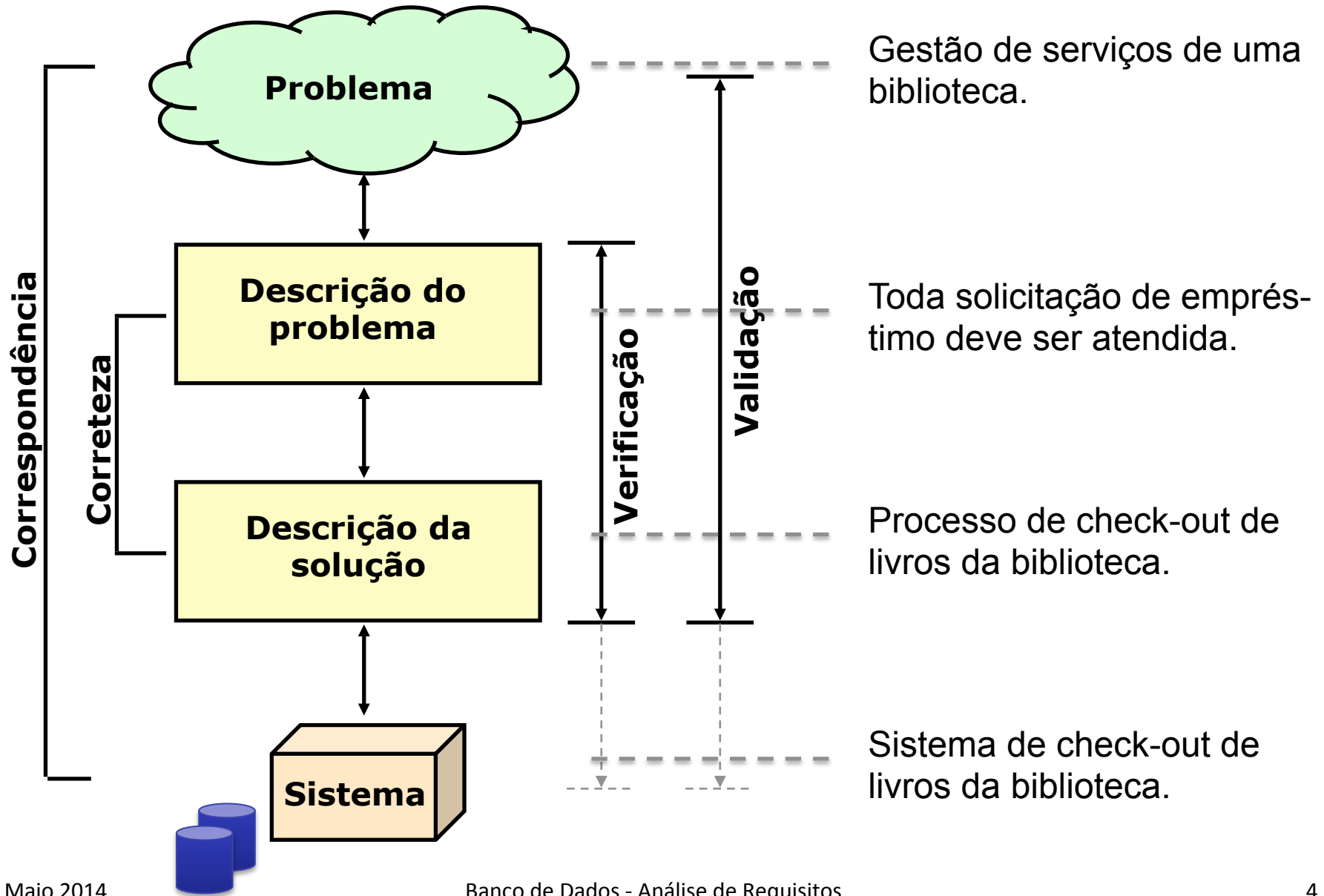
- Este obra está licenciada com uma licença Creative Commons Atribuição-CompartilhaIgual 4.0 Internacional;
- You are free to (for any purpose, even commercially):
 - Share: copy and redistribute the material in any medium or format;
 - Adapt: remix, transform, and build upon the material;
- Under the following terms:
 - Attribution: you must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use;
 - ShareAlike: if you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.



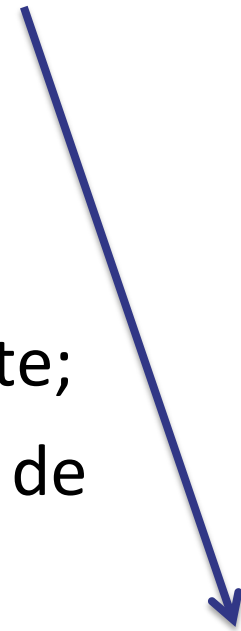
Mais informações podem ser encontradas em:
<http://creativecommons.org/licenses/by-sa/4.0/>



- Algumas informações foram retiradas e adaptadas de apresentações de um curso sobre Engenharia de Requisitos do prof. Steve Easterbrook, modificadas pelo prof. John Mylopoulos.



- O problema mais óbvio pode não ser o problema que deve ser resolvido;
- Deve-se discutir o problema com os *stakeholders* para um melhor entendimento;
- Uma descrição precisa do problema:
 - Auxilia nas escolhas de projeto;
 - Permite a criação de bons casos de teste;
 - Auxilia na comunicação entre a equipe de desenvolvimento.



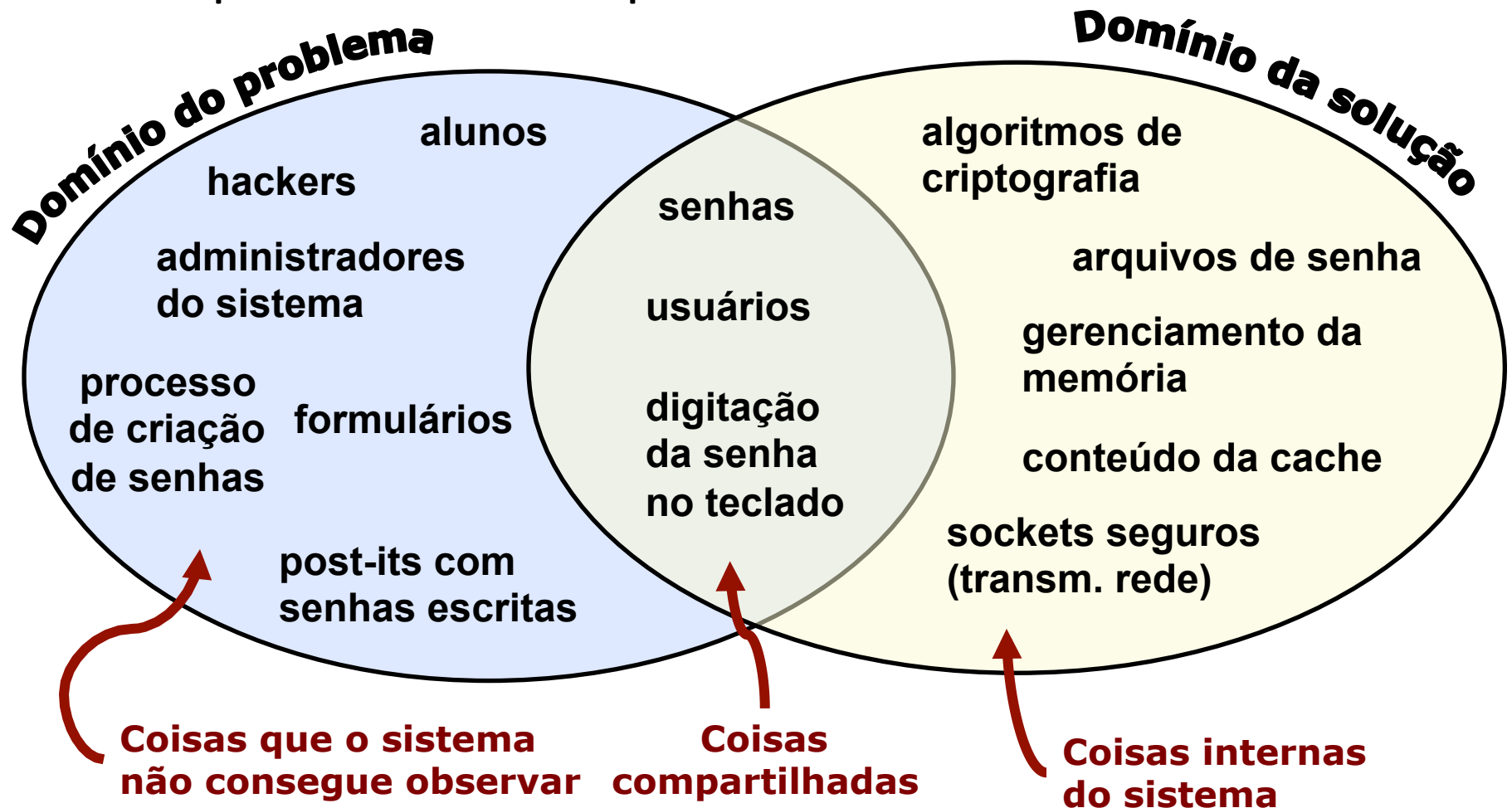
Stakeholders são pessoas que são afetadas pelo problema e, portanto, tem algo a dizer sobre sua solução. Ex.: clientes, usuários, etc.



- **Propriedades** do domínio:
 - Coisas que são verdadeiras independente de construirmos ou não um sistema;
- **Requisitos**:
 - Coisas que queremos tornar verdadeiras após a construção do sistema;
- **Especificação**:
 - Uma descrição do comportamento que o sistema deve ter para atender os requisitos.

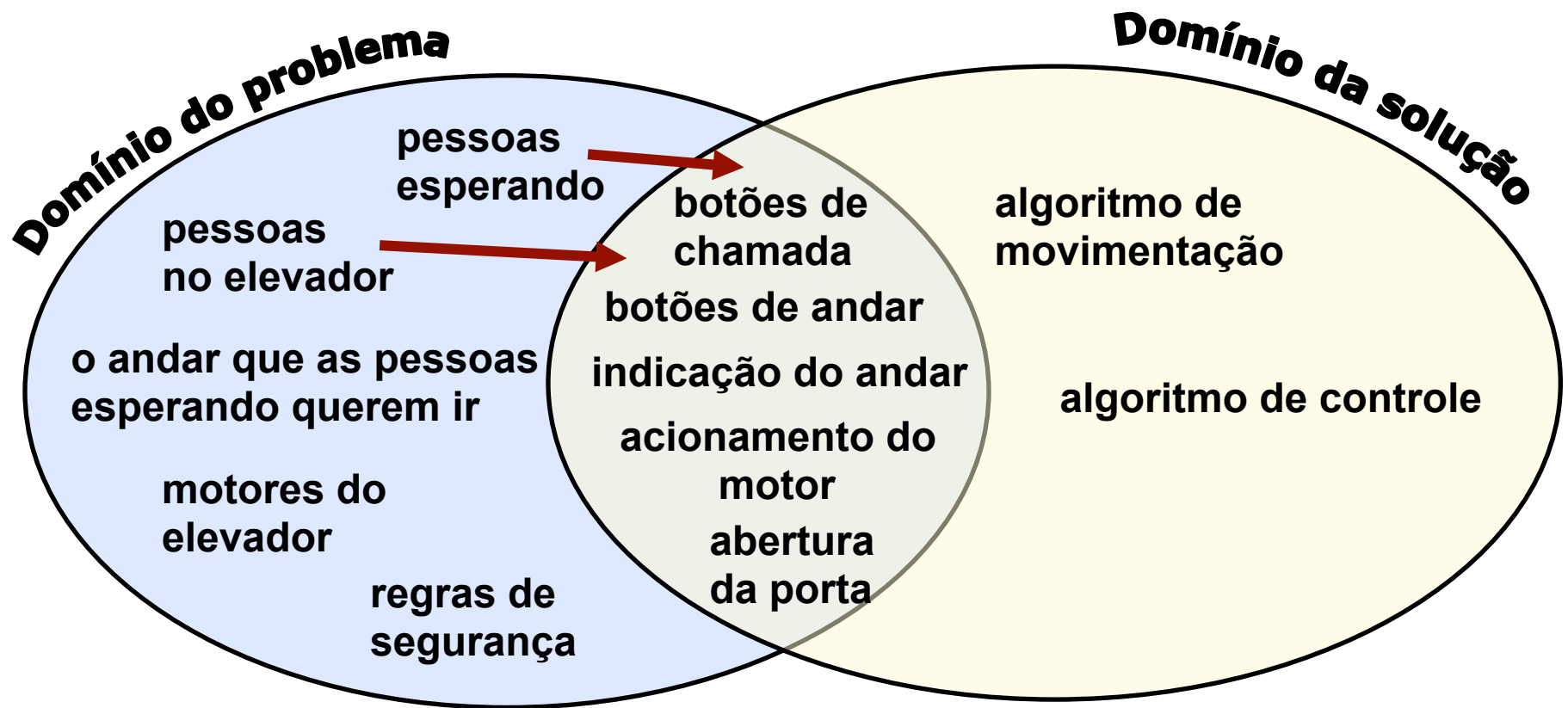
Problema vs. solução

- Exemplo: evitar acesso não autorizado aos computadores do departamento:



- Requisito (R):
 - Os computadores deverão ser acessados somente por pessoal autorizado;
- Propriedades do domínio (D):
 - Pessoal autorizado pode memorizar uma senha;
 - Senhas não são compartilhadas entre pessoas;
- Especificação (S):
 - Acesso aos computadores será liberado mediante digitação de nome de usuário e senha;
- $S, D \rightarrow R$
 - E se as suposições sobre o domínio forem falsas?

- As fronteiras são maleáveis:
 - Ex.: inclusão de sensores.



- Critérios de correção:
 - Um determinado **sistema** satisfaz a **especificação**;
 - A **especificação**, dadas as **propriedades** de domínio, satisfaz os **requisitos**.
- Critérios de completude:
 - Descobrimos todos os **requisitos** importantes;
 - Descobrimos todas as **propriedades** de domínio relevantes.

- Entender o que os *stakeholders* querem;
- Analisar a necessidade;
- Verificar a factibilidade;
- Negociar uma solução razoável;
- Especificar a solução sem ambiguidade;
- Validar a especificação;
- Gerenciar mudanças nos requisitos;
- Etc.

Neste curso veremos apenas uma parte da Engenharia de Requisitos.

Banco de Dados – Análise de Requisitos

ELICITAÇÃO DE REQUISITOS

- São funções desta fase do processo:
 - Concepção;
 - Elicitação;
 - Elaboração;
 - Negociação;
 - Especificação;
 - Validação;
 - Gerenciamento.

Independente se
feitas em sequência
ou em paralelo...

- Geralmente é a primeira fase do processo;
- Identificação de um problema ou oportunidade;
- Perguntas genéricas e superficiais;
- Objetivo é estabelecer:
 - Um entendimento básico do problema;
 - Quem são os *stakeholders*;
 - A natureza da solução desejada;
 - A eficácia da comunicação entre o engenheiro de requisitos e os especialistas de domínio.

- Quem está pedindo esta solução?
- Quem irá usá-la?
- Qual é seu benefício econômico?
- De que problema(s) esta solução irá tratar?
- Em que ambiente de negócio ela está inserida?
- Existem qualidades fundamentais (desempenho, segurança, etc.) relevantes ao problema?
- ...

- Meta-questões auxiliam a estabelecer a eficiência da comunicação analista-especialistas:
 - Você é a pessoa certa para responder estas questões? Suas respostas são “oficiais”?
 - Minhas questões são relevantes?
 - Estou perguntando coisas demais?
 - Existe outra pessoa que pode prover mais informações?
 - Existe alguma pergunta que eu deveria ter feito e não fiz?
 - Com quem mais eu deveria conversar sobre isso?

- A meta é descobrir informações sobre o problema:
 - Os objetivos dos *stakeholders* (problema);
 - As funções do sistema (solução) a ser construído (o que ele deve fazer);
 - Como o sistema se encaixa nas necessidades de negócio do cliente;
 - Como será usado no dia-a-dia.
- Atividade muito complicada;
- Requer alto nível de organização.

- As “primeiras perguntas” darão somente um entendimento básico do problema;
- Para elicitar os requisitos, devemos utilizar abordagens mais sofisticadas. Veremos algumas destas:
 - Amostragem;
 - Investigação;
 - Entrevista;
 - Observação;
 - Questionário;
 - Prototipação.

