

# INF016 – Arquitetura de Software

## 02 – Reorientação da Engenharia de Software

**Sandro Santos Andrade**  
sandroandrade@ifba.edu.br

**Instituto Federal de Educação, Ciência e Tecnologia da Bahia**  
**Departamento de Tecnologia Eletro-Eletrônica**  
**Graduação Tecnológica em Análise e Desenvolvimento de Sistemas**



# Arquiteturas em Contexto

- "o # o a arquitetura de *software* se relaciona # os # on # eitos de engenharia de *software* tradicionais # ente a # i # ados "
- (ais # on # eitos de) e # ser re\*orientados + &ois a arquitetura do *software* &assa a o # u # ar &a # e # funda # enta #
- " onhe # i # entos funda # entais,
  - (oda a # i # ação te # u # a arquitetura
  - (oda a # i # ação te # ao # enos u # arquiteto
  - Arquitetura **não é** u # a fase do desen) o # i # ento

# Arquiteturas em Contexto

- - uest . es %onsequentes,
  - / e onde surge a arquitetura de u # a a&\$i%ação '
  - " o # o u # a arquitetura de *software* &ode ser %ara%teri0ada '
- - uais são as suas &ro&riedades '
- 1 u # a arquitetura oa ou rui # '
- Suas defi%i2n%ias &ode # ser fa%i\$ # ente %orrigidas '
- 3s arquitetos estão se # &re %ons%ientes das de%is . es funda# entais de &ro4eto que to # a # '
- Essas de%is . es de &ro4eto &ode # ser arti%u\$adas %o # outras '

# Arquiteturas em Contexto

- - uest . es %onsequentes,
  - 3s arquitetos %onsegue # # anter a integridade %on%eitua\$ do &ro4eto ao \$ongo do te # &o '
  - A\$ternati)as fora # %onsideradas nos di)ersos # o # entos de de%isão '
- A arquitetura do *software* não 5 &roducto de u # a fase es&e%6fi%a do &ro%esso+ rea\$i0ada a&7s a an8\$ise de requisitos e antes do &ro4eto deta\$hado
- A %riação e # anutenção da arquitetura estão &resentes e # todo o &ro%esso+ e # ora tenha # destaque es&e%ia\$ e # u # a fase &arti%u\$ar

# Análise de Requisitos

- Considerações sobre a arquitetura no início do projeto
- Noções de estrutura de projeto e solução são consideradas durante a fase de análise de requisitos
- Visão tradicional, a análise e especificação de requisitos deve ser realizada de qualquer consideração sobre qual projeto irá satisfazer os requisitos
  - :A parte central deste artigo é sobre a abordagem para análise de requisitos que evita a atração negativa da orientação ; solução

# Análise de Requisitos

- Esta ideia não é nova,

*“Na análise nós começamos pelo que é requerido, o tomamos como certo e traçamos consequências até que alcancemos um ponto de onde podemos começar a síntese”*

Aa&us+ s5%u\$o B /!"!

- A síntese aqui entende-se o projeto ou a elaboração da solução

# Análise de Requisitos

- A ordem de George e a :How to solve it<sup>\*</sup> 1EFG,
  - Ari eiro o e reenda o e a
  - / eois entre u a oneHão entre os dados e o desonheido! E a agu # o ento )o%2 )ai encontrar u # &ano e ara a solução
  - EHeute o &ano
  - EHa#ine a solução o tida
- A# as a ordagens defende# a o # &eta eH&oração e o # e reensão dos requisitos antes de e o u a solução

# Análise de Requisitos

- É o uso desta abordagem, #8quinas de \$a)ar
  - #8quinas de \$a)ar não # rou&as e # ro%has ; eira de u# rio
  - 3 fo%o nos requisitos+ se # qualquer atração &\$a :orientação ; so\$uçã<+ &er#itiu a o tenção de so\$uç . es no)as e %riati)as, ta# ores rotati)os %o # agitadores
- Na &r8ti%a+ entretanto+ a an8\$ise de requisitos 5 rea\$i0ada de #odo r8&ido e su&erfi%ia\$,
  - Restriç . es de orça#ento e &ra0os
  - Aro%essos de desen)o\$)i#ento inferiores
  - Fa\$ta de %onfiança nos engenheiros res&ons8)eis

# Análise de Requisitos

- 3s #oti)os+ entretanto+ en)o\$)e# \$i#itaç . es hu#anas e# re\$ação a ra%ion6nio a strato+ e%ono#ia e e)o%ação
- Ana\$ogia %o# %onstrução de %asas,
  - Não ra%io%ina# os so re nossas ne%ecessidades inde&endente de %o# o e\$as serão satisfeitas
  - Aensa#os e# nJ#ero de %K#odos+ esti\$o das 4ane\$as+ fogão a g8s ou e\$5tri%o
  - Não &ensa#os e# ter#os de :u#a for#a de &ro)er a rigo a u# %\$i#a rigoroso< :u#a for#a de &ro)er i\$u#inação adequada< ou :u#a for#a de &re&arar %o#ida aque%ida<

# Análise de Requisitos

- 3 # es # o a % onte % e % o # *software*,
  - Se # refer2n%ia a arquiteturas 48 eHistentes torna\*se dif6%i\$ a) a\$iar a )ia i\$idade+ %ronogra#a e %usto do &ro4eto
  - " onhe%er as interfa%es de usu8rio+ *hardware* e ti&os de ser)iços dis&on6)eis a4uda a %hegar e # requisitos aseados nu # a %o # &reensão ra0o8)e\$ da )ia i\$idade
  - As **falhas** i # &u\$iona # a engenharia e são a ase &ara ino)ação, o ser)ação L dete%ção das \$i # itaç . es
  - EHe # &\$o, %riação do *zipper* – su%essor de u # a \$onga sequ2n%ia de in)enç . es
    - : " o # o # uitos outros &rodutos+ o 0i&&er não surgiu direta#ente das fun%ionaa\$idades # as de %orreç . es su%essi)as de fa\$has< -Aetros?i 1EE2@

# Análise de Requisitos

- 3 ser) aç . es funda # entais,
  - Aro4etos e arquiteturas 48 eHistentes define # u # )o%a u\$8rio &ara dis%utir as &ossi i\$idades
  - Nossa %o # &reensão so re o que fun%iona ho4e e %o # o e\$e fun%iona afeta nossos dese4os – fo%o na so\$ução
  - EH&eri2n%ias &r5)ias %o # siste # as nos a4uda a a)a\$iar a )ia i\$idade e definir %ustos e &ra0os
  - Requisitos M arti%u\$ação de # e\$horias ne%ess8rias ; arquitetura )igente
  - Isso não signifi%a \$i # itar ino)ação+ as # 8quinas de \$a)ar fora # &rogressi)a # ente a&erfeioadas

