

Elicitação

Modelagem

Análise

# Requisitos – Aula 26

Professores: Milene Serrano e Maurício Serrano

# Agenda

- › Considerações Iniciais
- › Pós Rastreabilidade
  - Gerência de Desenvolvimento de Software orientada à *baseline* de Requisitos
- › Em resumo...
- › Considerações Finais

# Considerações Iniciais

# Considerações Iniciais



Planos de  
Negócio e  
Outros  
Documentos

Negócio

Pré-Rastreabilidade



Documento  
de Requisitos

Engenharia de  
Requisitos

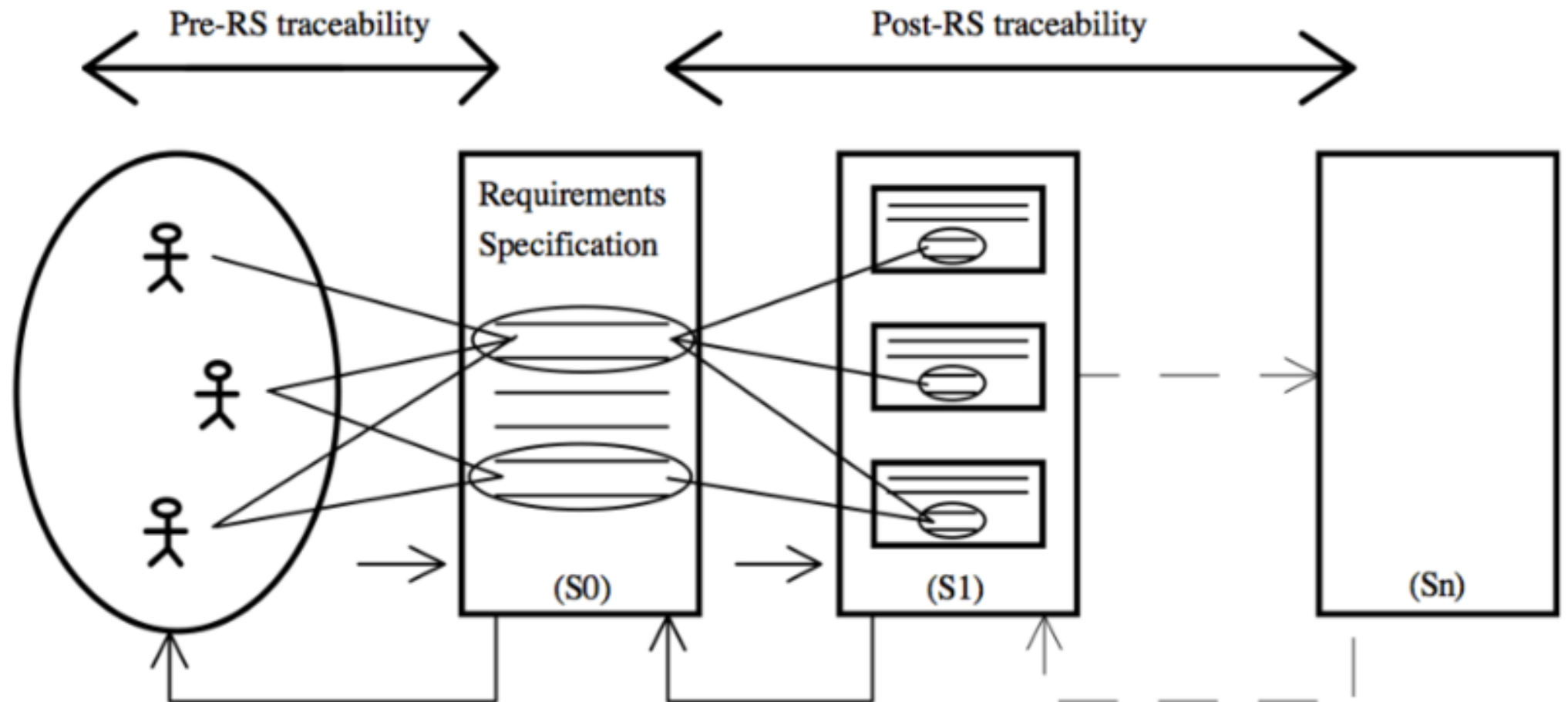
Pós-Rastreabilidade



Artefatos de  
Desenho e  
Implementação

Desenho e  
Implementação

# Considerações Iniciais



# Considerações Iniciais

Rastreabilidade *backward-from* (para trás, a partir de), ligando requisitos às suas fontes.