- De escopo:
 - Os limites do sistema não são bem definidos;
 - O cliente especifica muitos detalhes inúteis.
- De entendimento:
 - O cliente não tem certeza do que quer;
 - Não conhece as capacidades e limitações do ambiente computacional;
 - Possui problemas de comunicação com os engenheiros de software;
 - Omite informações consideradas “óbvias”;

- De entendimento (continua):
 - Especifica requisitos que conflitam com os de outro cliente;
 - Especifica requisitos ambíguos.
- Políticos:
 - Funcionários não colaboram por acharem que o sistema lhes custará o emprego;
 - Brigas políticas internas.
- De volatilidade:
 - Os requisitos mudam com o tempo.



Como o cliente explicou...



Como o lider de projeto entendeu...



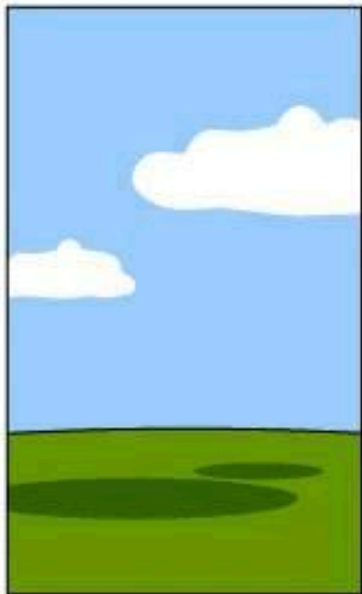
Como o analista projetou...



Como o programador construiu...



Como o Consultor de Negócios descreveu...



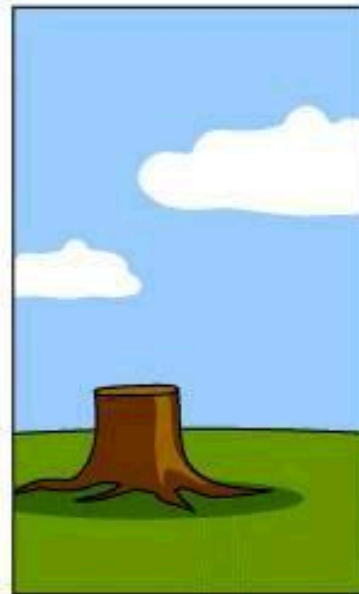
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...

- Tem por objetivo a redução da ambiguidade (se possível, sua eliminação);
- São utilizados modelos (o nível de formalidade pode variar, dependendo do projeto);
- Formalizar os requisitos em modelos auxilia na identificação de falhas na elicitação;
- Modelos são úteis também para negociação e validação.

- Não é incomum:
 - Clientes quererem mais do que é possível ser feito;
 - Stakeholders terem requisitos conflitantes.
- Deve-se reconhecer os múltiplos pontos de vista e tentar negociar uma solução adequada;
- Idealmente, deve-se evitar situações em que hajam “vencedores” e “perdedores”.

“Coloque três *stakeholders* numa sala e pergunte que tipo de sistema eles querem. Você provavelmente vai obter quatro ou mais opiniões diferentes” (Anônimo)

- Produto final da Engenharia de Requisitos;
- Uma especificação pode ser:
 - Um documento escrito;
 - Modelos gráficos;
 - Um modelo formal matemático;
 - Um protótipo;
 - Qualquer combinação dos itens acima...
- É a base das fase seguintes da Engenharia de Software;
- Varia de acordo com características do projeto (tamanho, “criticalidade”, etc.).

- Revisões técnicas para averiguar que:
 - Todos os requisitos foram elencados sem ambiguidade;
 - Inconsistências, omissões e erros foram detectadas e corrigidas;
 - Tudo está documentado de acordo com padrões estabelecidos pela organização.

- Atividades que ajudam no controle e rastreamento de mudanças nos requisitos:
 - Cada requisitos recebe um identificador;
 - São montadas tabelas de rastreamento: funcionalidades, dependências, subsistemas, interface, etc.;
 - Mudanças nos requisitos podem ser mais facilmente propagadas “para frente”;
 - Bugs no software pronto podem ser analisados em termos dos requisitos (“para trás”).

Banco de Dados – Análise de Requisitos

TÉCNICAS DE ELICITAÇÃO

- Há muitos *stakeholders*. Quais devem participar?
- Há muitos documentos. Quais devem ser analisados?
- Amostragem:
 - Seleção sistemática de elementos representativos;
 - Útil em combinação com outras técnicas;
 - Reduz custos, acelera o processo de elicitação;
 - Reduz tendências (de uma seleção sem critério).
- Tamanho da amostra: depende substancialmente do custo envolvido e do tempo requerido/disponível.

- Passos:
 1. Determinar o que será coletado e para que;
 2. Determinar a população a ser amostrada;
 3. Escolher o tipo de amostra;
 4. Decidir sobre o tamanho da amostra.

Elementos são selecionados...	Não baseada em probabilidades	Baseada em probabilidades
diretamente, sem restrições:	De Conveniência	Randômica Simples
segundo um critério específico:	Intencional	Randômica Complexa

- Irrestrita (não há critério de seleção);
- Não utiliza probabilidades;
- Mais fácil;
- Geralmente apresenta resultados irreais;
- Ex.: aviso chamando os interessados a participar de uma reunião.

- São estabelecidos critérios de seleção;
- Não leva em conta probabilidades;
- Amostra moderadamente confiável;
- Ex.: escolha de um grupo de indivíduos que aparenta ter conhecimento na área do novo sistema.

- Baseada em probabilidades;
- Não há critérios de seleção;
- Parte de uma lista de documentos/pessoas a ser amostrada:
 - Todos possuem igual probabilidade de serem escolhidos;
 - Escolhe-se N elementos;
 - Dependendo da quantidade de documentos e pessoas, se torna impraticável.

- Deve ser a escolha preferencial;
- Utiliza probabilidades em cima de critérios de seleção (não sobre toda a população).

- Algumas informações são difíceis de obter por entrevistas ou observação:
 - Histórico da organização;
 - Direcionamento futuro;
 - Informações financeiras;
 - Contextos da organização;
 - Etc.
- Utiliza-se de investigação (análise de documentos).

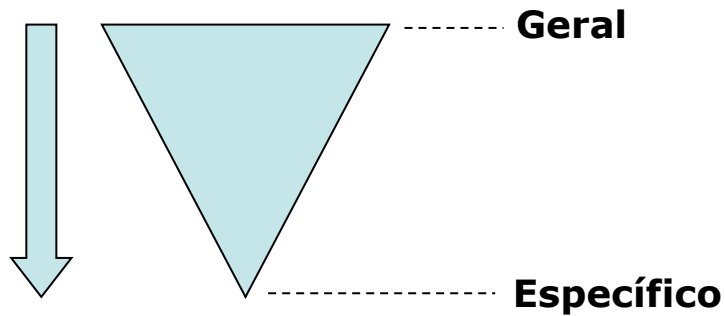
- Documentos quantitativos:
 - Relatórios de resultados, fichas e formulários, etc.
 - Observar como são feitos os cálculos, procurar oportunidades de otimização, fluxo de comunicação, redundância/campos inúteis, fluxos não oficiais, etc.;
- Documentos qualitativos:
 - Memorandos, quadros de aviso, manuais, etc.;
 - Identificar termos que caracterizam o que é “bom” ou “ruim” para a organização, disputas (internas e externas), políticas e características da org., etc.

- Conversa direcionada com um propósito específico;
- Formato pergunta – resposta, geralmente 2 pessoas (mas pode haver casos com mais de um entrevistador ou entrevistado);
- Objetiva obter:
 - As opiniões do entrevistado (descoberta de problemas-chave a serem tratados);
 - Seus sentimentos sobre o estado atual do sistema;
 - Metas organizacionais e pessoais;
 - Procedimentos informais.

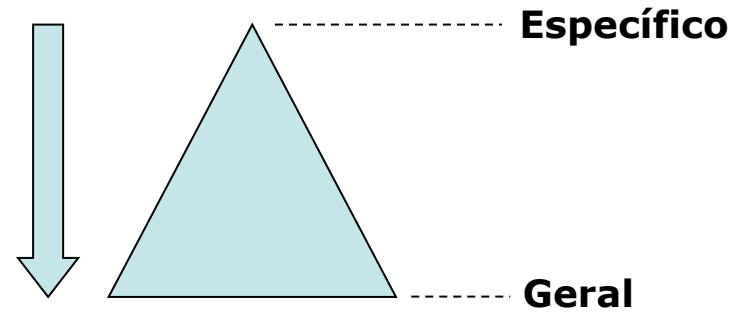
- É importante para o entrevistador:
 - Construir uma base de confiança e entendimento;
 - Manter o controle da entrevista;
 - Vender “a ideia do sistema”;
 - Evitar questões tendenciosas (que induzem).
- Etapas principais:
 - Planejamento: ler material existente, estabelecer objetivos, decidir quem entrevistar, marcar;
 - Condução: questões objetivas x subjetivas, organização estruturada x não estruturada;
 - Relatório da entrevista: documentação.

- Vantagens:
 - Proveem riqueza de detalhes;
 - Revelam novos questionamentos;
 - Colocam o entrevistado à vontade;
 - Permitem maior espontaneidade.
- Desvantagens:
 - Podem resultar em muitos detalhes irrelevantes;
 - Perda do controle da entrevista;
 - Respostas muito longas para se obter pouca informação útil;
 - Podem dar a impressão de que o entrevistador está perdido, sem objetivos.

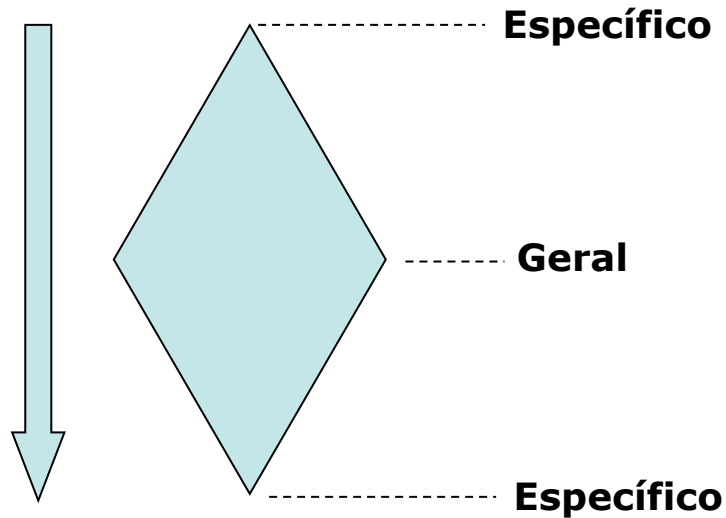
- Vantagens:
 - Ganho de tempo, vão direto ao ponto em questão;
 - Mantêm o controle da entrevista;
 - Levam a dados relevantes.
- Desvantagens:
 - Podem ser maçantes para o entrevistado;
 - Podem falhar na obtenção de detalhes importantes;
 - Não constroem uma afinidade entre entrevistador e entrevistado.



Funil



Pirâmide



Diamante

- Gravador/filmadora:
 - Requer permissão;
 - Registro completo;
 - Rapidez e melhor desenvolvimento;
 - Reprodução para outros membros da equipe;
 - Pode deixar o entrevistado nervoso ou distraído;
 - Pode haver necessidade de transcrever a fita.
- Anotações:
 - Mantém o entrevistador alerta;
 - Mostra interesse e preparação do entrevistador.
 - Pode haver perda do andamento da conversa;
 - Muita atenção a fatos e não a sentimentos e opiniões.

- Deve capturar a essência da entrevista;
- Escreva o relatório o mais breve possível para assegurar a qualidade;
- Registre: entrevistado, entrevistador, data, assunto e objetivos;
- Diga se os objetivos foram alcançados;
- Aponte objetivos para entrevistas futuras;
- Registre os pontos principais da entrevista e sua opinião.

- Questões escritas e distribuídas para um conjunto de pessoas envolvidas com o sistema;
- Objetivos:
 - Procurar quantificar o que foi achado em entrevistas;
 - Determinar quanto um sentimento é realmente difundido ou limitado;
 - Examinar uma grande amostra de usuários para sentir problemas ou levantar questões importantes antes das entrevistas (estudo exploratório).
- Muito similar às entrevistas, pode ser utilizado em conjunto para esclarecimentos ou aprofundamentos.

- Não é possível refinar/explicar questões ou mudar a estrutura como em entrevistas. Portanto se deve:
 - Ter questões claras e não ambíguas;
 - Ter fluxo definido (capture o interesse primeiro, deixe itens controversos por último);
 - Levantar, antecipadamente, as dúvidas das pessoas que irão respondê-lo. Testar o questionário.
- Use a linguagem dos usuários, garanta precisão técnica;
- Prime pela simplicidade, perguntas rápidas;
- Evitar que os usuários se sintam investigados ou obrigados. Pergunte só aquilo que saibam responder.

	Subjetivas	Objetivas
Tempo gasto para responder	Alto	Baixo
Natureza exploratória	Alta	Baixa
Amplitude e profundidade	Alta	Baixa
Facilidade de preparação	Fácil	Difícil
Facilidade de análise	Difícil	Fácil

- Questões subjetivas: cuidado com respostas muito amplas, que dificultem o processamento;
- Questões objetivas: cuidado com escalas para evitar perguntas tendenciosas ou o efeito “coluna do meio”.

- Observar o comportamento dos “tomadores de decisão” em seu ambiente de trabalho;
 - “Tomadas de decisão”: ações que influenciam o dia-a-dia da tarefa a ser automatizada;
 - Diversos níveis: operacional, gerencial, estratégico;
- Objetivos:
 - Levantar o que é realmente feito na prática;
 - Identificar relacionamentos e influências de líderes;
 - Obter informações não capturadas de outra forma e/ou confirmar/negar informações de investigação, entrevistas e questionários.

- Preparação:
 - Quem (ou o que)? (Quais atividades)
 - Quando? (Por horário ou por evento)
 - Onde?
 - Por que?
 - Como? (Nível de detalhes)
- Registro (documentação);
 - Preparar os formulários de registro de antemão para serem preenchidos durante a observação.

- Construção de uma versão incompleta do sistema para que futuros usuários o experimentem;
- Permite capturar:
 - Reações iniciais do usuário;
 - Sugestões de melhorias, novas capacidades;
 - Prioridades em relação às diferentes funções.
- Um protótipo pode ser:
 - Não operacional (somente interface);
 - “Arranjado às pressas” (completo, porém ruim);
 - “Primeiro de uma série” (completo, piloto);
 - Características selecionadas (uma fatia do sistema).

Vantagens

- Permite alterar o sistema mais cedo (quando o custo é menor);
- Oportunidade de avaliação de viabilidade do sistema;
- Leva a sistemas que atendem melhor as necessidades dos usuários.

Desvantagens

- Maior dificuldade de gerenciamento do projeto;
- Adoção do protótipo como sistema completo.

Protótipos devem ser construídos rapidamente, enfatizando a interface com o usuário e com envolvimento dos mesmos para feedback imediato e sua subsequente modificação.

- Elicitação é o meio;
- A informação a ser levantada é o fim;
- O que queremos descobrir?
 - Quem são os stakeholders?
 - Quais são seus objetivos?
 - Quais funcionalidades querem pro sistema?
 - Quais são os elementos de informação do sistema?

Cenas dos próximos capítulos...

- No contexto do problema escolhido para o seu trabalho prático, tente responder as seguintes questões:
 - Quais técnicas seriam úteis para entender o contexto atual e o problema a ser resolvido? Por que?
 - Quais técnicas não seriam úteis? Por que?
 - Para as técnicas escolhidas, esboce:
 - Investigação: que documentos investigar?
 - Entrevista/questionários: quem vai responder? Qual o teor das perguntas?
 - Observação: o que observar?
 - Prototipação: que parte do sistema seria prototipada? Por que?

Banco de Dados – Análise de Requisitos

ANÁLISE DE OBJETIVOS

- Requisitos prescrevem o que um sistema deve fazer;
- Objetivos explicam o porquê;
- Aumentam o nível de abstração e trazem vantagens:
 - Critério de completude/pertinência dos requisitos;
 - Maior legibilidade em documentos complexos;
 - Exploração de alternativas;
 - Detecção de conflitos entre *stakeholders*;
 - Maior estabilidade.

- Imagine uma biblioteca de uma faculdade:
 - “Livros não estão disponíveis no início do período”;
- Isso é um sintoma. Bibliotecários, por que?
 - “Porque os professores não nos mandam a bibliografia em tempo hábil”;
- Também um sintoma? Professores, por que?
 - “Porque as matérias são alocadas no último minuto”;
- Sintoma? Comissão de ensino, por que?
 - “Porque não sabemos quais professores estão disponíveis no período até o último minuto”;
- Sintoma? Chefe do departamento, por que?
 - “Porque existe incerteza com relação a afastamentos, novas contratações (substitutos), monitores, etc.”
- ...