# Análise de Requisitos

- 3 ser) aç . es funda # entais,
  - Não se \$i # itar+ entretanto+ aos &ro4etos atuais! / iferentes # e%anis # os de)e # ser usados &ara su ir e # u # a %asa+ arranha\*%5u ou at5 a \$ua
  - - uando não eHiste # ante%essores f6si%os ana\$ogias ou ante%essores %on%eituais &ode # a4udar
  - / esen)o\$)i # ento *greenfield* ta # 5 # uti\$i0a ante%essores &ara enquadrar os requisitos e so\$uç . es in5ditos
  - Ne # todas as arquiteturas+ entretanto+ são oas fontes de ins&iração

# Projeto

- Fase onde se dá maior atenção à definição das interfaces de comunicação com o sistema de arquitetura
- O desenho da arquitetura, entretanto, não é o uso desta fase
- A arquitetura é um **aspecto** de todas as outras atividades de desenho
- As interfaces arquiteturais refletem os aspectos do sistema, requerendo um relacionamento de transações com o sistema

# Projeto

- I o de \$o tradi%iona\$ Ne # %as%ata0,
  - 3 4eti)o, definir u# &ro4eto que &ossa ser re&assado &ara os &rogra#adores
  - Se a\$gu# requisito 5 %onsiderado in)i8)e\$+ retorna\*se ; fase de an8\$ise de requisitos Nse# entretanto reto# ar as&e%tos da so\$uçã0
  - Se+ na i# &\$e# entaçã0+ a\$gu# a &arte do &ro4eto 5 %onsiderada in)i8)e\$+ retorna\*se ; fase de &ro4eto

# Projeto

- Iode o %entrado e # arquiteturas,
  - Redu 0\*se ou e \$i # ina\*se as fronteiras entre as fases+ gera \$ # ente artifi %iais e i # &ro duti) as
  - An 8 \$ise de requisitos 48 re \$a %ionada a as &e %tos de arquitetura e &ro 4eto
  - An 8 \$ise+ &ro 4eto e i # & \$e # entação a %onte %e # de u # a for # a # ais integrada e enrique %ida
- 3 arquitetura \$ida %o # u # a a # & \$a fai Ha de quest . es,
  - Interesses dos *stakeholders*+ esti \$o e estrutura uti \$i 0ados+ ti & os de %one %tores de e \$e # entos+ estrutura &ri # 8rias de % \$asses e &a %otes+ as &e %tos de distri uição+ des %entra \$i 0ação+ i # & \$antação e segurança+ et %

# Projeto

- (5%ni%as de &ro4eto de siste # as,
  - Aro4eto 3rientado a 3 4etos,
    - Identifi%ação de o 4etos que en%a&su\$a# u# estado e funç . es que a%essa# e # ani&u\$a# este estado
    - Não 5 u#a a ordage # %o # &\$eta ne # 5 efi%a0 e# todas as situaç . es! Não 5 inefi%a0+ entretanto
    - Pi # itaç . es,
      - Não 5 u#a a ordage # %o # &\$eta de &ro4eto! Não a orda quest . es de i # &\$antação+ segurança+ %onfiança+ et%
      - Não &ossui # e%anis # os &ara transferir+ &ara no)as arquiteturas+ %onhe%i # ento de do # 6nio e so\$uç . es &resentes e# arquiteturas anteriores
      - EHige que todos os %on%eitos e entidades se4a# o 4etos! Não h8 su&orte eH&\$6%ito &ara a\$go que não se4a u#a %\$asse

# Projeto

- (5%ni%as de &ro4eto de siste # as,
  - Aro4eto 3rientado a 3 4etos,
    - Pi # itaç . es,
      - / is&oni i \$i0a so # ente u # ti&o de en%a&su\$a # ento No 4eto0+ u # a noção de interfa%e+ u # Jni%o ti&o de %one%tor eH&\$6%ito Nprocedure call0! Não su&orta a noção de *required interfaces*
      - Forte # ente \$igada a interesses e de%is . es das \$inguagens de &rogra # ação+ que &ode # %o # eçar a ditar quais de%is . es são i # &ortantes
      - Assu # e u # es&aço de endereça # ento %o # &arti\$hado e su&orte adequado ao geren%ia # ento de *heap* e *stack*
      - Assu # e a eHist2n%ia de u # a Jni%a *thread* de %ontro\$e
      - As&e%tos de %on%orr2n%ia+ distri uição e des%entra\$i0ação não são %onsiderados
    - A Q I P a4udou a dis%utir u # &ro4eto orientado a o 4etos se # de&ender da \$inguage # de &rogra # ação

# Projeto

- (5%ni%as de &ro4eto de siste # as,
  - *Domain-Specific Software Architectures N / SSAs0*
    - A&ro&riada quando eH&eri2n%ias e arquiteturas anteriores inf\$uen%ia# &oten%ia\$# ente os no)os &ro4etos
    - Cera\$# ente a eH&eri2n%ia tra0 a #e\$hor a ordage# e #e\$hor so\$uçãõ &ara o do#6nio e# questãõ
    - No)as arquiteturas serãõ )ariaç . es das anteriores
    - A re\*se es&aço &ara fo%ar e# )ariaç . es originais e %riati)as
    - Reuti\$i0a\*se &artes da arquitetura e da i# &\$e# entaçãõ
    - EHige o# su&orte t5%ni%o, arquiteturas anteriores de)e# ser %a&turadas e refinadas &ara reuso+ &ontos de )ariaçãõ de)e# ser identi%ados e iso\$ados+ interfa%es nos &ontos de )ariaçãõ de)e# ser eH&\$6%itas+ et%

# Implementação

- 3 4eti)o, %riar u# %7digo\*fonte que se4a fie\$ ;  
arquitetura e que i# &\$e# ente de for# a %o# &\$eta os  
requisitos
- A a ordage# %entrada e# arquiteturas d8 2nfase a  
a\$gu# as a ordagens ; i# &\$e# entação,
  - A i# &\$e# entação &ode estender ou # odifi%ar a arquitetura
  - A arquitetura s7 estar8 %o# &\$eta a&7s a i# &\$e# entação
  - / e)e\*se # anter as de%is . es que %onstitue# a arquitetura  
%onsistentes %o# o %7digo\*fonte &rodu0ido
  - Esti# u\$a a uti\$i0ação de t5%ni%as generati)as e forte# ente  
aseadas e# reuti\$i0ação NeH, uso de *frameworks*0

# Implementação

- Implementação final,
  - (todos os elementos estruturais da arquitetura estão implementados no código-fonte)
  - 3 código-fonte não de) e utilizar elementos opcionais que não estão presentes na arquitetura
  - 3 código-fonte não de) e manter apenas os elementos da arquitetura que não estão desativados na arquitetura