Rastreabilidade *forward-to* (para frente), ligando documentos obtidos no processo de elicitação no plano de negócio a requisitos relevantes.

Rastreabilidade *backward-to* (para trás ou de volta), ligando artefatos de desenho e implementação de volta a requisito.



Rastreabilidade *forward-from* (para frente, a partir de), ligando requisitos a artefatos de desenho e implementação.

# Considerações Iniciais

Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

Portanto, podemos entender RASTREABILIDADE como:

- Uma característica de sistemas, nos quais requisitos são claramente ligados às suas fontes bem como aos artefatos criados durante o ciclo de vida de desenvolvimento do sistema. Lembrando que a criação desses artefatos é orientada à própria *baseline* de requisitos.

Assim:

- Um elo é estabelecido entre as mudanças realizadas diante das necessidades dos usuários e a evolução dos sistemas de computação, estabelecendo-se uma base para o gerenciamento do conhecimento organizacional.
- **REQUISITOS NÃO PODEM SER EFETIVAMENTE GERENCIADOS SEM RASTREABILIDADE.**

# Considerações Iniciais

## Pré-rastreabilidade:

- Requisitos sendo ligados às suas fontes...
- Já estudamos, no começo do nosso curso.
- Exemplos de suporte: RichPicture e Argumentação.

## Pós-rastreabilidade:

- Requisitos sendo ligados a artefatos que são criados durante o ciclo de vida de desenvolvimento do sistema.
- Também conhecida como: Gerência de Desenvolvimento de Software orientada à *baseline* de Requisitos.
- Iremos conhecer na aula de hoje...



# Pós-Rastreabilidade

Gerência de Desenvolvimento de Software  
orientada à *baseline* de Requisitos

# Mudanças

Hammer indica 1% ao mês entre a chegada de novos requisitos bem como mudanças naqueles que já foram definidos. Isso para projetos bem estáveis...

Outros autores chegam a apontar um percentual de mudanças de até 50% dos requisitos, antes que o sistema seja colocado em produção.

Fontes geradoras de mudanças são os próprios interessados, que, no início do projeto, não têm muita clareza quanto aos objetivos a serem alcançados. Ao longo da evolução do desenvolvimento, são descobertas novas demandas a serem incluídas no projeto.

# Cascata?

Por isso, o ciclo de vida Cascata é tão criticado na literatura. Ele não prevê tratamento para alterações nos requisitos durante o processo de desenvolvimento.