- Todo problema pode ser visto como sintoma de outro;
- Cuidado para não continuar indefinidamente;
- Abordagem – verificar com os *stakeholders* se:
 - É possível resolver este problema, independente de sua causa (do problema maior)?
 - Resolver este problema (e não a causa) ajuda? Querem que seja resolvido?
 - Alguém vai pagar para resolver este problema (independente da causa)?

A análise de factibilidade ajuda com essas questões.

- Escolheu-se como problema o atraso no envio da bibliografia por parte dos professores:
 - Solução: informatizar os formulários de envio do programa da disciplina;
- E já que estamos fazendo isso:
 - Que tal informatizar também o envio de ordens de compra de livros que não temos às editoras?
 - Aproveita-se também para informatizar a gerência da coleção, livros que estragam, somem, etc.
 - Daí já automatizamos também o empréstimo/ devolução de livros pelos alunos...
- Opa! Este novo sistema está me custando muito caro!

- Podemos informatizar sempre cada vez mais;
- Abordagem – verificar com os *stakeholders* se:
 - A solução escolhida é implementável (independente das alternativas)?
 - Implementar esta alternativa (independente de outras) ajuda a resolver o problema? A solução satisfaz?
 - Alguém vai pagar para implementar esta solução?

A análise de alternativas ajuda com essas questões.

- Sistema atual: contas dos clientes são atualizadas duas vezes por dia, incluindo diária, serviço de quarto, frigobar, pay-per-view, telefone, etc.
- Quando o cliente deixa o hotel, deve mencionar cobranças recentes (daquele dia), que são então adicionadas à conta para serem pagas;
- Gerência do hotel gostaria de mudar o sistema porque:
 - Erros nas contas são frequentes: clientes não pagam alguns consumos ou são cobrados em dobro por outros;
 - Há uma expectativa de crescimento dos negócios: uma extensão do hotel está sendo construída;
 - Atualização manual dos registros dos hóspedes pode se tornar problemática após a expansão;
 - Querem ter atualizações em tempo real dos registros dos hóspedes a partir de qualquer ponto de consumo.

- Quais são os problemas?
- Quais são as alternativas?
- Quais são os critérios de seleção?
- Qual a sua recomendação?

- Quais são os problemas?
 - Perda de renda por registros defasados/errados;
 - Registrar informações tem um alto custo;
 - Potenciais problemas com expansão dos negócios;
- Quais são as alternativas?
 - Manter o sistema atual;
 - Manter o sistema, com atualizações mais frequentes;
 - Construir um sistema integrado, em tempo real;
- Quais são os critérios de seleção?
 - Custo-benefício (desenvolvimento e operação, sistema novo vs. sistema atual, redução de perdas);
 - Conveniência/satisfação dos hóspedes;
- Qual a sua recomendação?
 - ??

- Primeiras perguntas devem identificar os *stakeholders*:
 - Quem está apresentando o problema / solicitando sua solução?
 - Quem está pagando pelo desenvolvimento?
 - Quem se beneficia economicamente com ele?
 - Quem são as pessoas que usarão o sistema ou participarão do mesmo?
 - Em uma empresa, verificar se os gerentes/diretores estão cientes e apoiam a iniciativa. Departamento de marketing geralmente também se envolve;
- Ao longo do projeto, pode-se identificar novos.

- Identificados os *stakeholders*, passamos para a elicitação dos seus objetivos.



Quando perguntei dos seus objetivos para o ano que vem eu tinha outra ideia em mente.

Dilbert.com DilbertCartoonist@gmail.com



Não "trabalhar o mínimo possível enquanto evito a ira do troll de cabelos pontiagudos."

© 2011 Scott Adams, Inc./Dist. by Universal Uclick



Não chame-os de **meus** objetivos se você quer dizer os **seus** objetivos.

- Empresa de grande porte precisa fazer várias reuniões (internas e externas) diariamente;
- Reuniões variam em data, hora, local, participantes, recursos exigidos;
- Empresa possui várias salas de reunião, variando também em localização, tamanho, recursos;
- Gostariam de um sistema que:
 - Utilizasse as salas de maneira eficiente;
 - Promovesse reuniões com boa participação;
 - Priorizasse funcionários que usam os recursos de maneira responsável.

- Quem são os *stakeholders*? Existem diferentes papéis?

- Quem são os *stakeholders*? Existem diferentes papéis?
 - Alta gerência;
 - Organizadores de reunião;
 - Participantes de reunião;
 - Secretárias.

Exercício: agenda de reuniões

- O que a alta gerência quer?

- O que a alta gerência quer?
 - Evitar desperdício de recursos;
 - Alta participação em reuniões;
 - Prioridades no agendamento;
 - Reduzir a carga de trabalho das secretárias para que elas possam focar em outras coisas;
 - Uso responsável do sistema de agendamento.

Exercício: agenda de reuniões

- O que os organizadores de reunião querem?

- O que os organizadores de reunião querem?
 - Organizar reuniões mais rapidamente (sem depender das secretárias);
 - Verificar facilmente todas as salas disponíveis para uma escolha adequada de sala de reunião;
 - Re-agendamento facilitado em caso de mudanças;
 - Saber os compromissos dos potenciais participantes da reunião para melhor escolha de data/hora.

Exercício: agenda de reuniões

- O que os participantes da reunião querem?

- O que os participantes da reunião querem?
 - Que as reuniões possam se adaptar a mudanças em seus calendários pessoais;
 - Diminuir a quantidade de conflitos de agenda;
 - Privacidade.

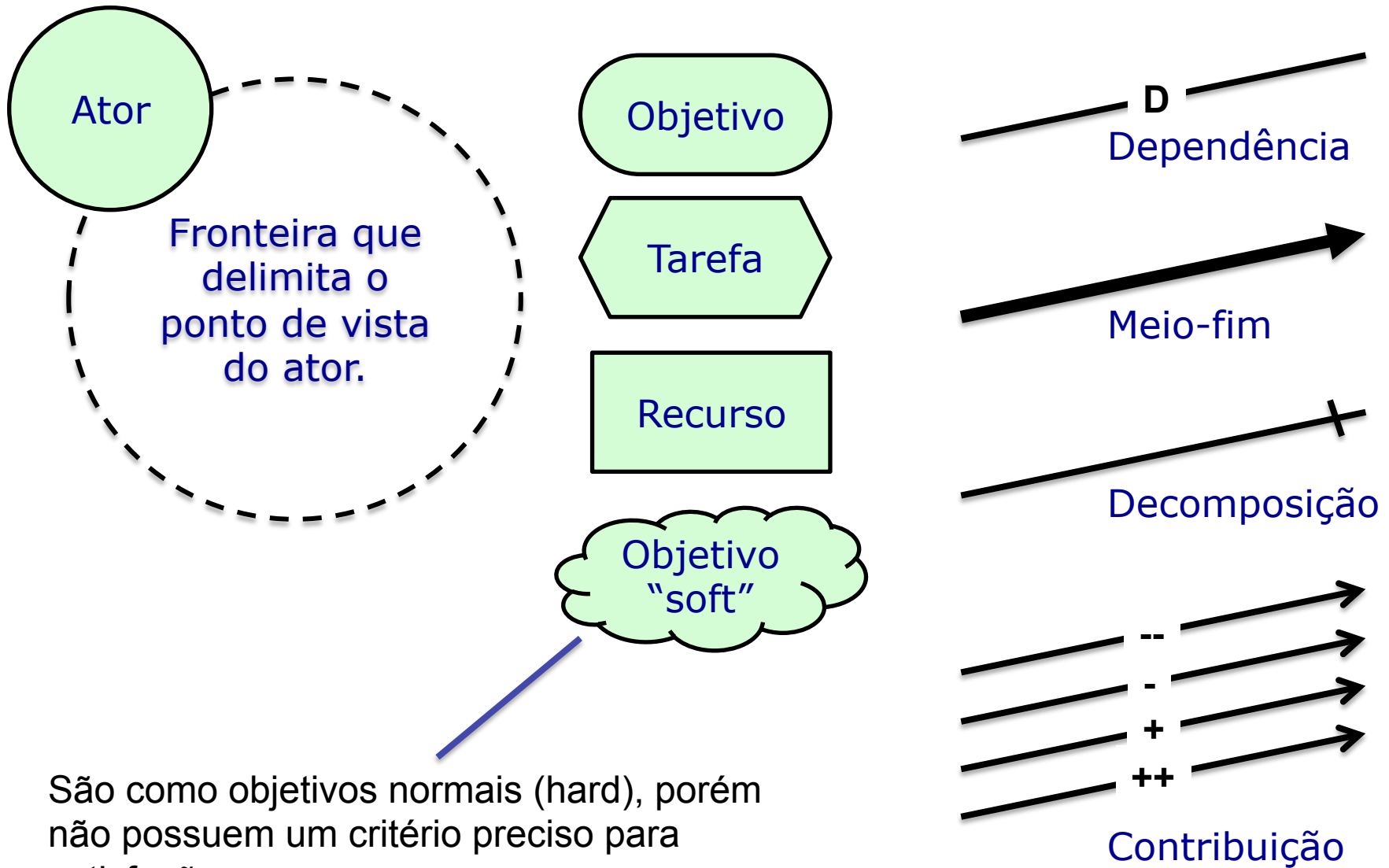
Exercício: agenda de reuniões

- O que as secretárias querem?

- O que as secretárias querem?
 - Reduzir a carga de trabalho de agendamento.

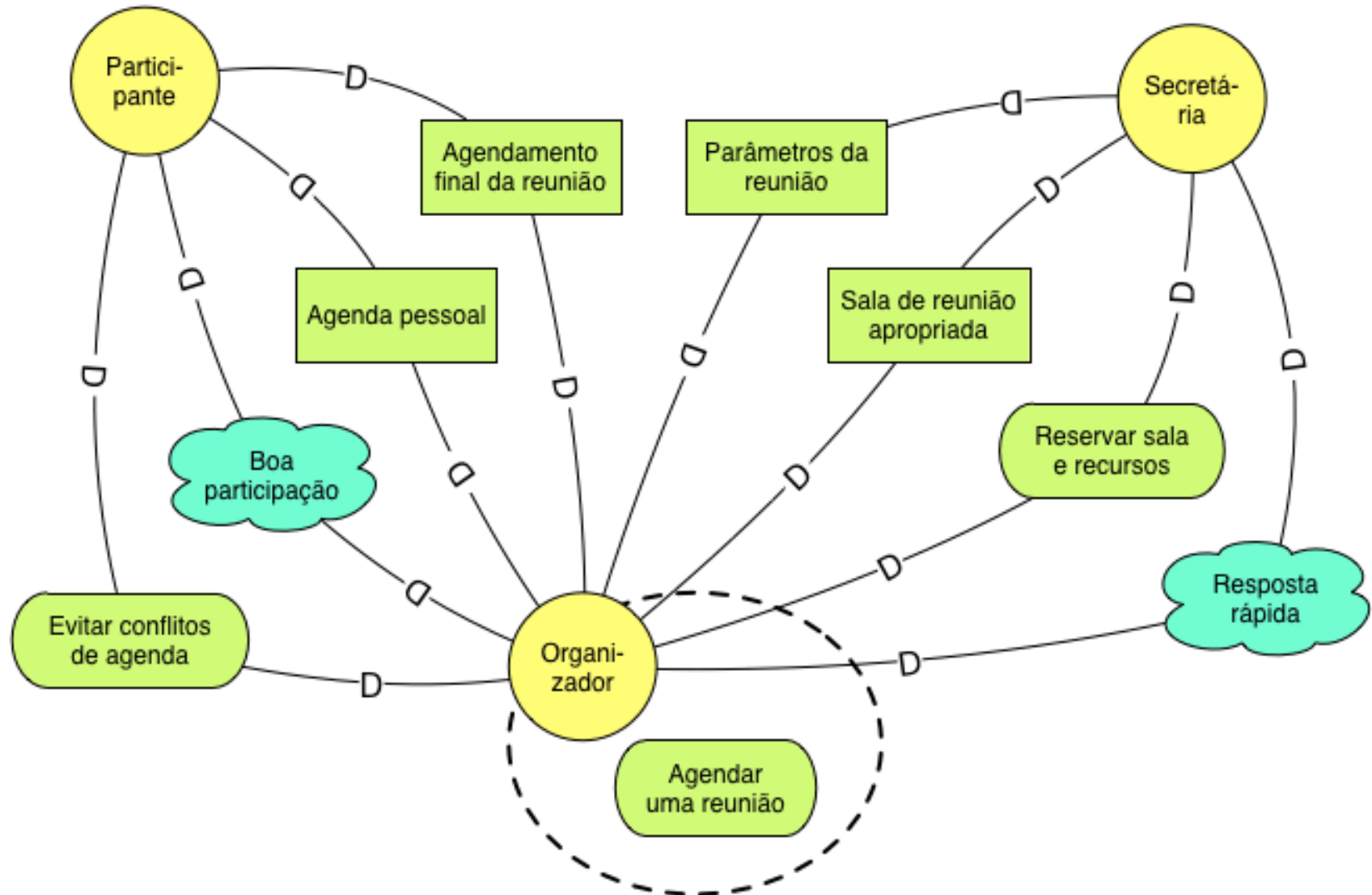
- Representam os objetivos dos *stakeholders*;
- Usado em diferentes propósitos:
 - Nos permitir comparar diferentes soluções para alcançar os mesmos objetivos;
 - Mostrar a interação entre os diferentes atores do sistema como ele é hoje (AS-IS);
 - Mostrar a interação entre os diferentes atores do novo sistema após sua implantação (TO-BE).

- Abordagem de modelagem criada pelo prof. Eric Yu na Universidade de Toronto;
- Propõe o uso de conceitos de interação social em modelos de análise de requisito e projeto de sistemas;
- Mais informações: <http://istar.rwth-aachen.de/>



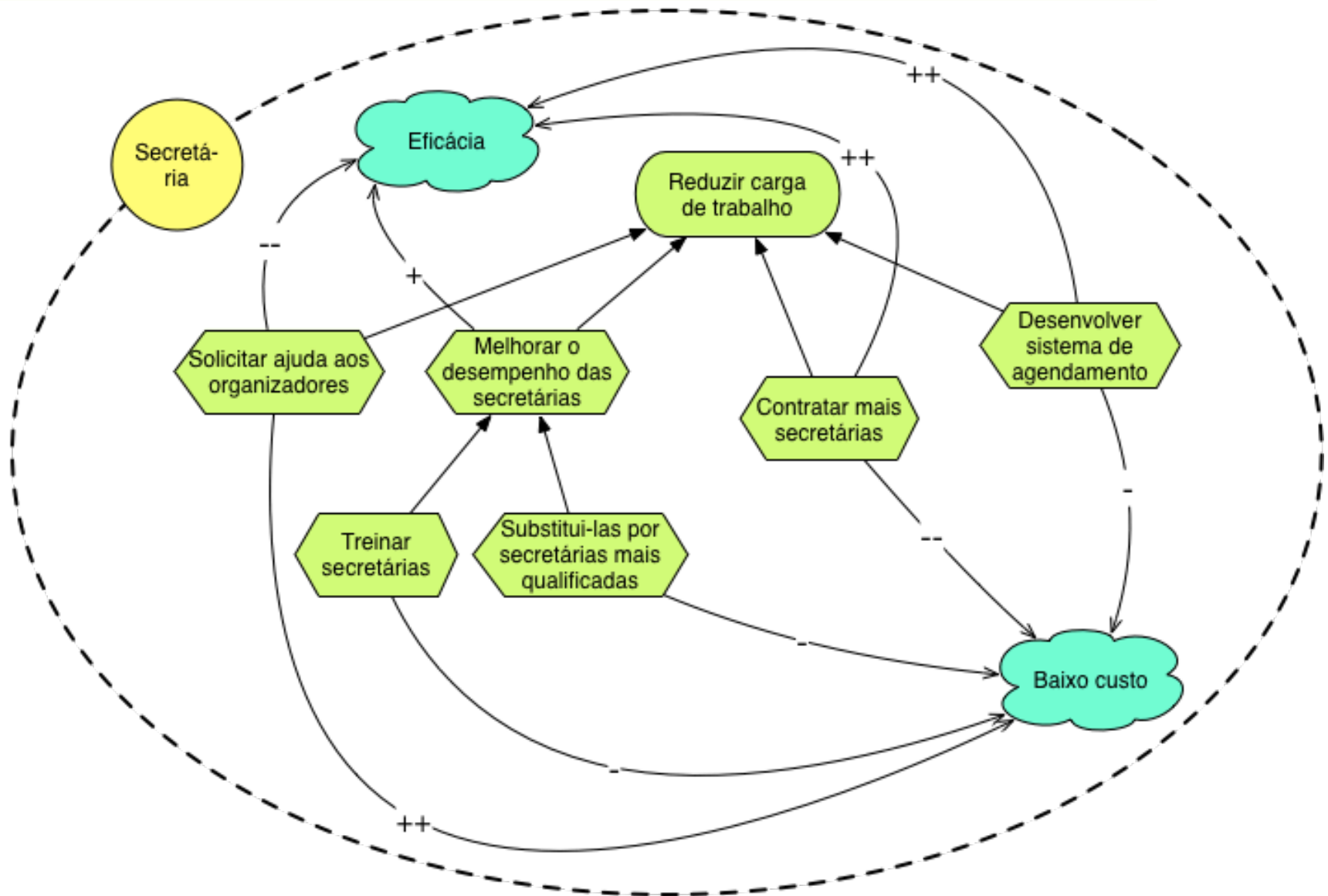
São como objetivos normais (hard), porém não possuem um critério preciso para satisfação.