# Implementação

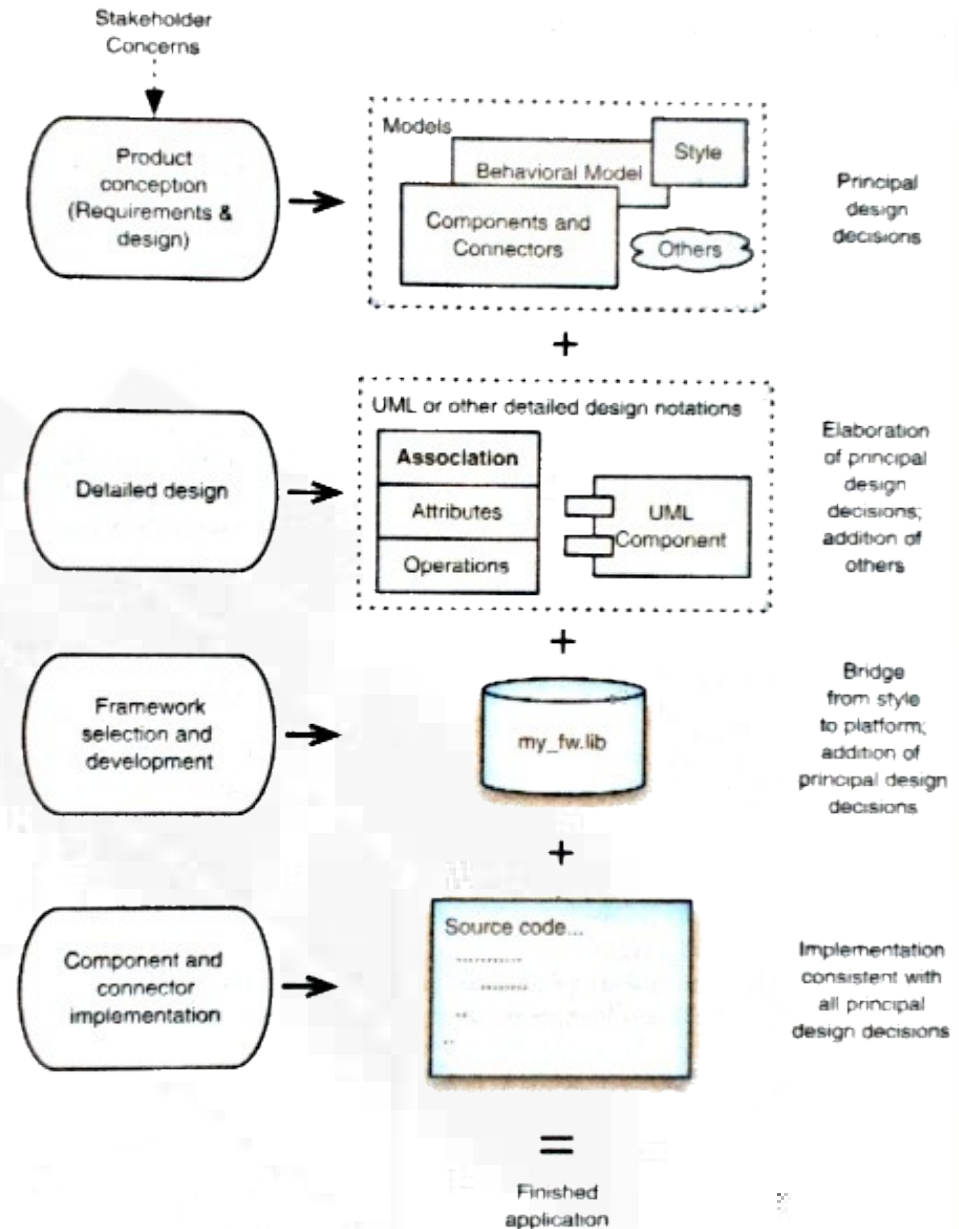
- Na &r8ti%a essa definição não 5 tota\$ # ente adequada,
  - Qso de i \$iote%as redu0e # o %usto e au # enta # a qua\$idade+ &or5 # %ont2 # func. es e interfa%es que não estão &resentes na arquitetura
  - Se u # a i \$iote%a arata e de qua\$idade i # &\$e # enta ER S da fun%iona\$idade dese4ada e se as %onsequ2n%ias da aus2n%ia dos outros 2 S fore # a%eit8)eis,
    - / e%ide\*se &e\$ uso da i \$iote%a
    - Rea\$i0a\*se a re)isão das es&e%ifi%aç. es e da arquitetura
  - 3 &onto %r6ti%o 5 se # &re # anter a arquitetura e a i # &\$e # entação e # estados %onsistentes

# Implementação

- Estratégias de implementação e prioridade,
  - 10 (5%ni%as generati)as, a implementação 5 gerada automaticamente e possui alta qualidade
    - Certeza e confiabilidade do sistema e seus efeitos
  - 20 (5%ni%as baseadas e reuso, de banda e menor tempo e introdução de maior qualidade do que construir o sistema do zero
    - *Architecture Implementation Framework* Monte entre um estio arquitetura e um conjunto de tecnologias de implementação! (ra0 garantias
    - Soluções de *middleware*+ *COTS*+ *open-source*
  - T0 / (eseno)imento anua%o#&eto, custos e riscos maiores! Maior necessidade de garantia da qualidade

# Implementação

- Architecture Implementation Framework,



# Implementação

- Se a implementação difere da arquitetura projetada, esta arquitetura não caracteriza a ação
- 3 sistemas ou a arquitetura, o sistema e o contraste; que do sistema
- Fazhas e reconhecer esta diferença,
  - Rou a a ha idade de raciocinar, no futuro, so re a arquitetura i & e entada da ação
  - Engana os *stakeholders* e reação ao que eles :a%redita# que t2#< e o que eles :rea\$# ente t2#<
  - (orna qualquer estratégia de desen) i# ento ou e) o\$ução+ aseada na arquitetura+ i# &re%isa e fadada ao fra%asso

# Análise e Teste

- Atividades realizadas para garantir a qualidade de um artefato
- Na abordagem tradicional o código-fonte é compilado e executado para verificar a correção funcional e eventualmente detectar erros
- Entretanto, a análise de um programa determina a robustez e pode ser realizada assim que o artefato existir, seja ele o que for
- Porque só o código-fonte é testado
- Porque o teste é realizado somente em relação aos requisitos funcionais da aplicação

# Análise e Teste

- Resposta, devido ; ausência de qualquer representação suficiente e rigorosa da ação que não seja o código-fonte
- Arquiteturas e iterativa a análise antecedente e a hora do código-fonte
- A resequência para a análise de propriedades não-funcionais
- - quais os benefícios que as arquiteturas de *software* trazem ; fase de análise e teste