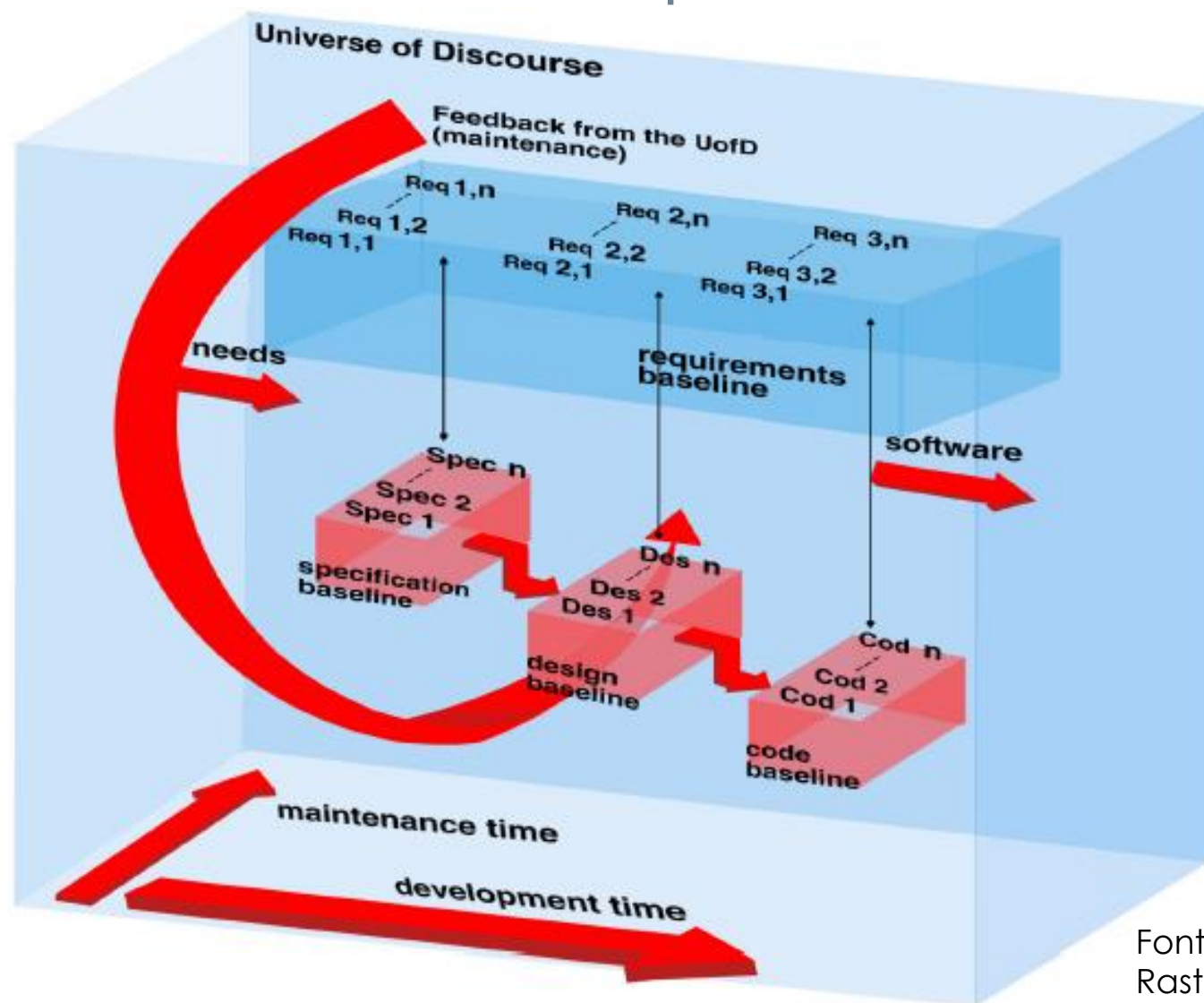
# Baseline de Requisitos e Evolução

O modelo – registrado na baseline de requisitos – deve evoluir em consonância com o sistema e o Universo de Informações.

A *baseline* torna-se referência para o desenvolvimento, orientando as várias atividades relacionadas à geração do sistema desejado.

Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

# Baseline de Requisitos e Evolução



**Eixo Longitudinal:** registra a evolução da *baseline* em cada etapa do processo de desenvolvimento

**Eixo Horizontal:** evolução da *baseline* ao longo do processo de desenvolvimento

# Rastreabilidade

Gerentes e desenvolvedores podem ser auxiliados pela rastreabilidade em vários aspectos, tais como:

- Verificação da alocação de requisitos a componentes de software, sendo possível identificar requisitos que ainda não foram alocados ou implementados.
- Resolução de requisitos em conflito, permitindo identificar as origens dos requisitos em conflito, e buscando uma solução em consenso.
- Estimativa de possíveis variações nos cronogramas bem como nos custos do desenvolvimento. Consequentemente, tal previsão permite gerenciar mais adequadamente os riscos do projeto.
- Verificação orientada a casos de testes, identificando requisitos para os quais não foram previstos casos de teste, complementando essa *baseline*.

# Rastreabilidade

Continuação...

Gerentes e desenvolvedores podem ser auxiliados pela rastreabilidade em vários aspectos, tais como:

- Correção de defeitos, identificando as origens desses defeitos, as quais podem não estar no código, necessitando rever outros artefatos.
- Validação, possibilitando demonstrar aos clientes que a implementação atende à *baseline* de requisitos.
- Melhorias de hardware e/ou de sistema operacional, pois existe uma relação muito forte entre componentes de hardware e software. Os elos de rastreabilidade agilizam a identificação de componentes atingidos por mudanças de versão do sistema operacional e/ou na troca de hardware.

# Rastreabilidade

Continuação...

Gerentes e desenvolvedores podem ser auxiliados pela rastreabilidade em vários aspectos, tais como:

- Reutilização de componentes, sendo os elos de rastreabilidade fortes aliados na criação de bibliotecas de componentes, pois deixam evidentes as correlações entre código e demais artefatos, sejam de desenho e/ou de requisitos. Dada a observância dos mesmos problemas ou das mesmas demandas, em um projeto similar, perceberá que é possível fazer uso das mesmas soluções, ou seja, dos mesmos componentes. Isso proporciona maior produtividade.



# Rastreabilidade

Portanto...