Exemplo: interação dos atores AS-IS

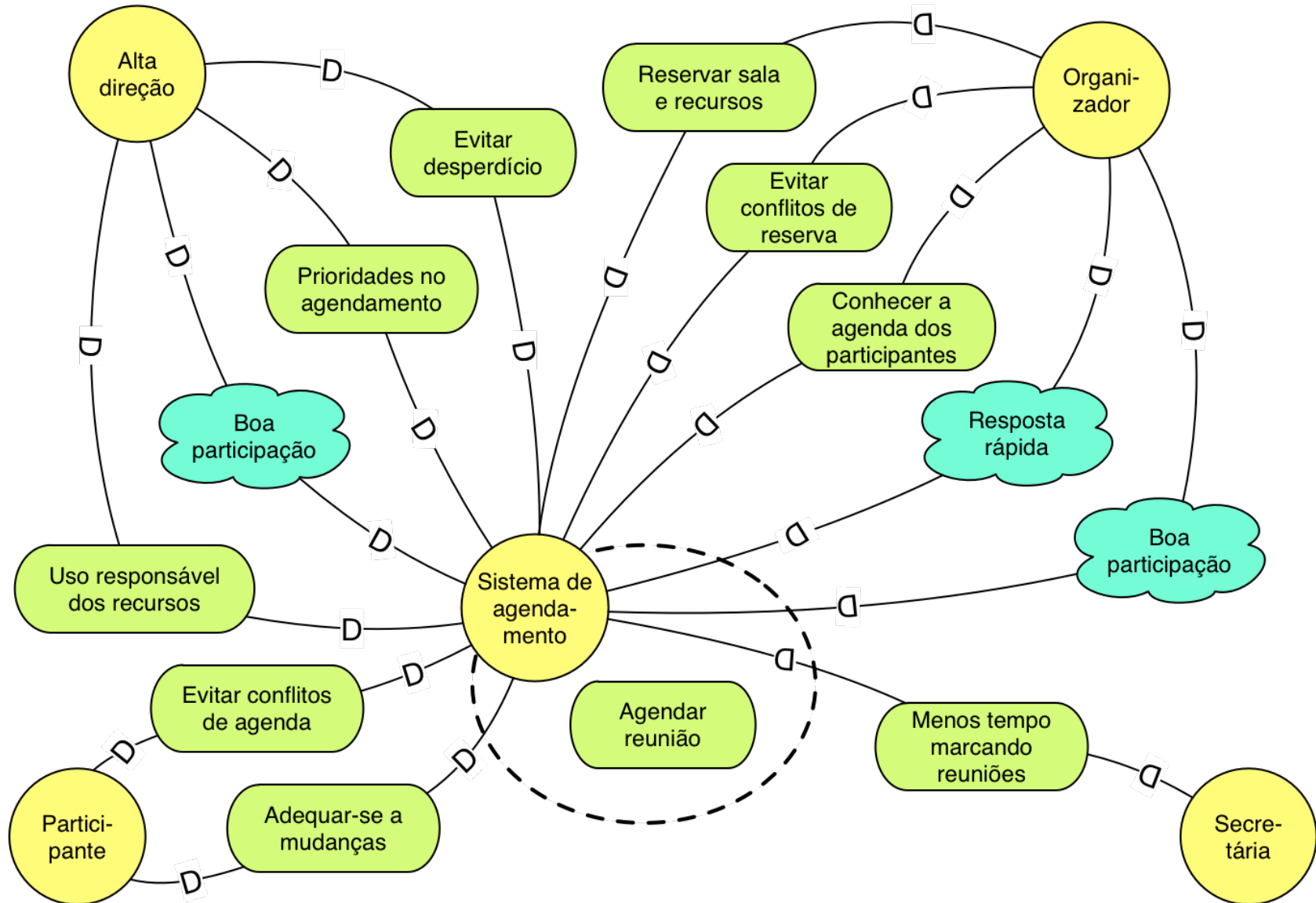


- Exercício: avalie o objetivo das secretárias “Reduzir a carga de trabalho de agendamento”, com relação aos critérios de qualidade: “Baixo custo” e “Eficácia”.

Exercício: comparação entre alternativas



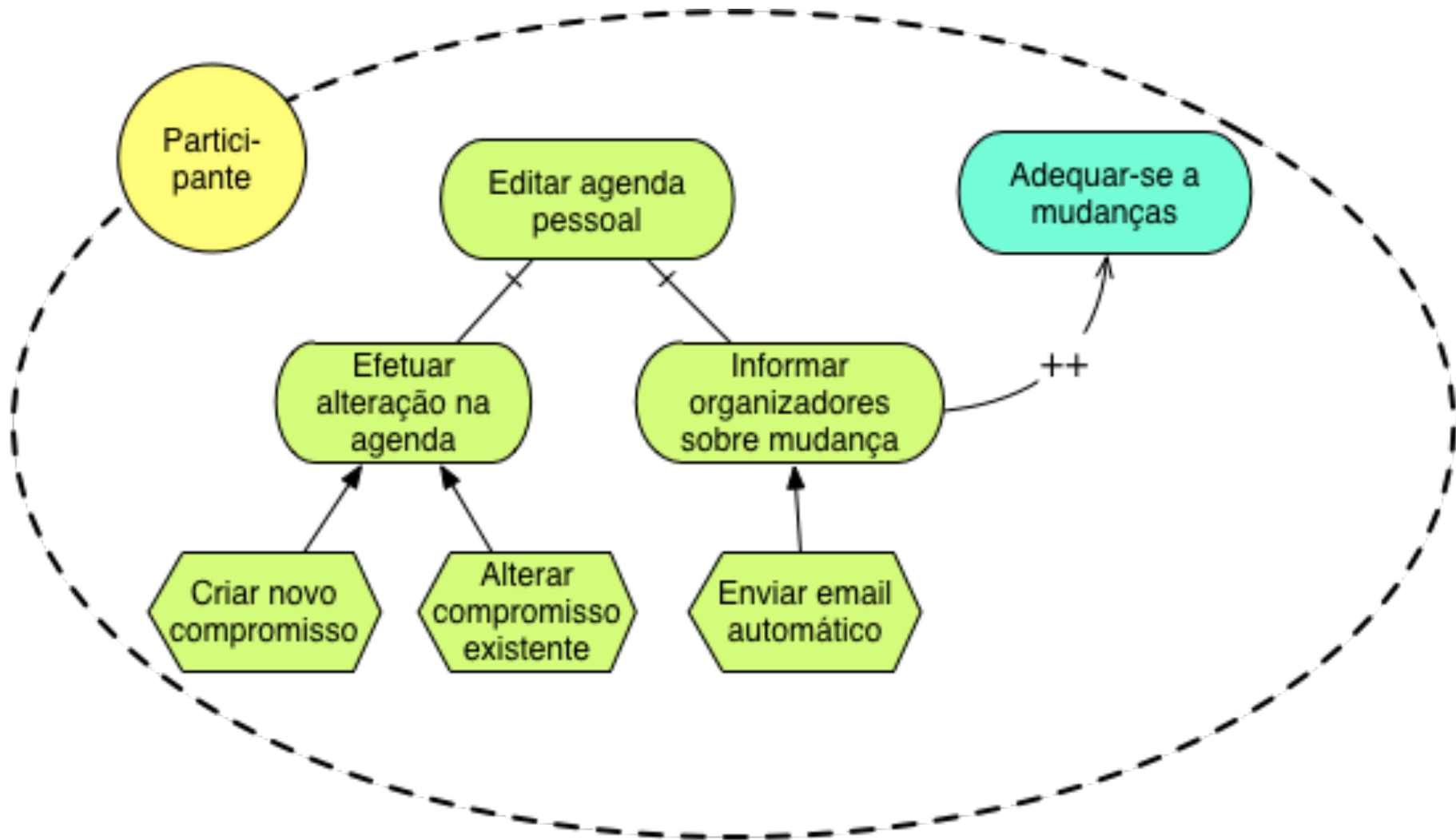
Exemplo: interação dos atores TO-BE



- Específicas:
 - OpenOME:
<http://sourceforge.net/projects/openome/>
- Genéricas:
 - LibreOffice Draw:
<https://www.libreoffice.org/discover/draw/>
 - OmniGraffle (Mac, não-gratuita):
<http://www.omnigroup.com/omnigraffle/>
 - Microsoft Visio (Windows, não-gratuita):
<http://office.microsoft.com/pt-br/software-de-diagramacao-profissional-microsoft-visio-FX103472299.aspx>.

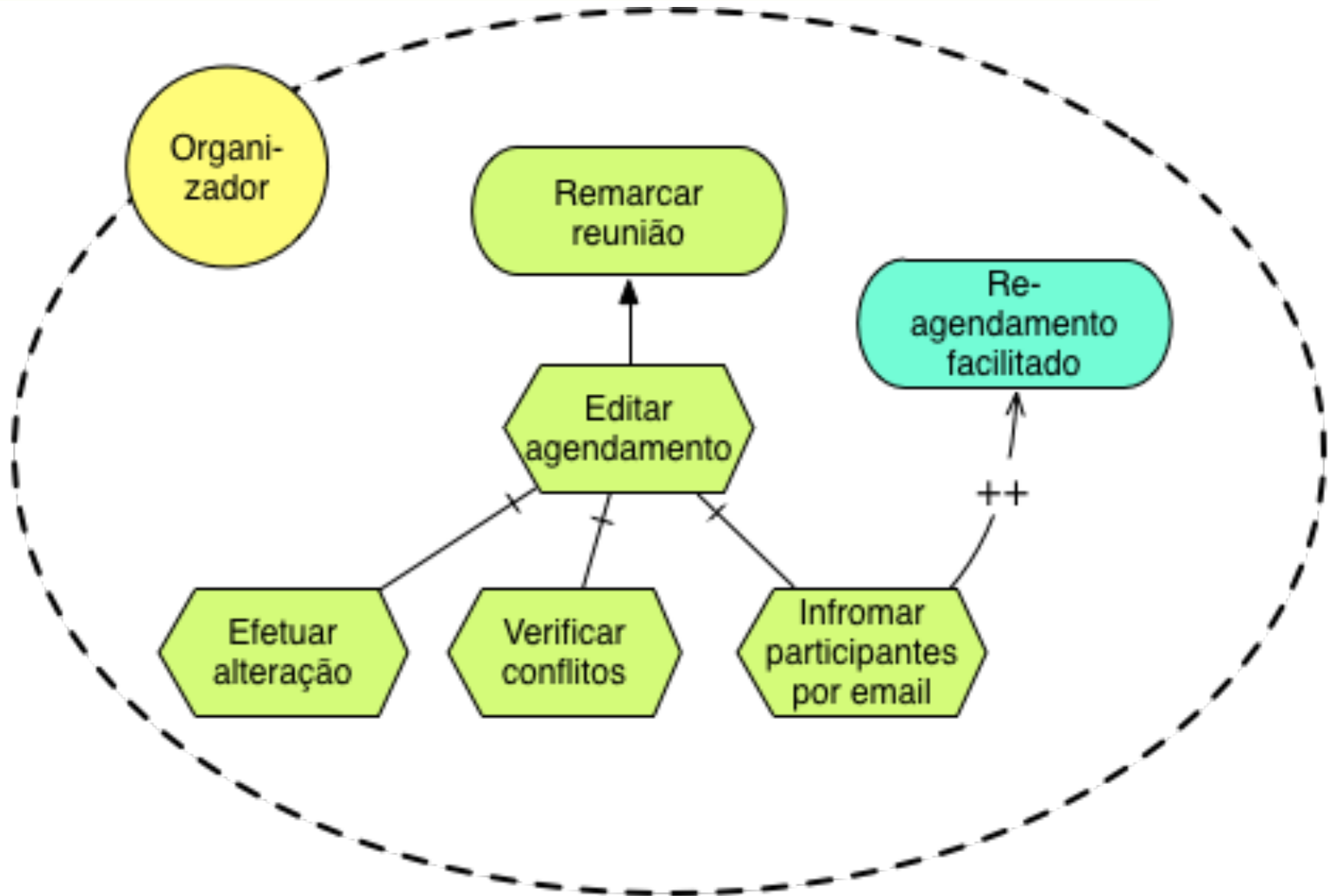
- Representam como os objetivos de um determinado ator são alcançados;
- Relação com modelo estratégico mostra:
 - Pertinência: um elemento tático (requisito) existe porque satisfaz um elemento estratégico (objetivo);
 - Completude: todos os elementos estratégicos (objetivos) são satisfeitos pelos elementos táticos (requisitos).
- Abordagem para construção:
 - Para cada objetivo estratégico, perguntar: o que o sistema deve fazer para satisfazer este objetivo?

Exemplo: modelo de objetivos tático

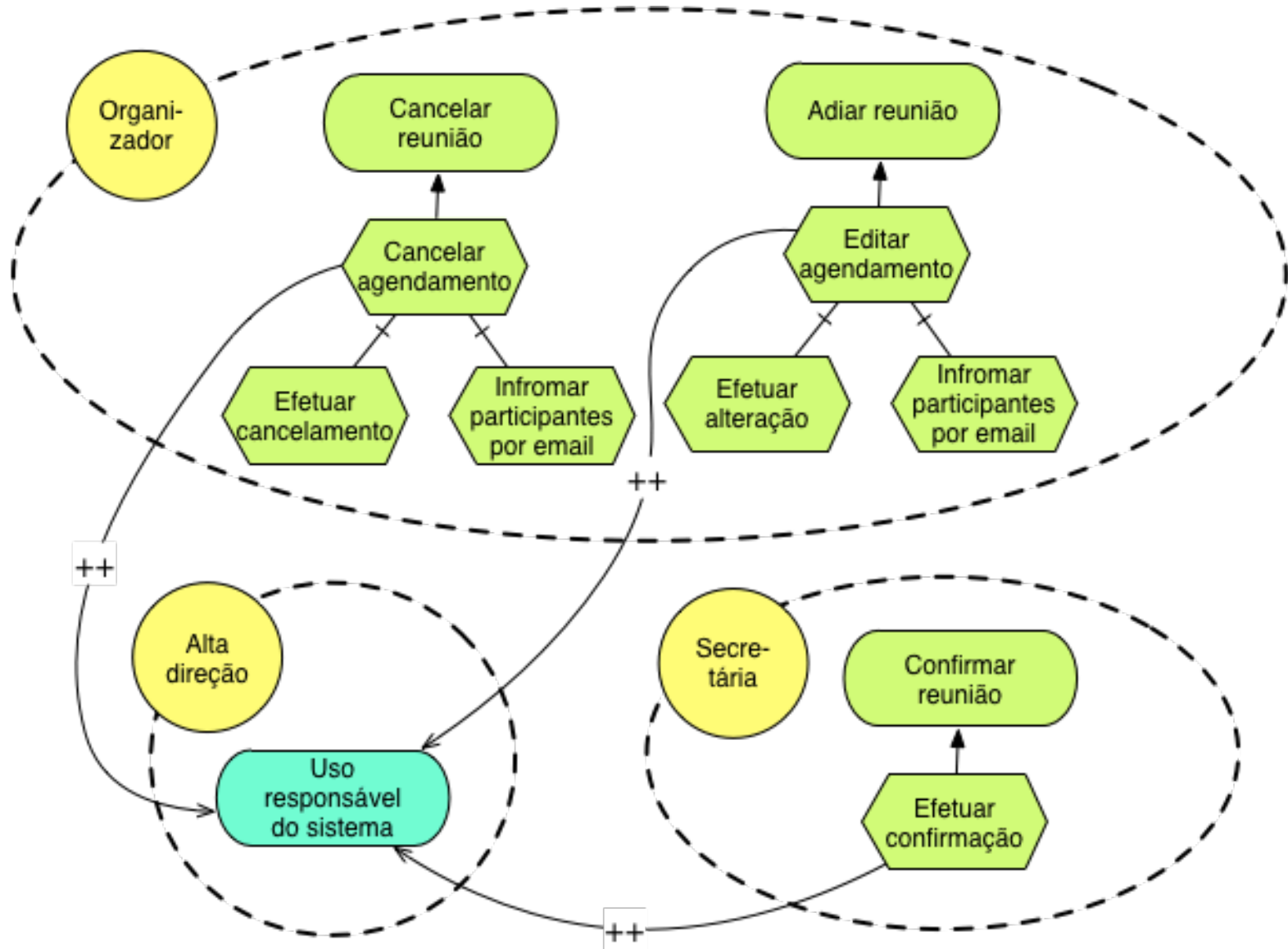


- Como podemos satisfazer o seguinte objetivo estratégico dos organizadores de reunião?
 - Re-agendamento facilitado em caso de mudanças.