# Análise e Teste

10 3 #ode\$o arquitetura\$ &ode ser a) a\$iado e # re\$ação ; sua %onsist2n%ia interna e %orretude,

- 9erifi%aç . es sint8ti%as do #ode\$ &ode # identifi%ar+ &or eHe # &\$o+ %oneH . es entre %o # &onentes não %o # &at6) eis *Ninterface mismatch*0+ es&e%ifi%ação in%o # &\$eta de &ro&riedades e &adr . es de %o # uni%ação indese4ados
- An8\$ise de f\$uHo de dados &ode ser a&\$i%ada &ara deter #inar in%o # &ati i\$idades de definiçãoUuso e &ara dete%tar fa\$has de segurança
- (5%ni%as de *model-checking* &ode # ana\$isar &ro \$e # as de *deadlock*
- (5%ni%as de si # u\$ação &ode # rea\$i0ar for # as si # &\$es de an8\$ise dinV # i%a

# Análise e Teste

- 20 3 #ode\$o arquitetura\$ &ode ser a)a\$iado e# re\$ação  
; sua %onsist2n%ia %o# os requisitos,
- Inde&endente do &ro%esso uti\$i0ado o #ode\$o arquitetura\$ de)e ser %onsistente %o# os requisitos
  - Esta )erifi%ação ta\$)e0 &re%ise ser feita de for#a # anua\$+ %aso os requisitos este4a# des%ritos e# \$anguage# natura\$
  - A )erifi%ação 5 essen%ia\$

# Análise e Teste

**T0** 3 #ode\$o arquitetura\$ &ode ser uti\$i0ado &ara deter#inar e su&ortar estrat5gias de an8\$ise e teste a&\$i%adas ao %7digo\*fonte,

- A arquitetura &ro)2 o &ro4eto do %7digo\*fonte+ %onsist2n%ia entre e\$es 5 essen%ia\$
- A arquitetura ser)e %o#o u#a fonte de infor#ação &ara go)ernar testes+ aseados na es&e%ifi%ação+ que atua# e# todos os n6)eis, unidade+ su \*siste#a e siste#a
  - 3 arquiteto &ode &riori0ar an8\$ises e testes %o# ase na arquitetura+ fo%ando os %o#&onentes e #ontagens #ais %r6ti%os
  - EH, %o#&onentes %o#uns e# todos os #e# ros de u#a fa#6\$ia de &rodutos

# Análise e Teste

**T0 3** #ode\$o arquitetura\$ &ode ser uti\$i0ado &ara deter#inar e su&ortar estrat5gias de an8\$ise e teste a&\$i%adas ao %7digo\*fonte,

- A arquitetura ser)e %o#o u#a fonte de infor#ação &ara go)ernar testes+ aseados na es&e%ifi%ação+ que atua# e# todos os n6)eis, unidade+ su \*siste#a e siste#a
  - A arquitetura dis&oni i\$i0a u# #eio &ara re&assar+ &ara no)os &ro4etos+ resu\$tados &r5)ios de an8\$ise
  - EH, a eHtensão do teste de unidade de u# %o#&onente 5 redu0ido se e\$e est8 sendo a&\$i%ado e# u# #es#o %onteHto e %ondiç.es de uso
  - A arquitetura a4uda a des)iar a atenção do ana\$ista &ara os %one%tores e# u#a i#&\$e#entação do siste#a

# Análise e Teste

**B0** 3 #ode\$o arquitetura\$ &ode ser %o#&arado %o# u#  
#ode\$o deri)ado a &artir do %7digo\*fonte da  
a&\$i%ação,

- 1 u# a for # a de %he%ar a sua so\$ução
- Se4a  $P$  u# &rogra# a deri)ado da arquitetura  $A$
- $Q$  # gru&o diferente de engenheiros+ se # a%esso a  $A$ +  
desen)o\$)e u# #ode\$o arquitetura\$  $A'$  a &artir da an8\$ise de  
 $P$
- Se tudo esti)er %orreto  $A$  ser8 %onsistente %o#  $A'$
- " aso %ontr8rio+ ou  $P$  não i# &\$e #enta  $A$  fie\$ #ente ou  $A'$  não  
ref\$ete fie\$ #ente a arquitetura de  $P$
- E# qua\$quer %aso 5 ne%ess8rio u# a )erifi%ação

# Evolução e Manutenção

- Evolução ou Manutenção de *Software* refere-se a todo tipo de atividade relacionada ao lançamento e ao desenvolvimento da aplicação
- A abordagem tradicional para a evolução é *ad-hoc* e gera constantemente problemas; fase do processo relacionada; mudança
- 3 riscos à degradação da qualidade da aplicação,
  - / e)ido a mudanças realizadas e qualquer alteração ou qualquer meio que se faça o sistema e o sistema
  - "o# o te#&o+ realizar mudanças sucessivas se torna entre #a#ente difícil )isto que de #end2n%ias %o# &\$eHas entre mudanças independentes anteriores )2# ; tona

# Evolução e Manutenção

- A a ordage # %entrada e # arquitetura ofere%e u # a ase s7\$ida &ara u # a e) o\$ução efi%a0,
  - Fo%o sustentado e # u # # ode\$o arquitetura\$ eH&\$6%ito+ rea\$ e # odifi%8)e\$
- Fases do &ro%esso de e) o\$ução,
  - 10 I oti)ação
  - 20 A) a\$iação
  - T0 Es%o\$ha e &ro4eto da a ordage #
  - B0 EHe%ução+ in%\$uindo a &re&aração &ara a &r7Hi # a rodada de ada&tação

# Evolução e Manutenção

## 10) Iotação,

- / entre as di)ersas #oti)aç .es &ara e)o\$ução+ desta%a\*se a %riação de no)as )ers .es de u# &roduto
- >ustifi%a o estudo de fa#6\$ias de &roduto

## 20) A)iação,

- A #udança 5 eHa#inada &ara deter#inar sua )ia i\$idade! "aso se4a )i8)e\$+ %o#o e\$a ser8 a\$%ançada '
- Requer %onhe%i#ento a&rofundado so re o &roduto e# questão
- Se u# #ode\$o arquitetura\$ fie\$ ; i#&\$e# entação est8 dis&on6)e\$ a %o#&reensão e an8\$ise da #udança o%orre# de for# a efi%a0