- Rastreabilidade não é apenas um fator que auxilia a gerência dos requisitos. Adicionalmente, rastreabilidade pode ser vista como um fator que auxilia a **GERÊNCIA DA QUALIDADE**.

# Tipos de Elos da Rastreabilidade

Os principais meta-modelos que permitem debater sobre os elos da rastreabilidade, são:

- Meta-modelo de Toranzo (será visto mais adiante), e
- Meta-modelo de Ramesh&Jarke (mais abstrato e mais confuso para especificar o rastro)

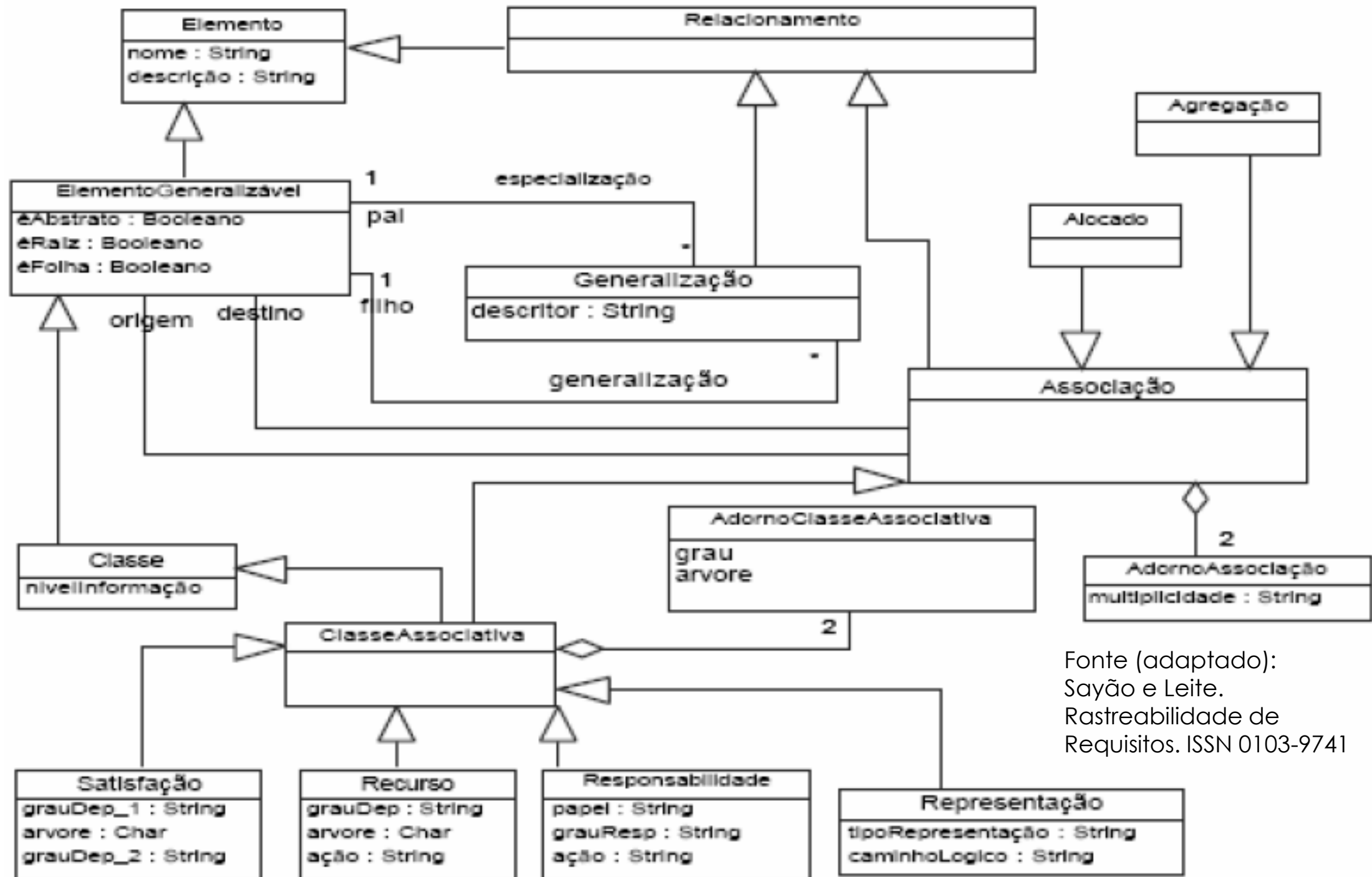
Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

# Tipos de Elos da Rastreabilidade

## Meta-modelo de Toranzo:

Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

- As informações a serem rastreadas devem ser classificadas em quatro níveis:
  - **Ambiental:** informações oriundas do contexto no qual a organização está inserida;
  - **Organizacional:** informações pertencentes à organização (missão, objetivos e estratégias);
  - **Gerencial:** informações que auxiliam a gerência do projeto, e
  - **Desenvolvimento:** informações associadas aos diversos artefatos gerados ao longo do processo de desenvolvimento (artefatos de requisitos, diagramas, códigos, casos de teste e outros).