Exemplo: modelo de objetivos tático

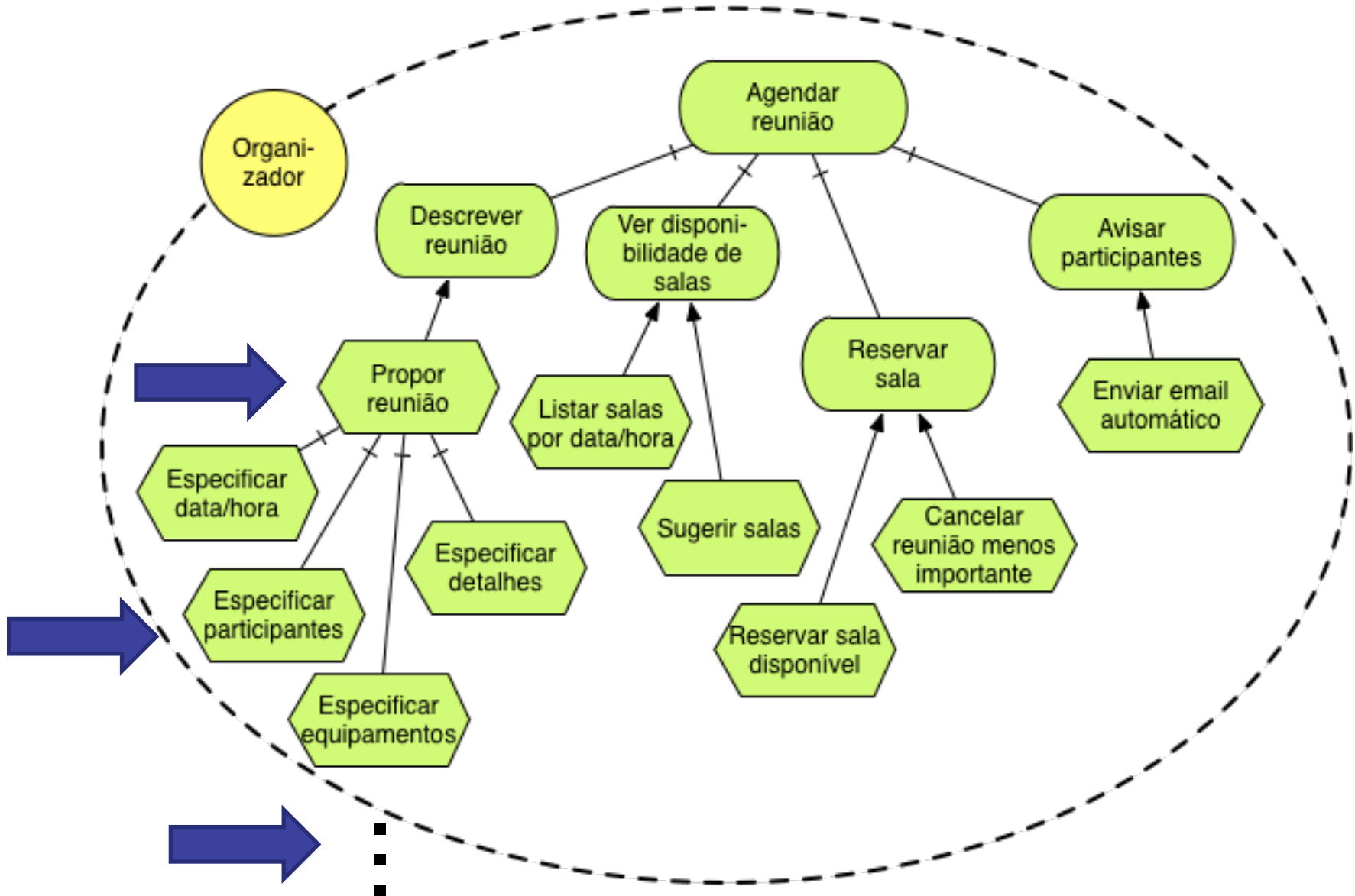


Exemplo: múltiplos atores envolvidos



- Objetivos estratégicos aumentam o nível de abstração, trazendo algumas vantagens;
 - Primeiro identificamos os *stakeholders*;
 - Em seguida levantamos seus objetivos até que se acredite que o modelo esteja completo;
- Objetivos táticos fazem a transição para um nível de abstração mais baixo (funcionalidades do sistema):
 - Para cada objetivo estratégico, verificar como ele pode ser satisfeito pelo sistema;
 - Continuar até que o modelo esteja completo, i.e., satisfaz todos os objetivos estratégicos.

- Descrever as funcionalidades do sistema utilizando cenários / casos de uso;
- Toda tarefa deve estar contemplada nos casos descritos, assegurando novamente a completude;
- A decomposição de tarefas, no entanto, cria vários níveis de granularidade;
- É preciso escolher qual nível será associado a um caso de uso.

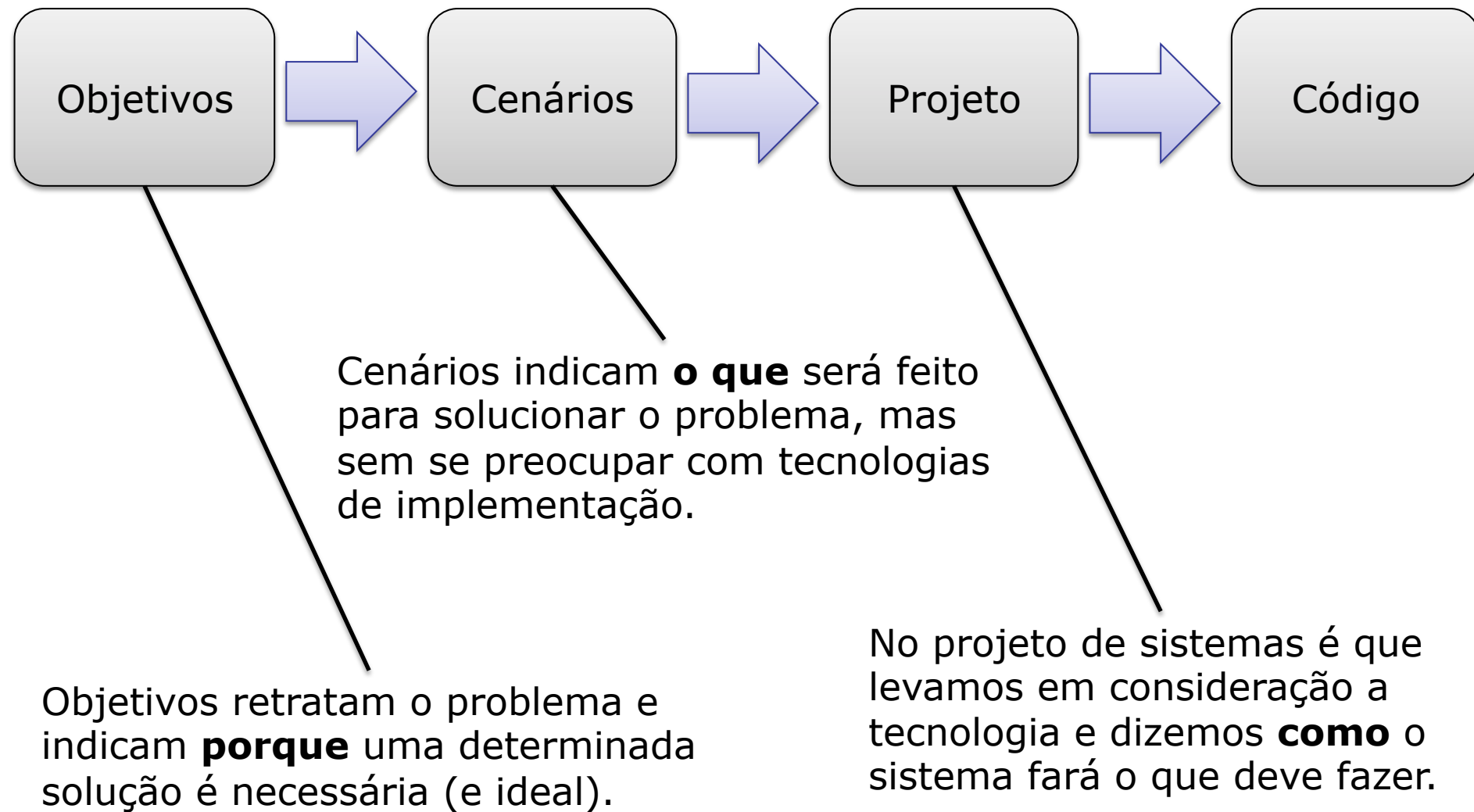


1. Dado um problema, analise sua descrição (use o problema entregue pelo professor ou o problema escolhido para seu trabalho prático, se quiser);
2. Identifique os *stakeholders*;
3. Identifique os objetivos dos *stakeholders* e crie modelos de objetivos na visão estratégica:
 - Como é o sistema atualmente (AS-IS)?
 - Quais são as alternativas para melhoria? Como avaliar estas alternativas segundo critérios relevantes?
 - Como você imagina o novo sistema (TO-BE)?
4. Crie modelos de objetivos na visão tática, mostrando como os objetivos dos *stakeholders* seriam satisfeitos.

Banco de Dados – Análise de Requisitos

CASOS DE USO

Por quê? → O que? → Como?



- Linguagem natural;
- Linguagem natural estruturada;
 - Formulários e modelos (*templates*).
- Linguagem de descrição de projeto;
 - Baseadas em linguagem de programação.
- Notações gráficas;
- Especificações matemáticas.

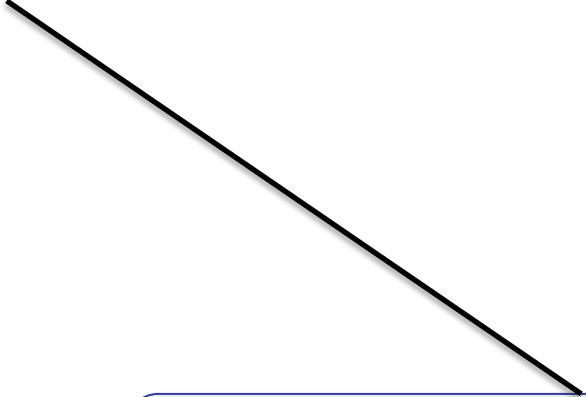
- Uma forma de estruturar requisitos:
 - Modelos gráficos e linguagem natural baseada em formulários;
 - Representam o que os usuários podem fazer no sistema;
 - São independentes do método de análise (OO, estruturado, etc.).

“Um caso de uso captura um contrato que descreve o comportamento do sistema sob várias condições a medida que ele responde a requisições de um de seus usuários.” (Alistair Cockburn)

“Um caso de uso conta uma história sobre como um usuário final (interpretando um de uma série de papéis) interage com o sistema dentro de um conjunto de circunstâncias.” (Roger Pressman)

“Um caso de uso especifica um comportamento de um sistema segundo uma perspectiva externa e é uma descrição de um conjunto de sequências de ações realizadas pelo sistema para produzir um resultado de valor observável por um ator.” (Grady Booch)

- São interações típicas entre o sistema e um ator – humano, outro sistema ou dispositivo;
- Capturam uma função visível ao ator;
- Busca atingir uma meta do usuário.



Por isso a ligação com o modelo de objetivos, que veremos à frente.

- Durante a análise de requisitos, trabalhamos com a ideia de sistema sócio-técnico;
- A partir do modelo de casos de uso, focaremos na parte técnica do sistema: software + hardware;
 - Cenários são implementados em software;
 - Componentes de hardware relevantes são representados como atores;
- Interações entre componentes humanos e organizacionais seriam representadas em outros modelos (fora do contexto dessa disciplina).

- Devem responder [Jacobson]:
 - Quem são os atores?
 - O que podem fazer no sistema?
 - Que pré-condições existem?
 - Quais as tarefas principais realizadas?
 - Que exceções devem ser consideradas?
 - Que variações são possíveis nas interações?
 - Que informações do sistema serão adquiridas, produzidas ou alteradas?

- Em resumo: representar o comportamento desejado do software (em termos de requisitos funcionais);
- Podem ser usados como base para:
 - Construção de casos de teste;
 - Estimativas de custo (cronograma) e tempo;
 - Identificação dos riscos;
 - Definição de prioridades;
 - Prototipação;
 - Manuais de usuário e documentação em geral.

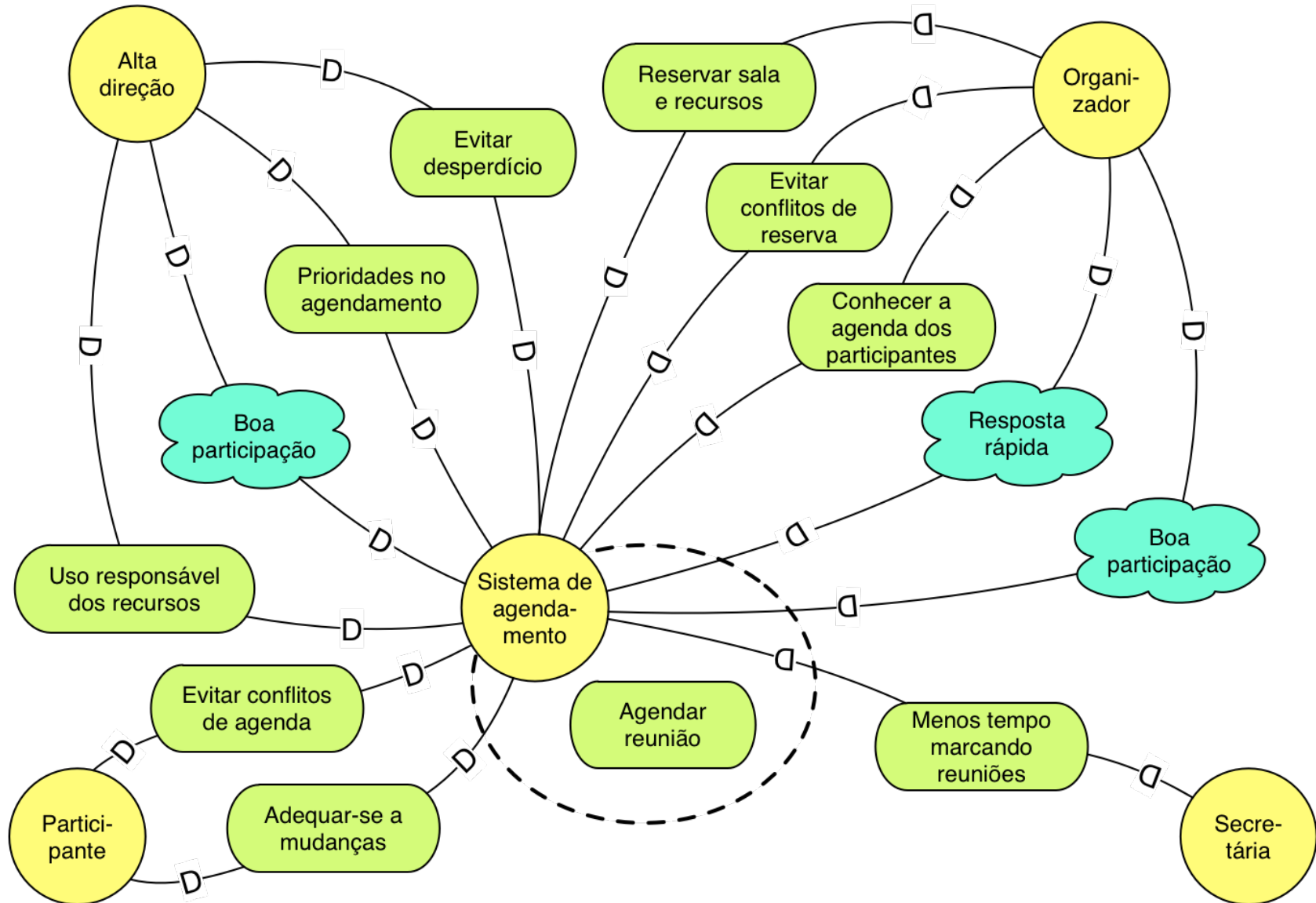
1. Identificação dos atores;
2. Captura dos casos de uso;
3. Criação de diagramas de casos de uso;
4. Elaboração da descrição de cada caso de uso;
5. Análise de possíveis associações entre casos de uso;
6. Separe os casos de uso em subsistemas.