# Evolução e Manutenção

## 20 A) a\$iação,

- Se não existe #ode\$o arquitetura\$ ou o #ode\$o não 5 %onsistente %o# a i#&\$e# entação &ode\*se uti\$i0ar engenharia re)ersa! Isso %usta te #&o e dinheiro
- Erros de #anutenção surge# de inde)idas a)a\$iaç . es,
  - Se não h8 %onhe%i# ento sufi%iente so re a estrutura eHistente os &\$anos de #odifi%ação irão fa\$har+ &rin%i&a\$# ente nos &ontos des%onhe%idos
- Q# o# #ode\$o arquitetura\$ ofere%e u#a ase 4ustifi%ada &ara de%idir se u#a #udança dese4ada 5 ra0o8)e\$ ou não
  - Aode\*se )erifi%ar que a #udança 5 eHtre #a# ente %ustosa ou que in)ia i\$i0a &ro&riedades do siste#a
  - I ant2 #\*se a integridade arquitetura\$ e atenua a )o\$ati\$idade dos requisitos

# Evolução e Manutenção

**T0** Especificação e Projeto da arquitetura,

- A arquitetura deve satisfazer todos os requisitos que motivaram a mudança
- 1 realizada durante a especificação e projeto

**B0** Evolução,

- 3 requisitos artefatos a serem modificados o código da arquitetura
- 7 então mudanças no código-fonte são realizadas
- A consistência deve ser sempre mantida! Ferramentas ajudam nesta tarefa
- A manutenção não está concluída até que os códigos estejam consistentes

# Processos de Desenvolvimento

- A arquitetura é a base para todas as atividades de desenvolvimento e manutenção do *software*
- O conjunto de decisões arquiteturais é criado e refinado continuamente com os requisitos e continua sendo atualizado com a manutenção
- A natureza da arquitetura é diferente da encontrada nas características tradicionais das atividades de desenvolvimento de *software*,
  - As atividades do processo são o ponto de partida e não o ponto de chegada da arquitetura e nenhum lugar
  - Existem limites rígidos entre os tipos de atividades

# Processos de Desenvolvimento

- Afirma-se que o processo de desenvolvimento de software é uma atividade que não quer dizer que existe uma única forma correta de realizar o desenvolvimento,
  - Estratégias alternativas podem ser utilizadas dos pontos fortes e referências da organização
  - Diferentes atividades de modificação irão demandar maior atenção a certas atividades
- Quando for analisado o comportamento e o desempenho de diferentes estratégias de desenvolvimento de software, as tarefas do desenvolvimento das arquiteturas

# Processos de Desenvolvimento

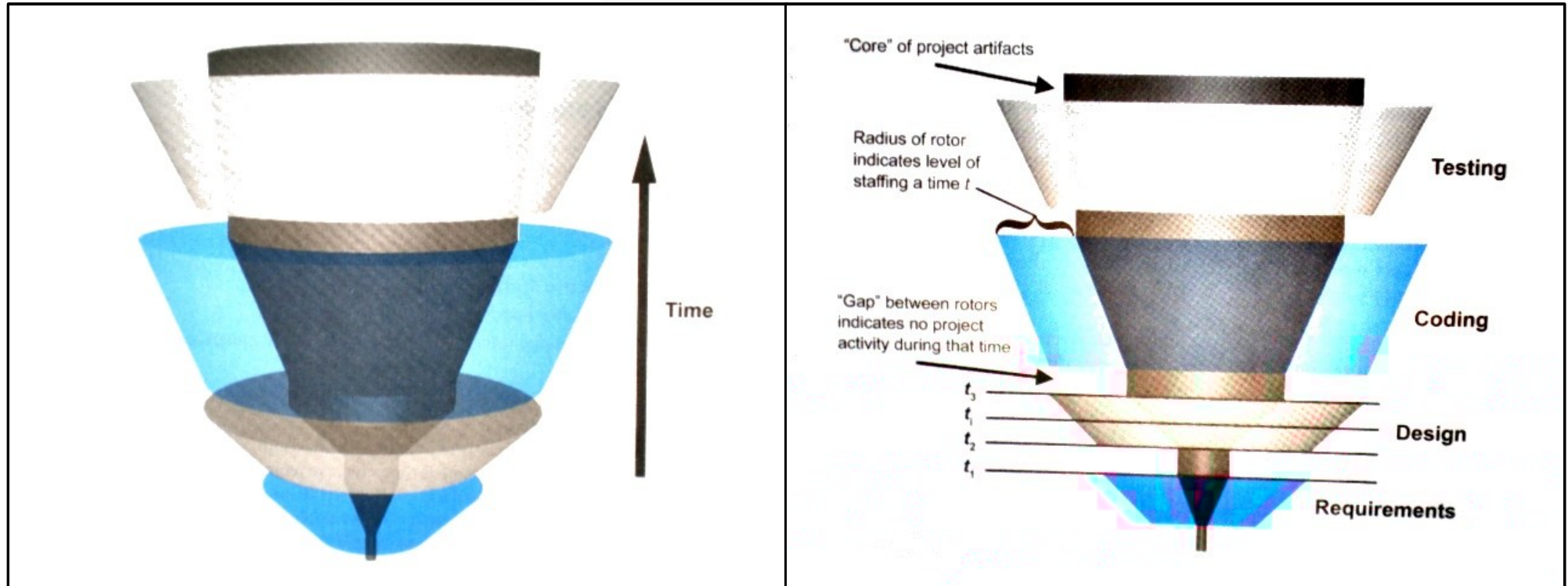
- 9 isua\$ i0 açãõ e # (ur ina,
  - I 5 todo & ara i\$ ustrar u# % on4 unto integrado de ati) idades de desen) o\$) i # ento no qua\$ 5 e) iden% iado o & a& e\$ % entra\$ da arquitetura do *software*
- Pe) a e # % onsideração diferentes as& e% tos do desen) o\$) i # ento,
  - (e # & o
  - (i& os de ati) idades ati) as e # u# deter # inado te # & o
  - Esforço NeH, horas tra a\$ hadas0 e # u# deter # inado te # & o
  - Estado do & roduto NeH, % onteJ do gera\$ do & ro4eto ou % onhe% i # ento a% u # u\$ ado0

# Processos de Desenvolvimento

- 9 isua\$ioação e# (ur ina,
  - 1 definida es&a%ia\$# ente &or u# %on4unto de an5is %o# \$argura e es&essura )ari8)eis+ dis&ostos ao \$ongo de u# nJ%\$eo
  - 3 eiHo do nJ%\$eo 5 o te# &o
  - 3 nJ%\$eo re&resenta o &roduto sendo desen)o\$)ido
  - 3 s an5is re&resenta# fases de\$i# itadas &e\$ o te# &o
  - A es&essura do ane\$ denota a duração das N&ossi)e\$# ente %on%orrentes0 ati)idades do ane\$
  - 3 )o\$u# e do ane\$ re&resenta o in)esti# ento rea\$i0ado
  - Se%ç . es trans)ersais de u# #es# o ane\$ &ode# ser diferentes e inter)a\$os entre an5is &ode# eHistir