Fonte (adaptado):  
Sayão e Leite.  
Rastreabilidade de  
Requisitos. ISSN 0103-9741

# Tipos de Elos da Rastreabilidade

Cuidado que  
Classe aqui não  
representa o  
conceito de classe  
da notação da  
UML

## Meta-modelo de Toranzo:

Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

- Os principais elos de rastreabilidade são:
  - **Satisfação**: classe origem tem dependência de satisfação com a classe destino.
  - **Recurso**: classe origem tem dependência de recurso com a classe destino.
  - **Responsabilidade**: registra a participação, responsabilidade e ação de pessoas sobre artefatos.
  - **Representação**: captura a representação ou modelagem dos requisitos em outras linguagens.
  - **Alocado**: classe origem está relacionada à classe destino, que representa um subsistema.
  - **Agregação**: indica “composição” de elementos.

Exemplos nos  
próximos *slides*

# Elo de Rastreabilidade SATISFAÇÃO

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - SIG que acorda em suas operacionalizações a necessidade de usar uma arquitetura cliente-servidor
  - Documento de Arquitetura de Software (DAS)
- Elo:
  - Satisfação: DAS **satisfaz** as operacionalizações desse SIG.

# Elo de Rastreabilidade RECURSO

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - Backlog do Produto
  - Backlog da Sprint 01
- Elo:
  - Recurso: Backlog da Sprint 01 depende de um **recurso** provido pelo Backlog do Produto, no caso, os requisitos priorizados para primeira Sprint.

# Elo de Rastreabilidade RESPONSABILIDADE

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - SIG que acorda em suas operacionalizações a necessidade de usar uma arquitetura cliente-servidor
  - Documento de Arquitetura de Software (DAS)
- Elo:
  - Responsabilidade: o Engenheiro de Requisitos é o **responsável** por elaborar o SIG, e o Arquiteto de Software é o **responsável** por elaborar o DAS.



# Elo de Rastreabilidade REPRESENTAÇÃO

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - Cenário RealizarLogin
  - Caso de Uso RealizarLogin
- Elo:
  - Representação: Caso de uso RealizarLogin **representa** o Cenário RealizarLogin.

# Elo de Rastreabilidade ALOCADO

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - Backlog do Produto, História “US\_01: Eu, como usuário do sistema, quero poder realizar *login* no MyApp usando o Facebook.”
  - Backlog da Sprint 01
- Elo:
  - Alocado: A História “US\_01: Eu, como usuário do sistema, quero poder realizar *login* no MyApp usando o Facebook.”, do Backlog do Produto do Projeto MyApp, está **alocada** no Backlog da Sprint 01.

# Elo de Rastreabilidade AGREGAÇÃO

Rastreabilidade entre Artefatos  
oriundos do **Processo de  
Desenvolvimento**

- Categoria: **Desenvolvimento**
- Elementos Rastreáveis:
  - Cenário Realizar Pagamento
  - Subcenário Realizar Pagamento à Vista
  - Subcenário Realizar Pagamento com Cartão de Crédito
- Elo:
  - Agregação: Tomando como base que os Cenários podem ser compostos de Subcenários, tem-se que o Cenário de Realizar Pagamento **é composto** pelos Subcenários de Realizar Pagamento à Vista e Realizar Pagamento com Cartão de Crédito.