1) Identificação dos atores

- Um ator é um papel específico que um usuário pode desempenhar;
 - Um mesmo usuário pode desempenhar vários papéis, cada hora sendo um ator diferente.
- Modela qualquer coisa externa que possa interagir com o software:
 - Usuários, outros softwares, dispositivos, etc.;
 - Delimitam o escopo do software;
 - Não é necessário ser descrito em detalhes (basta um parágrafo).

- Quem utiliza o software?
- Quem instala e mantém o software?
- Que outros softwares/dispositivos utilizam o software ou são utilizados por ele?
- Quem obtém informação do software?
- Quem provê informação ao software?
- O que o software faz automaticamente?

Exercício: quais os atores deste software?



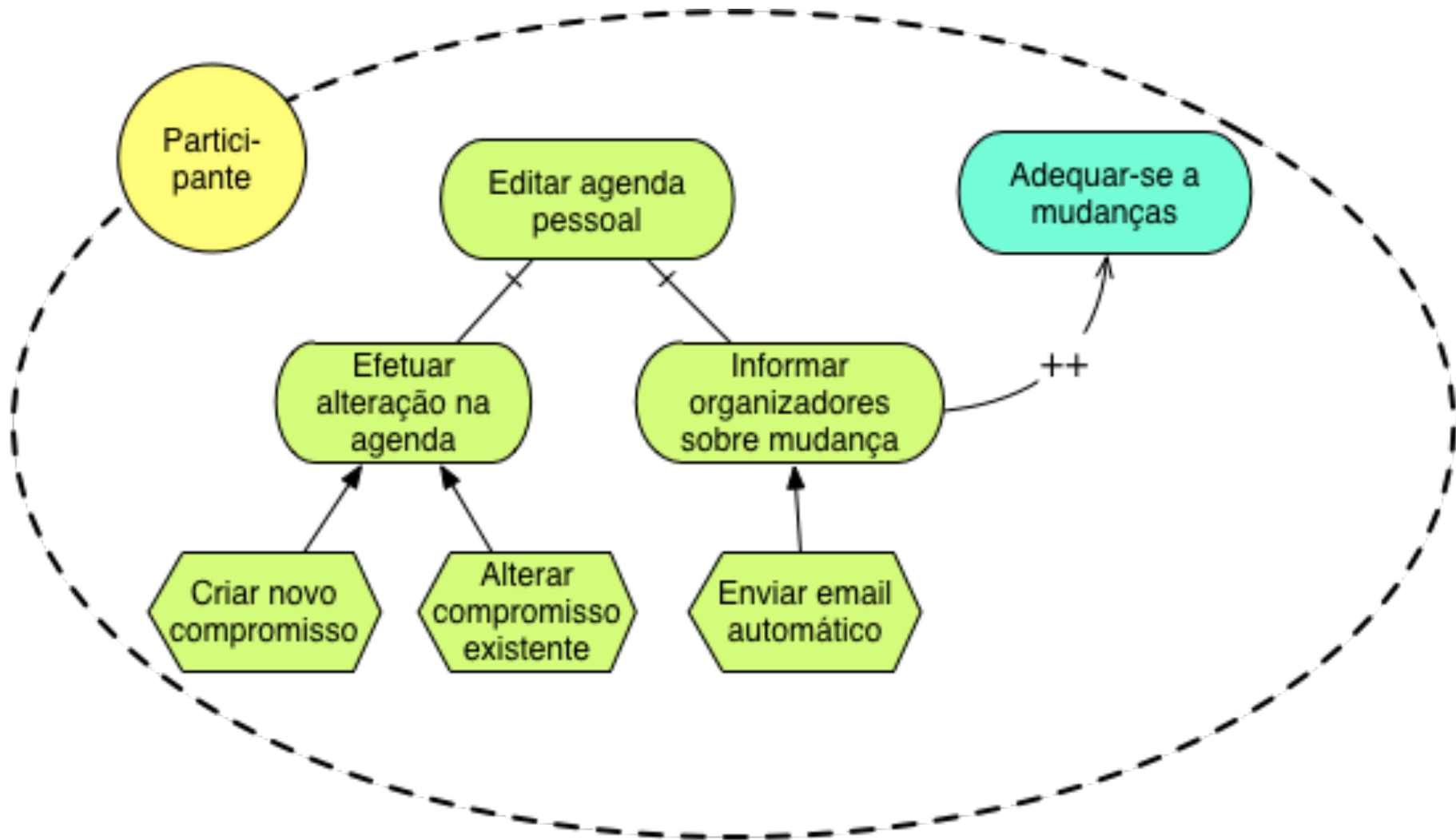
2) Captura dos casos de uso (UCs)

- Pode ser feita a partir dos modelos de objetivos;
 - Tarefas do modelo de objetivos tático que sejam implementadas em software;
- Pode ser feita durante a concepção (conversas iniciais) e elicitaco (entrevistas, etc.);
 - Identifique as interaoes discretas entre usurios e software;
- Geralmente so identificados em paralelo com a identificao dos atores;
- Alguns casos e atores podem ser capturados em fases mais avanadas.

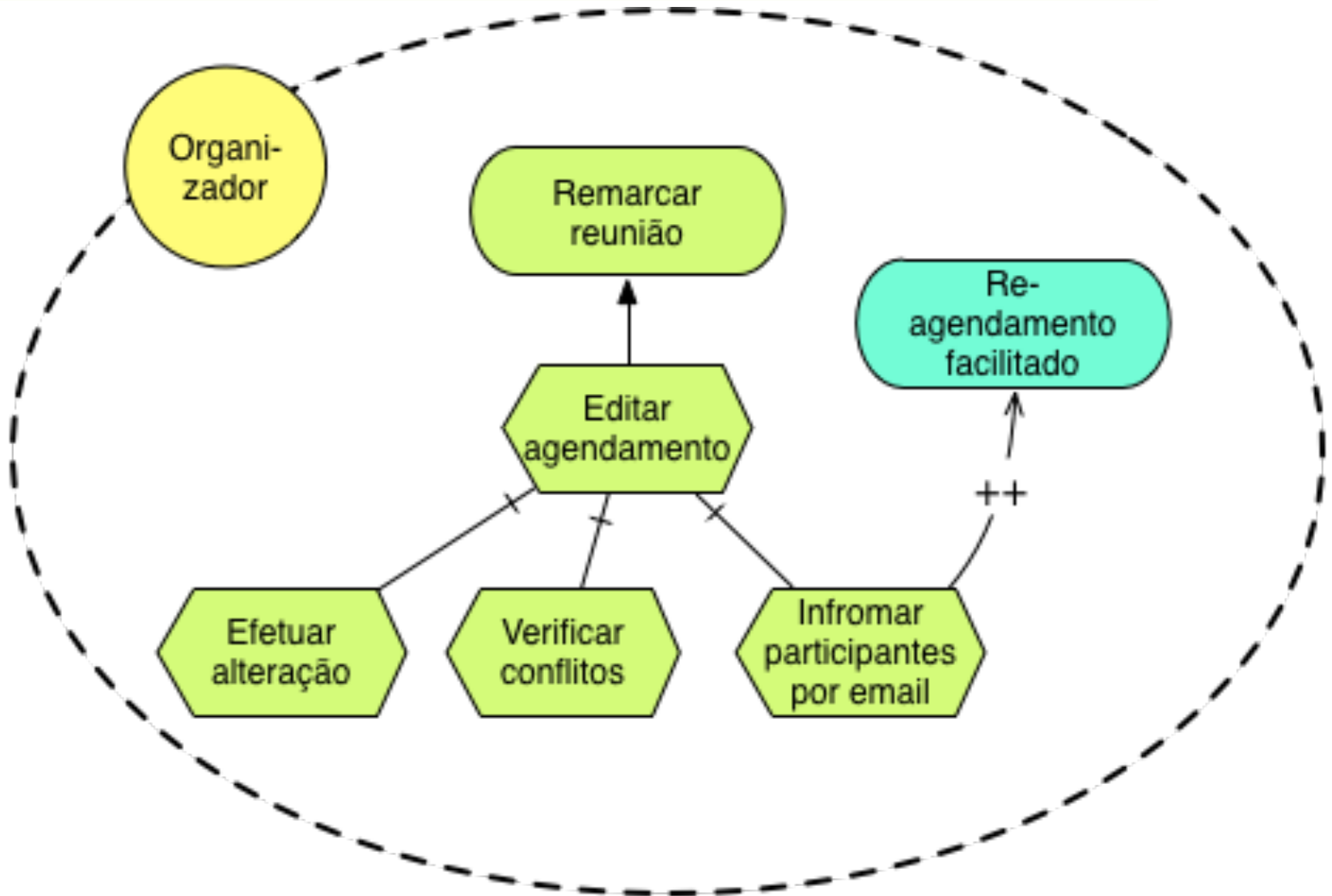
- Que funções o ator irá querer do software?
- O software armazena informações?
- Que atores irão criar, ler, atualizar ou apagar estas informações?
- O software precisa notificar algum ator sobre alguma mudança interna?
- Existem eventos externos que o software precisa estar ciente?
- Que atores informam o software sobre estes eventos?

- Casos de uso não devem ser muito pequenos nem muito grandes;
- Um bom caso de uso compreende uma sequência de transações realizadas pelo software que produzem *um resultado de valor observável para um particular ator.*

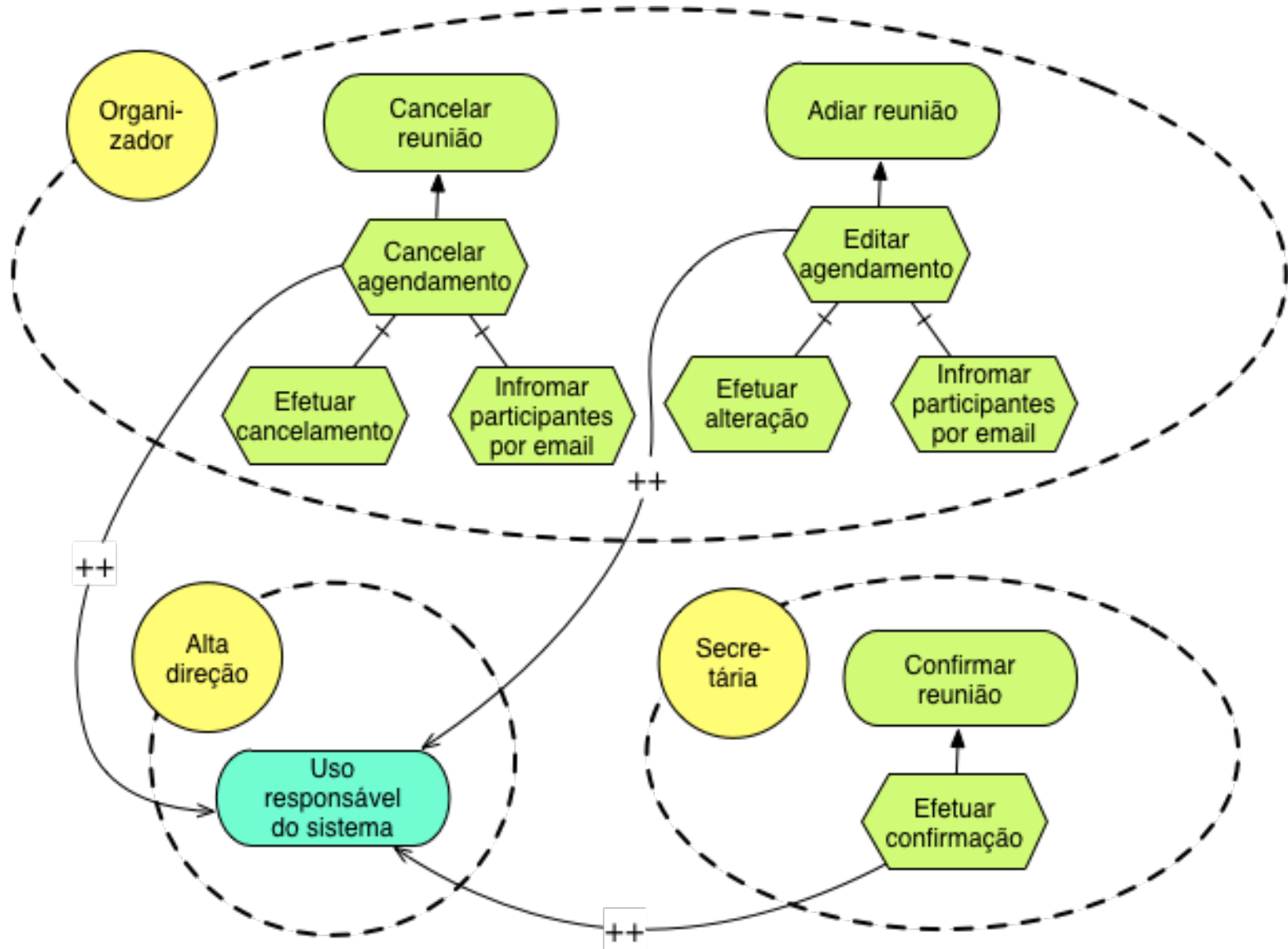
Exercício: que UCs podemos extrair daqui?



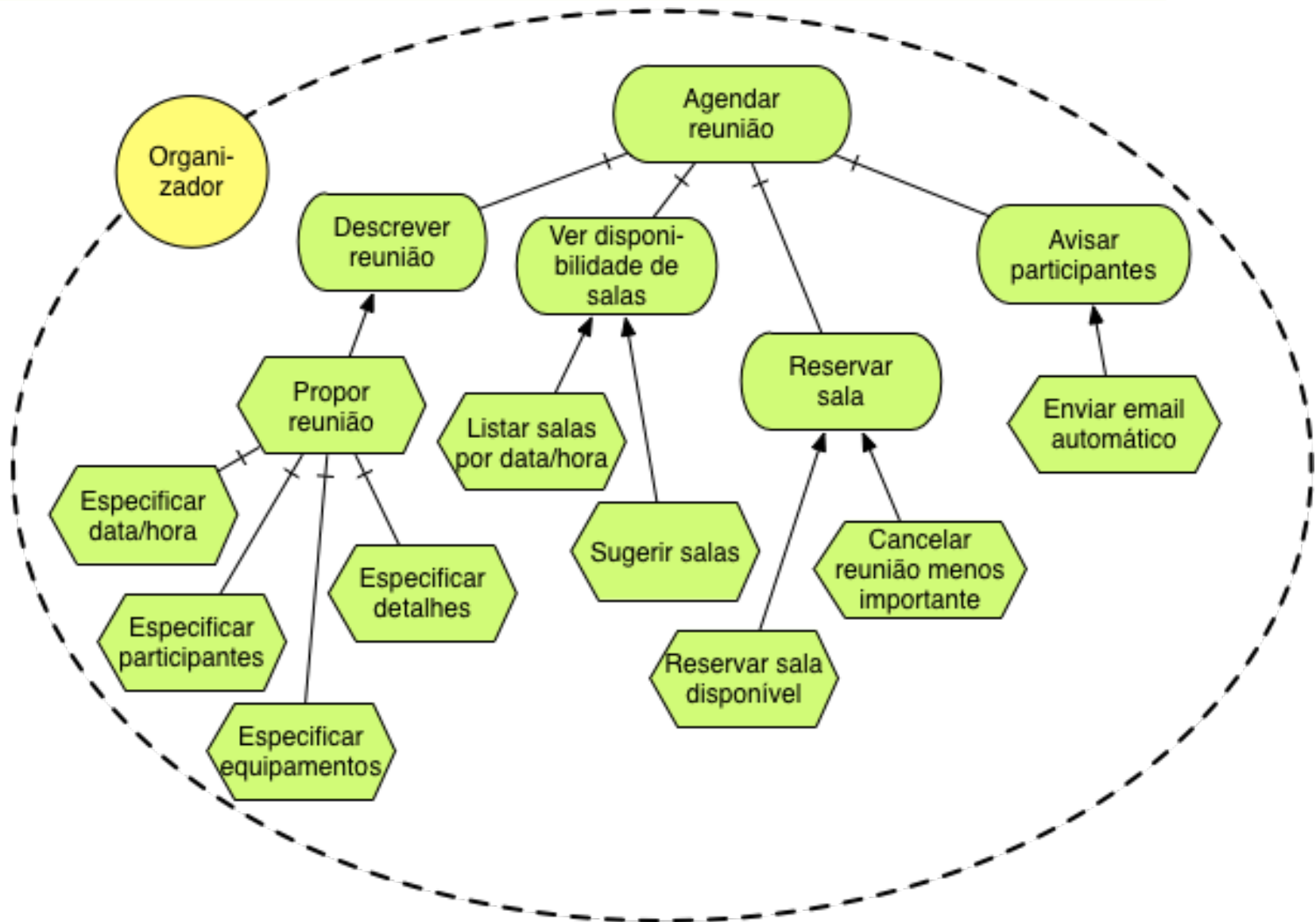
Exercício: que UCs podemos extrair daqui?