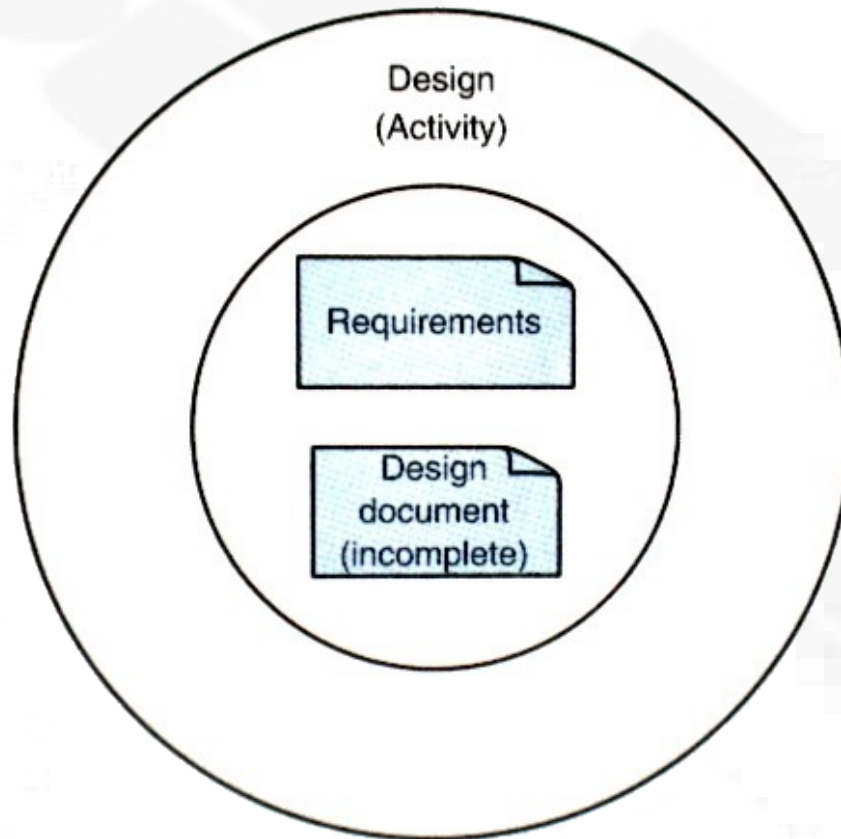
# Processos de Desenvolvimento

- 9isua\$ioação e# (ur ina N#ode\$o e# %as%ata0,



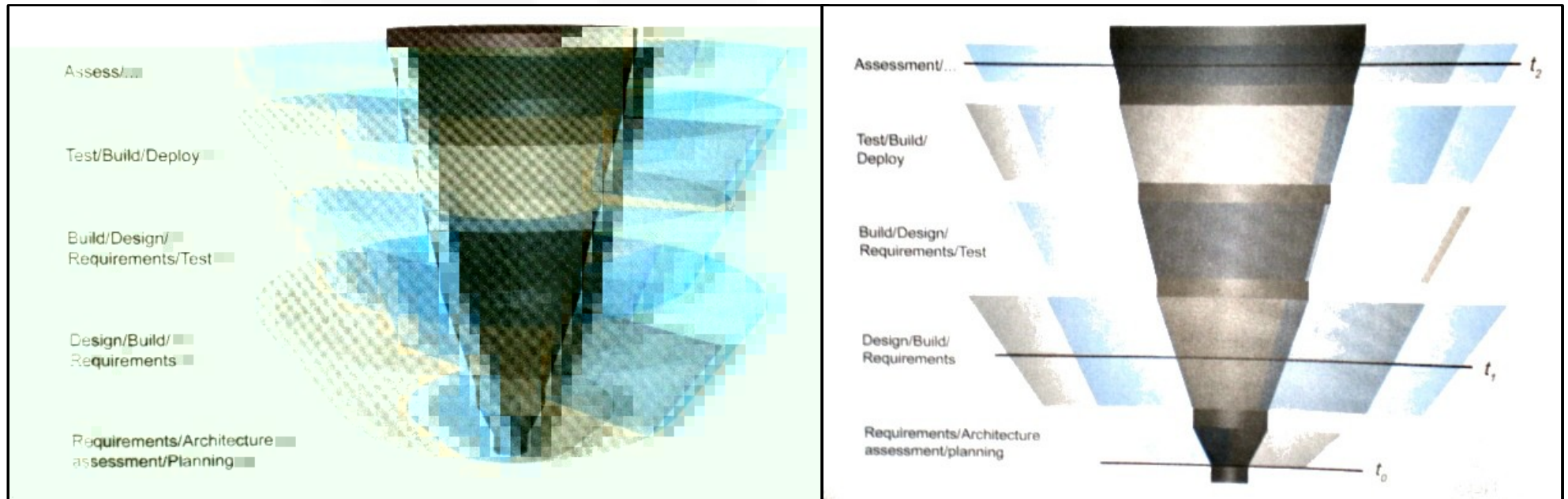
# Processos de Desenvolvimento

- 9isua\$ioação e# (ur ina N#ode\$o e# %as%ata0,
  - Se%ção trans)ersa\$ no te#&o  $t_1$