# Matriz de Rastreabilidade – Exemplo II

Uma matriz de rastreabilidade pode ser tão complexa quanto se queira. Muitas informações podem ser registradas em uma matriz de rastreabilidade. Considerem os exemplos:

ID	System Requirement	Use Cases	Design Elements	Test Cases
RQ1	Ability to create a new book in catalog	UC1, UC4, UC5	DE3, DE6	TC1, TC6, TC9
RQ2	Ability to edit existing book in catalog	UC1, UC2	DE4	TC3, TC8
RQ3	Ability to create a new author in catalog	UC1, UC8, UC9	DE22	TC1, TC9

# Matriz de Rastreabilidade – Exemplo III

Uma matriz de rastreabilidade pode ser tão complexa quanto se queira. Muitas informações podem ser registradas em uma matriz de rastreabilidade. Considerem os exemplos:

ID	System Requirement	Source Documents	Stakeholders	Elicitation Activity
RQ1	Ability to create a new book in catalog	Project Goals List 1.22.2000	Chief Librarian	Goals Development Workshop
RQ2	Ability to edit existing book in catalog	None – implied	Librarian Joe Smith	Requirements analysis meeting 1.30.2000
RQ3	Ability to create a new author in catalog	Project Goals List 1.22.2000	Chief Librarian	Goals Development Workshop

# Matriz de Rastreabilidade – Outros

Outros exemplos:

<<consultar pacote .zip, disponível no mesmo tópico do Moodle que consta essa aula.>>



Exemplos  
apresentados  
em sala de aula

# Matriz de Rastreabilidade – Outros

Fonte (adaptado): Sayão e Leite.  
Rastreabilidade de Requisitos. ISSN 0103-9741

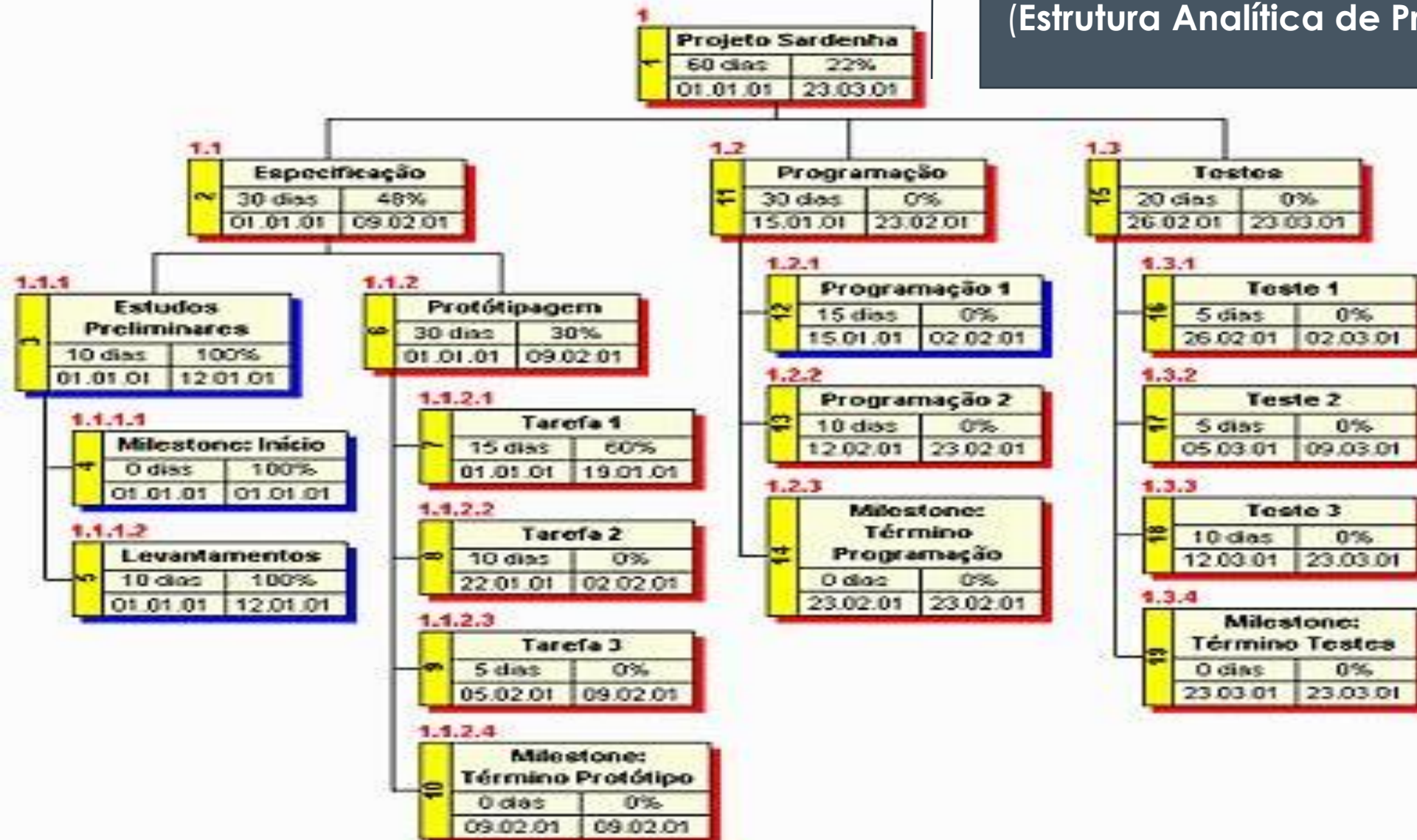
Em resumo, uma boa matriz de rastreabilidade deverá manter os elos das fontes de informações aos casos de teste...