Exercício: que UCs podemos extrair daqui?

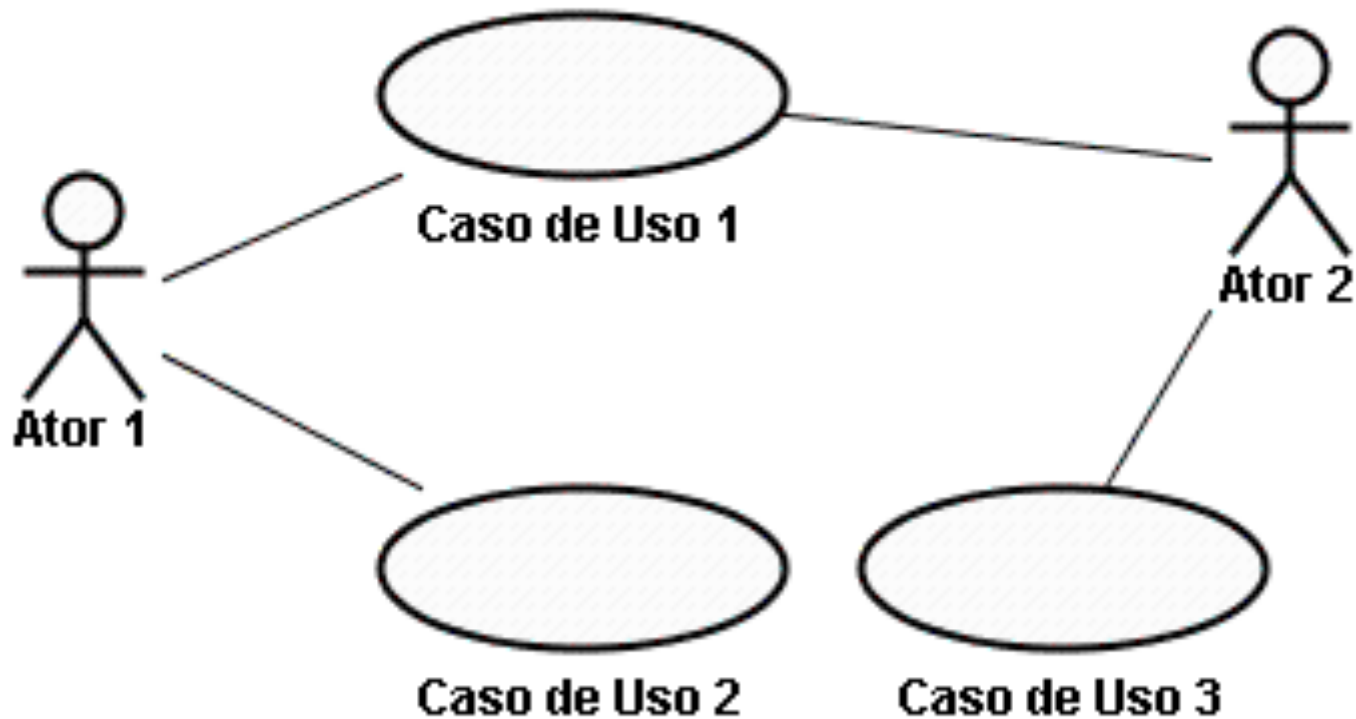


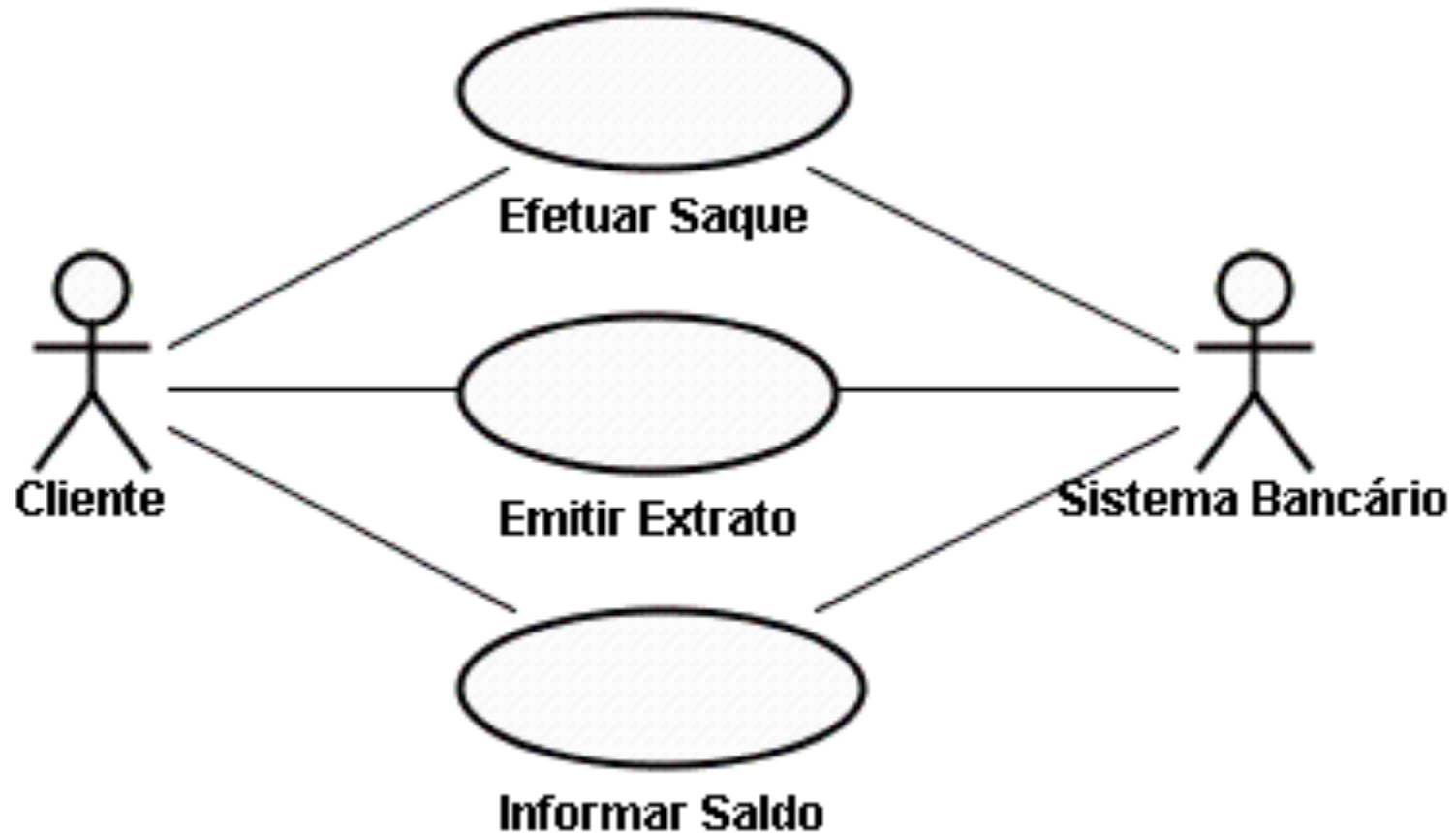
Exercício: que UCs podemos extrair daqui?



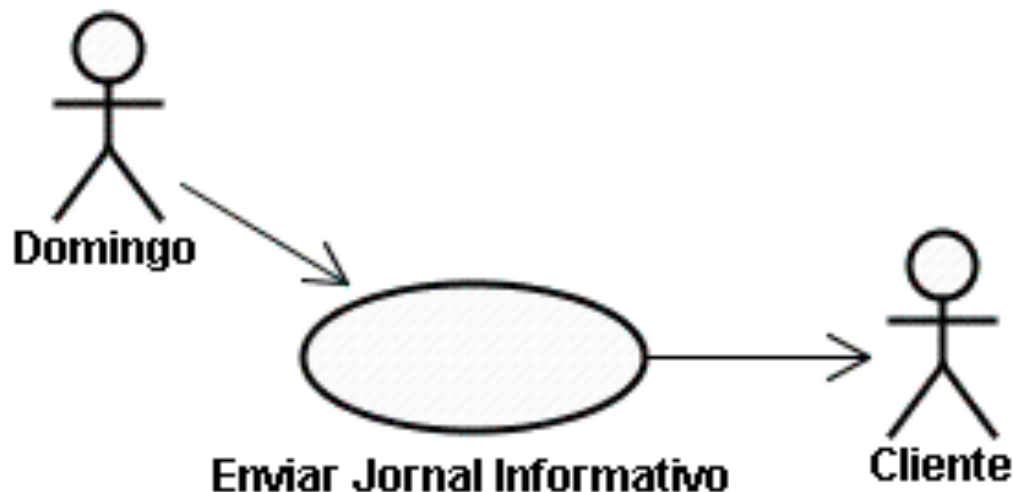
3) Diagramas de casos de uso

- Representam atores, casos de uso e suas associações;
- Uma associação entre um ator e um caso de uso significa que estímulos podem ser enviados entre atores e casos de uso, que se comunicam entre si;
- Proveem uma visão geral das funcionalidades do software.

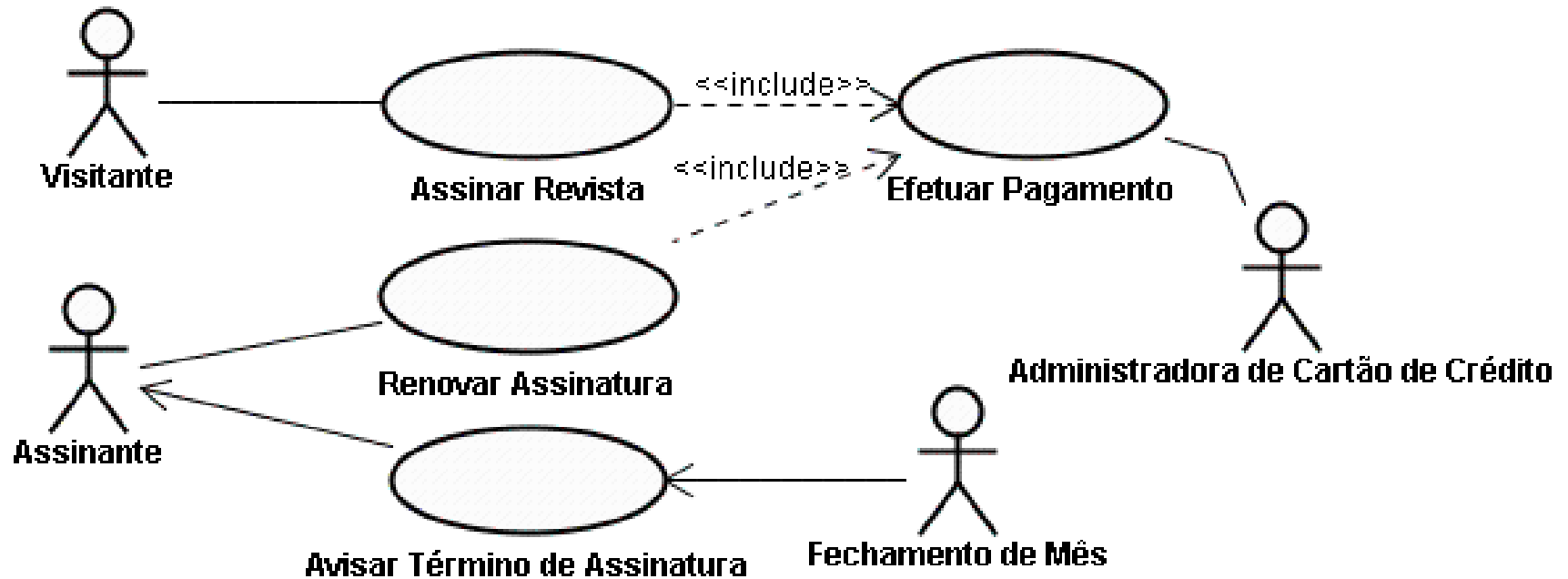




- Eventos podem ser disparados por determinadas condições, geralmente temporais:
 - Ex.: realizar backup a cada sexta-feira.
- Podemos mapear o evento como um ator ou tratá-lo como um elemento interno.



Diagramas de casos de uso - exemplo



- Represente os atores e os casos de uso do sistema de agendamento de reuniões em um diagrama.

4) Descrição dos casos de uso

- O diagrama é insuficiente para dizer o que cada caso de uso faz;
- Deve-se descrever textualmente o fluxo de eventos de cada caso separadamente;
- Esta tarefa deve ser iniciada após alguma estabilidade dos casos de uso, para evitar perda de tempo.

O que pode constar na descrição

- Identificador do caso de uso (UC1, UC2, etc.);
- Nome do caso de uso;
- Descrição breve / objetivos;
- Pré-condições e pós-condições;
- Entradas e saídas de dados;
- Fluxos (normal, alternativos, cenários);
- Classes/entidades participantes;
- Restrições de domínio;
- Requisitos não-funcionais associados;
- Outras observações.

- Curso normal: mundo perfeito, tudo ocorre como planejado;
- Cursos alternativos: exceções, erros, fluxos alternativos, etc.
- Para encontrá-los, analise o curso normal e pergunte, para cada item:
 - Tem alguma outra ação que pode ser feita?
 - Tem alguma coisa que pode dar errado?
 - Existe algum comportamento que pode ocorrer a qualquer momento?

- O ator sai da aplicação;
- O ator cancela a operação corrente;
- O ator pede ajuda;
- O ator provê dados inválidos;
- O ator provê dados incompletos;
- O ator escolhe uma maneira alternativa de realizar o caso de uso;
- O sistema falha;
- O sistema está indisponível.

- Curso Normal:
 - Parágrafos;
 - Lista numerada (preferível).
- Cursos Alternativos:
 - Intercalados no curso normal como itens;
 - Intercalados no curso normal como parágrafos;
 - Em uma seção separada, de forma resumida;
 - Em uma seção separada, de forma detalhada (semelhante ao curso normal).

- **Projeto:** RecMed
- **Subsistema:** controleRecomendacaoMedicao
- **Identificador do Caso de Uso:** UC10
- **Caso de Uso:** Registrar o Uso de Recomendação
- **Descrição Sucinta:** Este caso de uso permite o registro de informações sobre os resultados obtidos com a utilização das recomendações.

A documentação completa deste sistema pode ser vista em https://www.dropbox.com/sh/vud44xbrwz91xkf/X7_4sQnpYG, pasta "Exemplo TCC e Documentação Software", arquivo "RecMed - Documento_Especificacao_Requisitos_v1.0.docx"

- Fluxos de Eventos Normais:

Fluxo	Pré-condição	Descrição
Registrar o uso de uma recomendação	Nenhuma	<ol style="list-style-type: none">1. O usuário informa a recomendação para a qual deseja registrar os resultados de utilização;2. O usuário registra informações sobre os resultados obtidos com a utilização da recomendação;3. O usuário informa o contexto em que a recomendação foi utilizada (número de projetos e características dos projetos);4. O usuário indica se o resultado obtido com o uso da recomendação foi satisfatório;5. O usuário avalia a recomendação, registrando sua opinião sobre ela (indica se a recomendação é ótima, boa, regular, ruim ou péssima);6. O registro de utilização é armazenado e associado à recomendação selecionada pelo usuário no passo 1.

Fluxo	Pré-condição	Descrição
Registrar o uso das recomendações relaciona das a um aspecto	Nenhuma	<ol style="list-style-type: none">1. O usuário informa o aspecto (conjunto de recomendações) para o qual deseja registrar resultados de uso;2. As recomendações presentes no aspecto informado são apresentadas para o usuário;3. O usuário registra informações sobre os resultados obtidos com a utilização das recomendações do aspecto;4. O usuário informa o contexto em que as recomendações foram utilizadas (número de projetos e características dos projetos);5. O usuário indica se o resultado obtido com o uso das recomendações presentes no aspecto foi satisfatório;6. O usuário avalia o aspecto, registrando sua opinião sobre as recomendações presentes nele (indica se são ótimas, boas, regulares, ruins ou péssimas);7. O registro de utilização é armazenado e associado a todas as recomendações presentes no aspecto selecionado pelo usuário no passo 1.