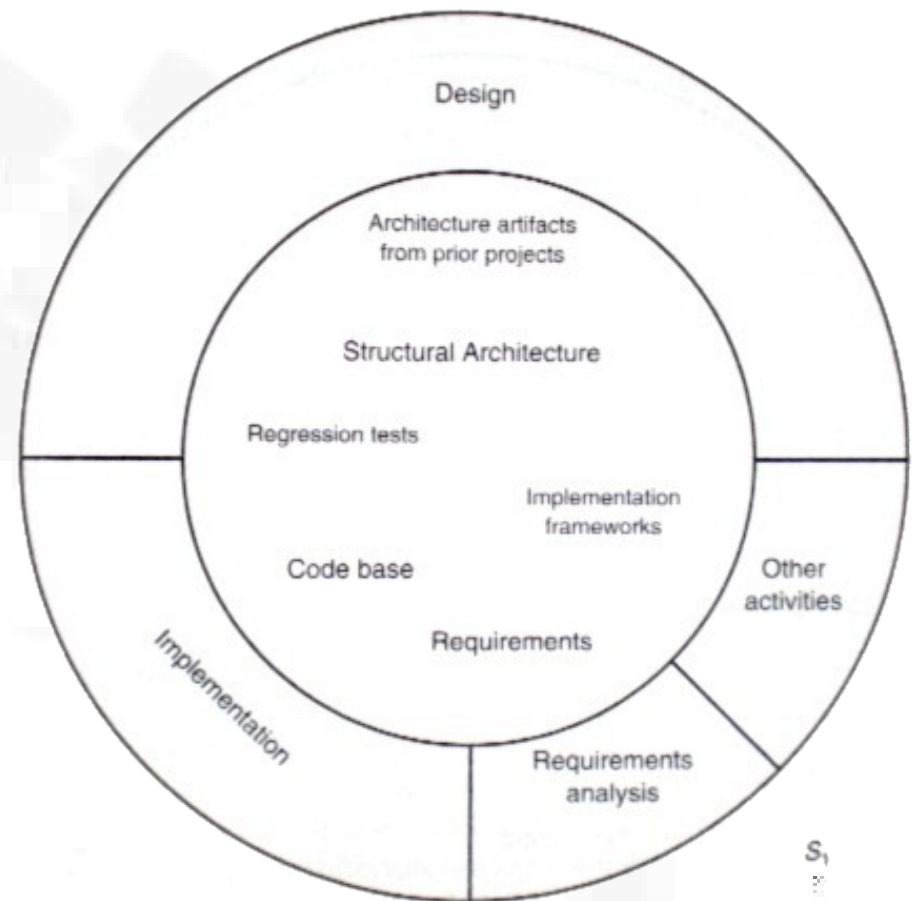
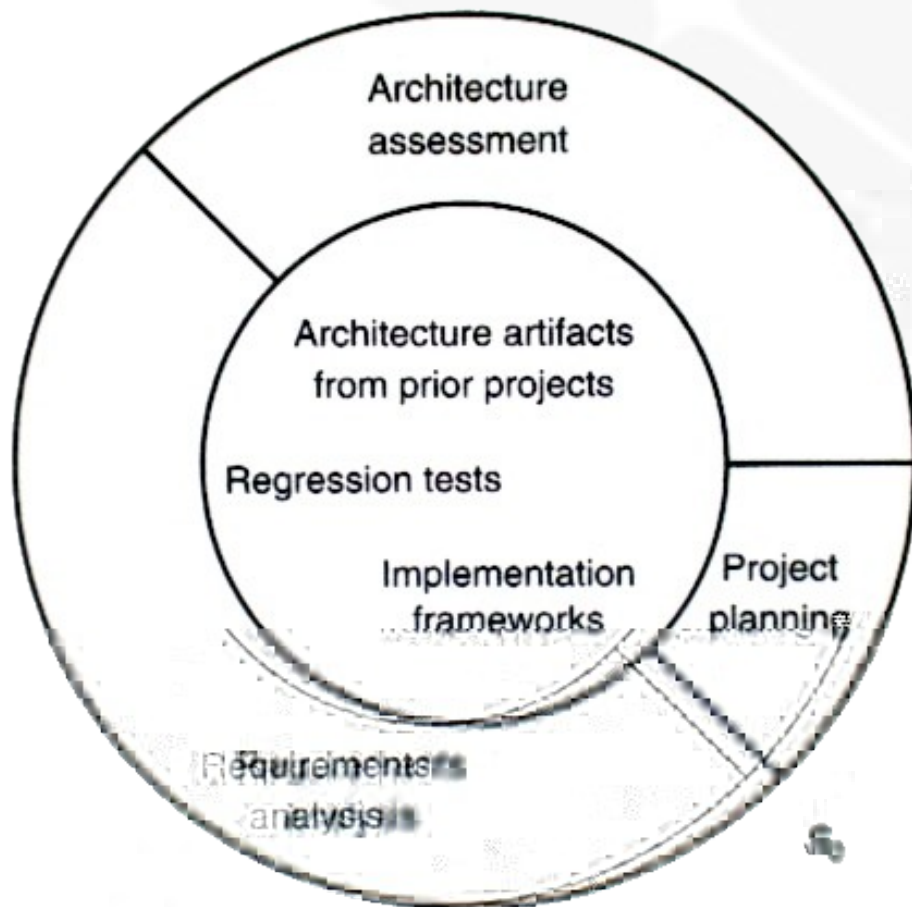
# Processos de Desenvolvimento

- 9isua\$ioação e# (ur ina Nati)idades %on%orrentes0,



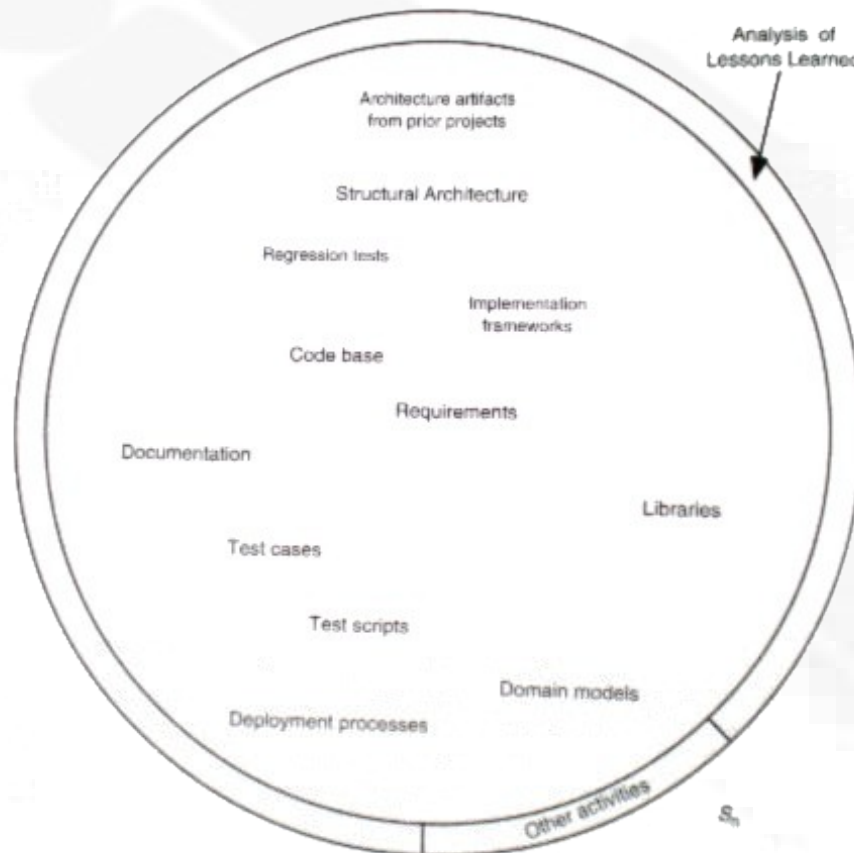
# Processos de Desenvolvimento

- 9 isua\$ioação e# (ur ina Nati)idades %on%orrentes0,
  - Se%ç . es trans)ersais nos te#&os  $t_0$  e  $t_1$