Projeto <nome_projeto> - Matriz de Rastreabilidade				
Requisito	Documento fonte	Arquitetura	Componente	Caso de teste

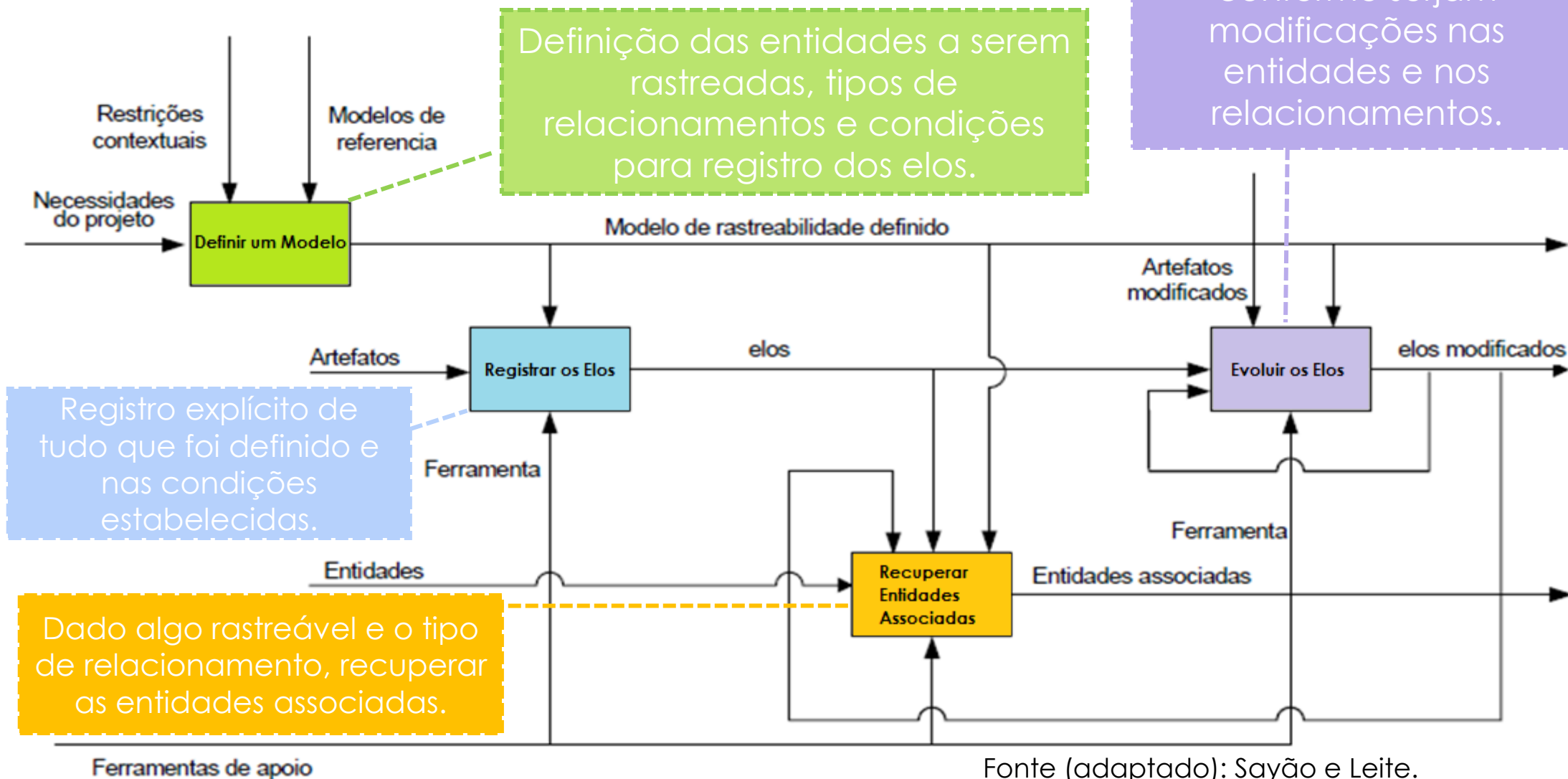


# EAP

Alguns materiais sobre rastreabilidade sugerem o uso de uma EAP (Estrutura Analítica de Projeto)



# Processo de Rastreabilidade



# Heurísticas Úteis

Definir os tipos de elos a serem registrados.