- Fluxos de Exceção:

Nome do fluxo normal relacionado	Exceção	Descrição
Registrar o uso de uma recomendação	6. Dados inválidos (os dados fornecidos nos passos 2, 3, 4 ou 5 são inválidos ou não foram informados)	6.a Uma mensagem de erro é exibida solicitando ao usuário a correção dos dados.
Registrar o uso das recomendações relacionadas a um aspecto	7. Dados inválidos (os dados fornecidos nos passos 3, 4, 5 ou 6 são inválidos ou não foram informados)	7.a Uma mensagem de erro é exibida solicitando ao usuário a correção dos dados.

- Podemos agrupar casos de uso pequenos e similares num único caso de uso;
- Neste caso, temos um caso de uso e vários cenários;
 - Ex.: Cadastrar Cliente = Incluir + Consultar + Alterar + Excluir Cliente;
 - O exemplo anterior agrupou 2 cenários: “Registrar o uso de uma recomendação” e “Registrar o uso das recomendações relacionadas a um aspecto”;
- Agrupar ou não é decisão do analista;
- Levar em consideração legibilidade e organização.

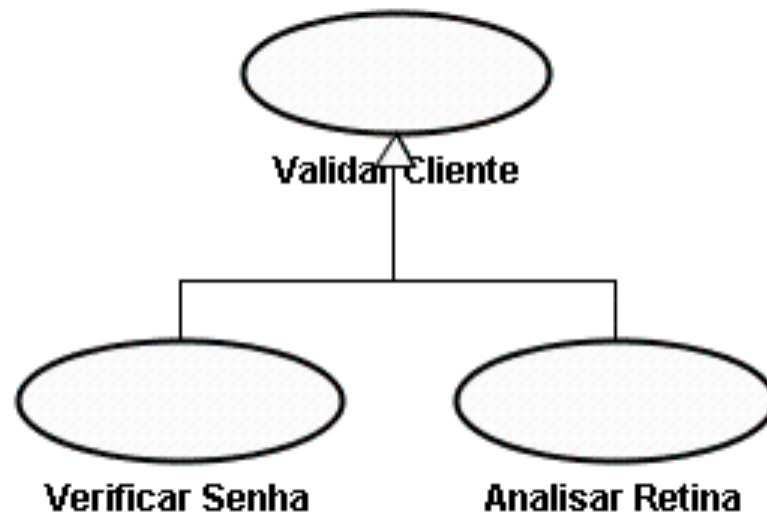
- Os casos de uso cadastrais (IACE: Incluir, Alterar, Consultar e Excluir) são muito comuns e similares;
- Pode-se fazer uma descrição tabular:
 - Identificador;
 - Caso de uso;
 - Ações possíveis (I, A, C, E);
 - Observações (dados a informar, restrições de integridade, fluxos alternativos);
 - Possíveis classes relacionadas.

Casos de uso cadastrais – exemplo

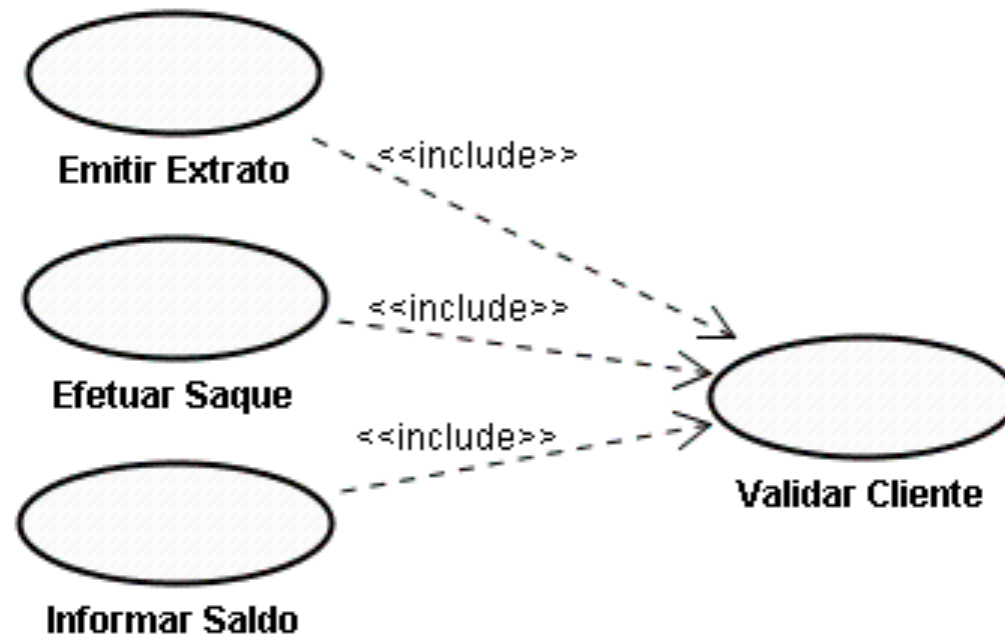
Id.	UC	Ações	Observações	Classes
UC 01	Cadastrar grupo	I, A, C, E	[I] Informar: nome, código e descrição; [E]: Grupos que possuem aspectos associados não podem ser excluídos; [A]: Caso o grupo esteja associado a um aspecto, é exibida uma mensagem informando isso ao especialista e é solicitada a confirmação da alteração.	Grupo, Aspecto
UC 02	Cadastrar aspecto	I, A, C, E	[I]: Informar: nome, grupo ao qual pertence, propósito e fundamentação teórica; [E]: Aspectos que possuem recomendações associadas não podem ser excluídos; [A]: Caso o aspecto esteja associado a uma recomendação, é exibida uma mensagem informando isso ao especialista e é solicitada a confirmação da alteração.	Grupo, Aspecto, RecomendacaoMedica

- Existem três tipos de associações que podem ser detectadas à medida que os diagramas de casos de uso são refinados:
 - Relação de especialização (herança);
 - Relação de inclusão;
 - Relação de extensão.

- Um caso de uso filho herda o comportamento e o significado do caso de uso pai;
- Acrescenta ou sobrescreve comportamento do pai e pode substituir o pai em qualquer lugar que este apareça.

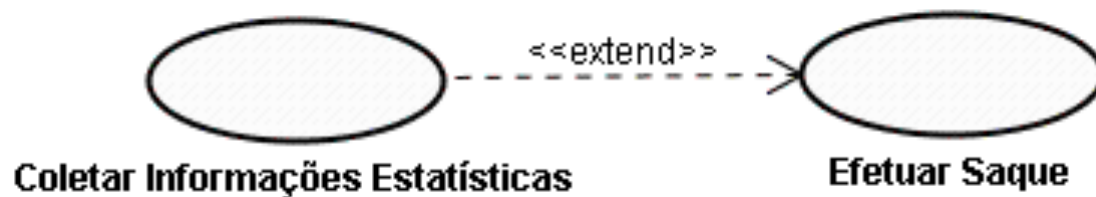


- Um caso de uso incorpora explicitamente o comportamento de outro;
- Funcionalidade comum é separada em um caso que é reutilizado por outros.



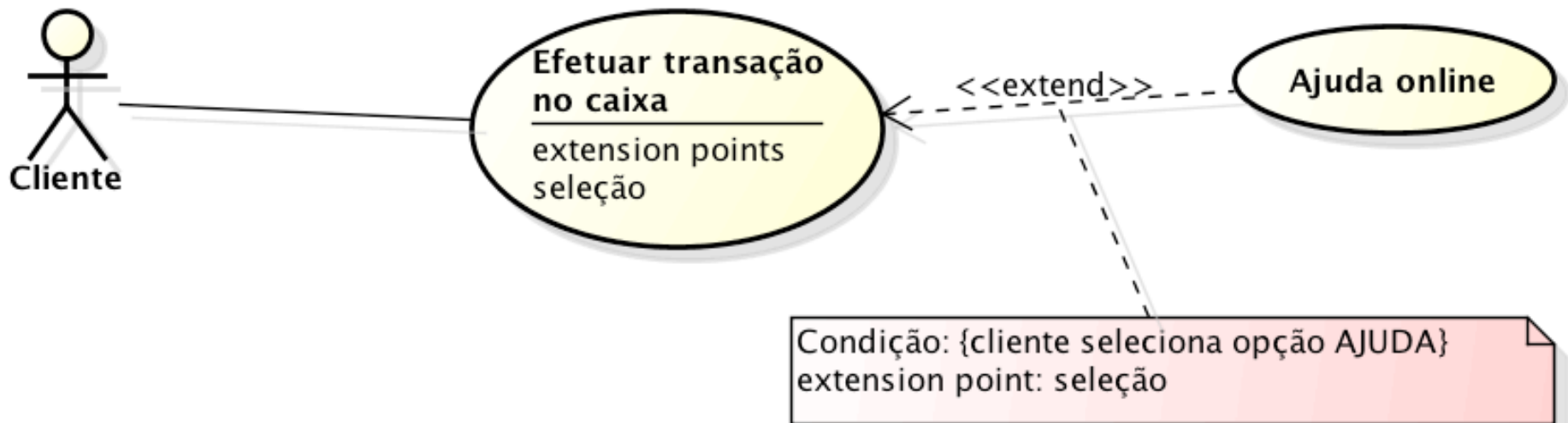
- Uma relação de inclusão entre dois casos de uso (UCs) significa que o comportamento definido no UC incluído é adicionado ao comportamento do UC base;
- A relação de inclusão foi feita para ser usada quando existem partes comuns do comportamento de dois ou mais UCs;
- Esta parte comum é então extraída e descrita em um UC separado, a ser incluído por todos os UC base que possuam esta parte em comum.

- Um caso de uso base incorpora implicitamente o comportamento de um outro caso de uso em um local especificado;
- Permite capturar os requisitos funcionais de um sistema de forma incremental.

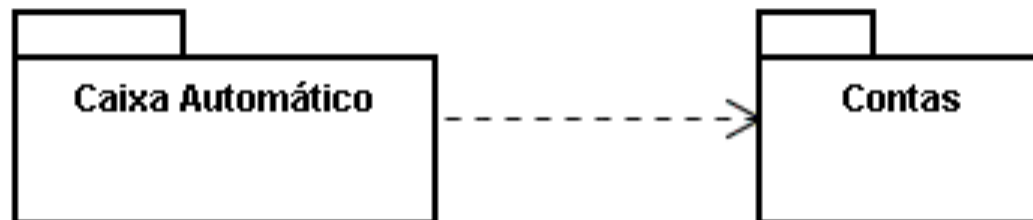


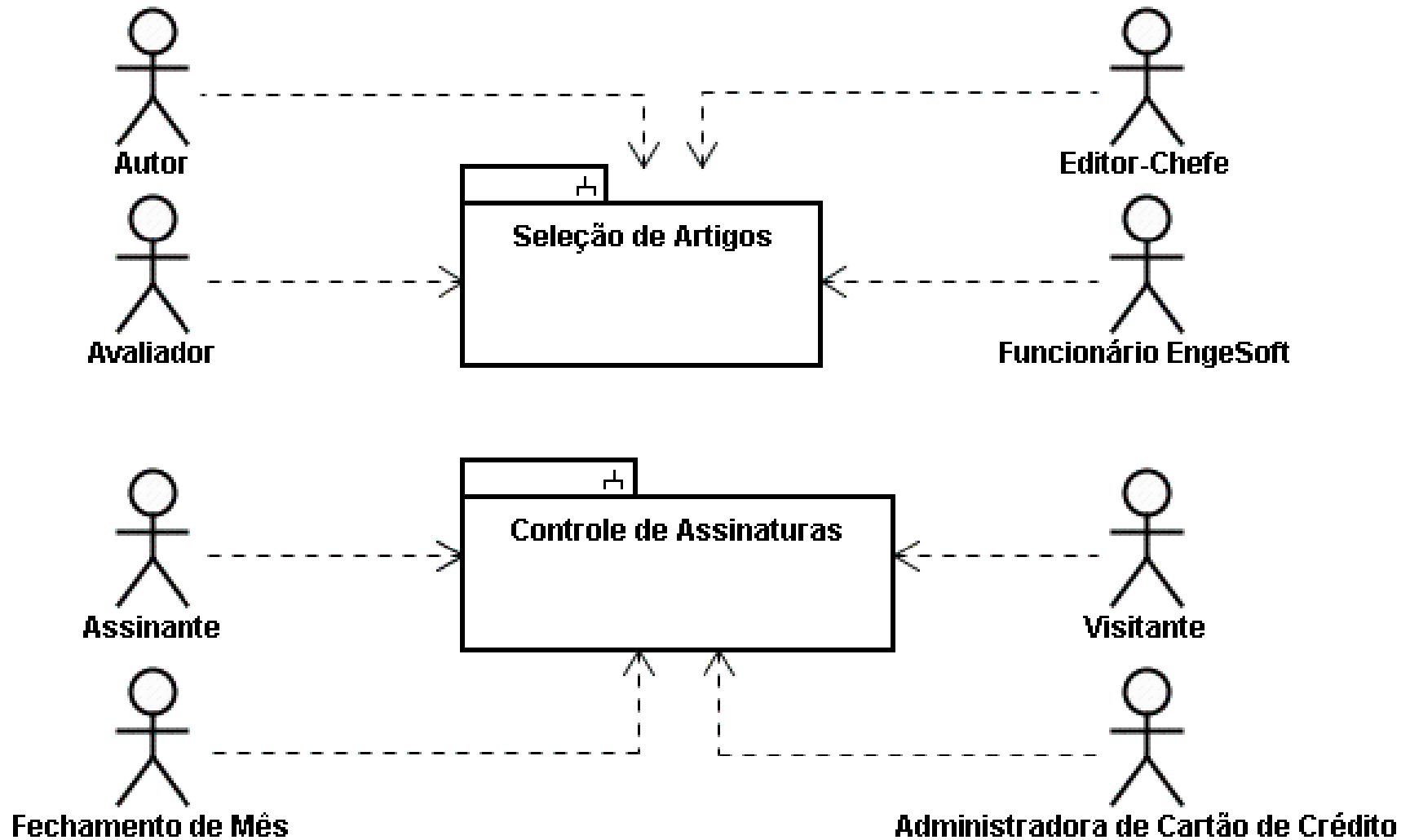
- Especifica que o comportamento de um caso de uso (UC) pode ser estendido pelo comportamento de outro UC (de maneira geral, suplementar);
- A extensão se dá em um ou mais pontos de extensão específicos definidos no UC estendido;
- O UC estendido é definido independente do UC extensor. Ele possui significado independentemente;
- Por outro lado, o UC extensor tipicamente define comportamento que não necessariamente possui sentido sozinho;
- Ao invés, o UC extensor define um conjunto de comportamentos modulares que incrementam e melhoram a execução do UC estendido em condições específicas.

- Extensão: quando estiver descrevendo uma variação de um curso normal, comportamento opcional;
- Inclusão: quando houver repetição de um mesmo fluxo em dois ou mais casos de uso e se quer evitar isso.



- Facilita o entendimento e a leitura;
- Utiliza-se o ícone de pacote da UML;
- Setas pontilhadas indicam dependência – um pacote solicita serviços de outro (inclusão, extensão, etc.).



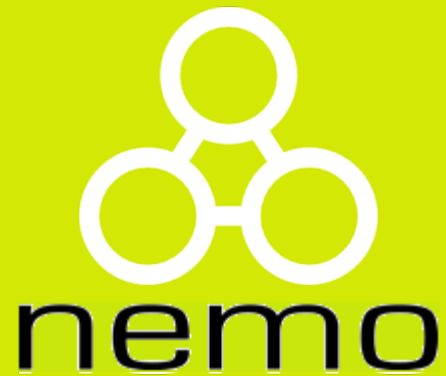


- Uma ferramenta CASE auxilia no desenho de diagramas de caso de uso;
- Várias disponíveis:
 - Astah community (<http://astah.net/editions/community>) – nossa sugestão;
 - Visual Paradigm (<http://www.visual-paradigm.com>);
 - Magic Draw (<http://www.nomagic.com/products/magicdraw.html>);
 - Papyrus (<http://www.eclipse.org/papyrus/>);
 - Dentre outras...



- Verifique o padrão de nomenclatura antes de começar:
 - Atores;
 - Casos de uso;
 - Pacotes;
 - Descrição do caso de uso.
- Para a descrição, use o modelo.

- A partir de modelos de objetivos de nível tático (utilize o projeto Pizza a Pezzi ou seu trabalho prático), produza diagramas de caso de uso, identificando:
 - Atores;
 - Casos de uso (UCs);
 - Relações ator x UC, UC x UC;
 - Para os UCs cadastrais, identifique as ações possíveis e as restrições de integridade associadas.



<http://nemo.inf.ufes.br/>