Identificar suportes para apoiar a rastreabilidade.



Conscientizar os envolvidos sobre a importância da rastreabilidade.



# Heurísticas Úteis

Estabelecer as entidades a serem rastreadas e os pontos de registro.



Verificar se os elos de rastreabilidade estão sendo registrados.



Analisar a efetividade do modelo de rastreabilidade, junto à equipe, após a liberação do software. Identificadas as necessidades, prover retroalimentação no modelo.



Em resumo...

## Em resumo...

*Um requisito é rastreável se é possível descobrir quem sugeriu o requisito (**a fonte**), por que o requisito existe (**rationale**), que outros requisitos estão relacionados a ele (**dependência entre requisitos**), e como o requisito se relaciona com outras informações, tais como desenho do sistema, implementação e documentação [Sommerville 98].*

Notem o quanto modelos como i\* e NFR atendem a essas expectativas! : )

# Considerações Finais

# Considerações Finais

- › Nessa aula, foi apresentada a atividade de pós-rastreabilidade de requisitos.
- › Nosso curso chega ao fim, em termo de conteúdo!
- › Agradecemos a todos pela ótimo semestre!
- › Continuem os estudos!





# Referências

# Referências

## Bibliografia Básica

---

1. [Ebrary] Young, Ralph. Requirements Engineering Handbook. Norwood, US: Artech House Books, 2003.

---

2. [Open Access] Leite, Julio Cesar Sampaio do Prado. Livro Vivo - Engenharia de Requisitos. <http://livrodeengenhariaderequisitos.blogspot.com.br/> (último acesso: 2017)

---

3. [Ebrary] Chemuturi, Murali. Mastering Software Quality Assurance : Best Practices, Tools and Technique for Software Developers. Ft. Lauderdale, US: J. Ross Publishing Inc., 2010.

---

4. Software & Systems Requirements Engineering: In Practice - Brian Berenbach, Daniel Paulish, Juergen Kazmeier, Arnold Rudorfer (Livro bem completo mas, não tem exemplar físico na biblioteca, nem mesmo consta na Ebrary)

---

5. Requirements Engineering and Management for Software Development Projects - Murali Chemuturi (Livro bem completo mas, não tem exemplar físico na biblioteca, nem mesmo consta na Ebrary)

---

# Referências

## Bibliografia Complementar

1. [BIBLIOTECA – 15 exemplares] Pfleeger, Shari Lawrence. Engenharia de Software: Teoria e Prática. 2ª. Edição. São Paulo: Prentice Hall, c2004. xix, 535 p. ISBN 978858791831
2. [BIBLIOTECA – 3 exemplares] Withall, Stephen. Software Requirement Patterns. Redmond: Microsoft Press, c2007. xvi, 366 p. ISBN 978735623989.
3. [BIBLIOTECA - vários exemplares] Leffingwell, 2011, Agile Software Requirements, <http://www.scaledagileframework.com/> (último acesso: 2017)
4. [Ebrary] Evans, Isabel. Achieving Software Quality Through Teamwork. Norwood, US: Artech House Books, 2004.
5. [Ebrary] Yu, Eric, Giorgini, Paolo, and Maiden, Neil, eds. Cooperative Information Systems: Social Modeling for Requirements Engineering. Cambridge, US: MIT Press, 2010.
6. [Open Access] Slides disponíveis em: <https://www.wou.edu/~eltonm/Marketing/PP%20Slides/> (último acesso: 2017)

Dúvidas?

Orientações?

Sugestões?

FIM

[mileneserrano@unb.br](mailto:mileneserrano@unb.br) ou [mileneserrano@gmail.com](mailto:mileneserrano@gmail.com)  
[serrano@unb.br](mailto:serrano@unb.br) ou [serr.mau@gmail.com](mailto:serr.mau@gmail.com)