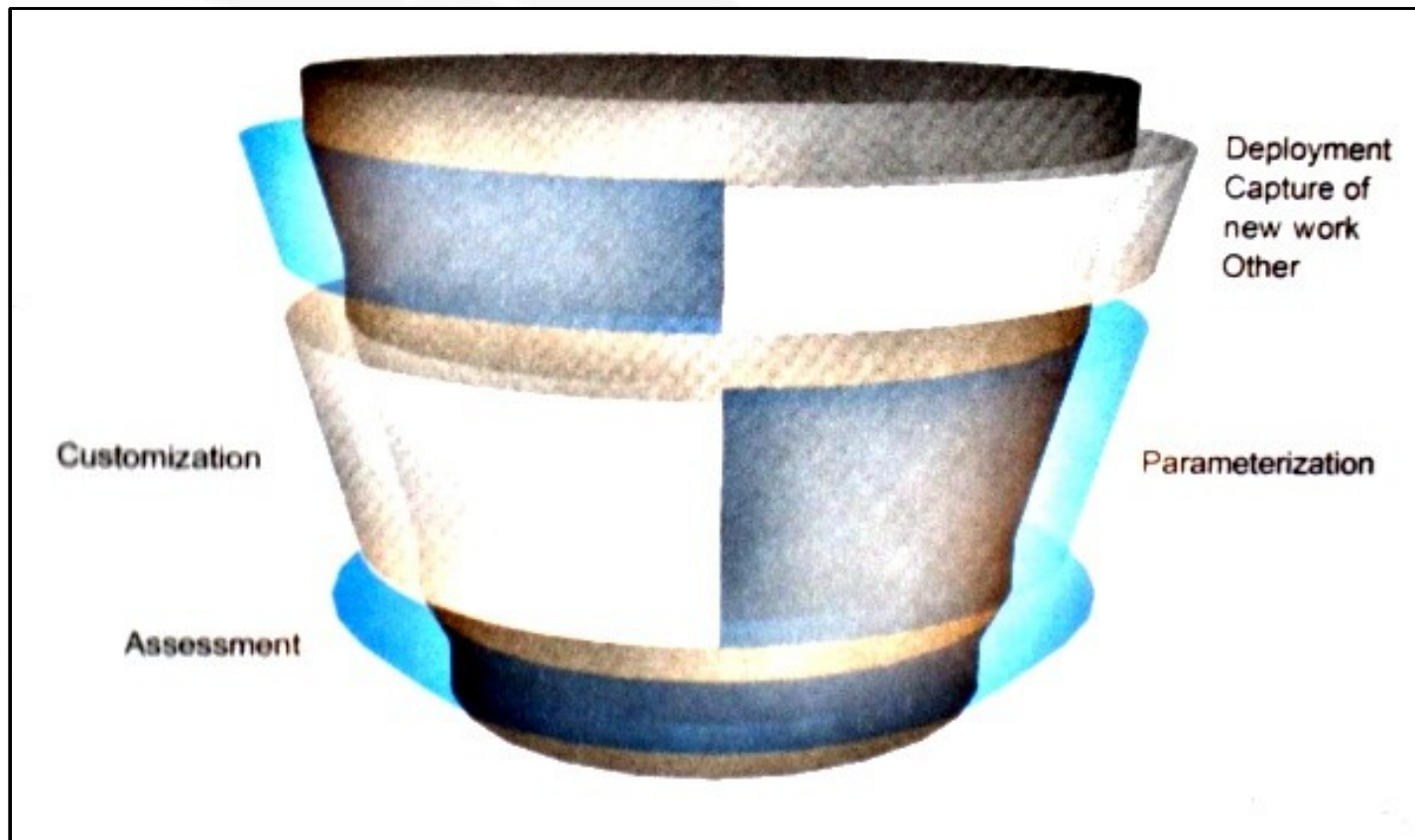
# Processos de Desenvolvimento

- 9 isua\$ i0 ação e # (ur ina Nati) idades % on % orrentes 0,
  - Se % ção trans) ersa \$ & r7 Hi # o ao fi # do & ro4 eto



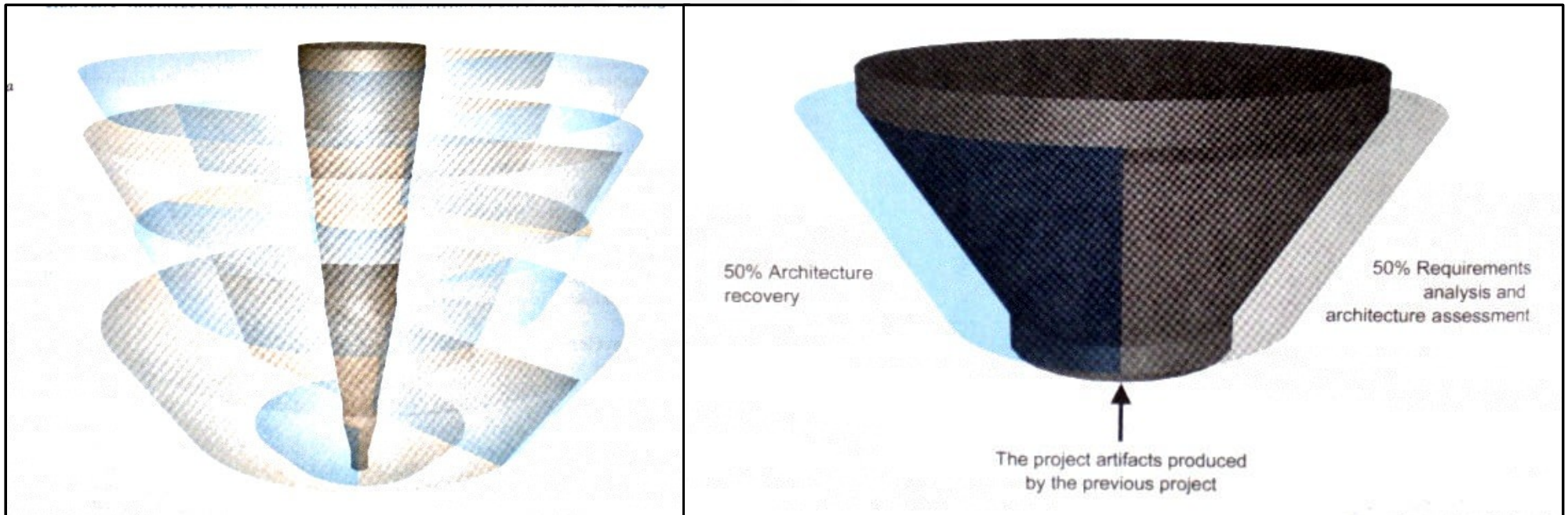
# Processos de Desenvolvimento

- EHe # &\$o de des%rição de &ro%esso,
  - *Domain-Specific Software Architecture* N / SSA0



# Processos de Desenvolvimento

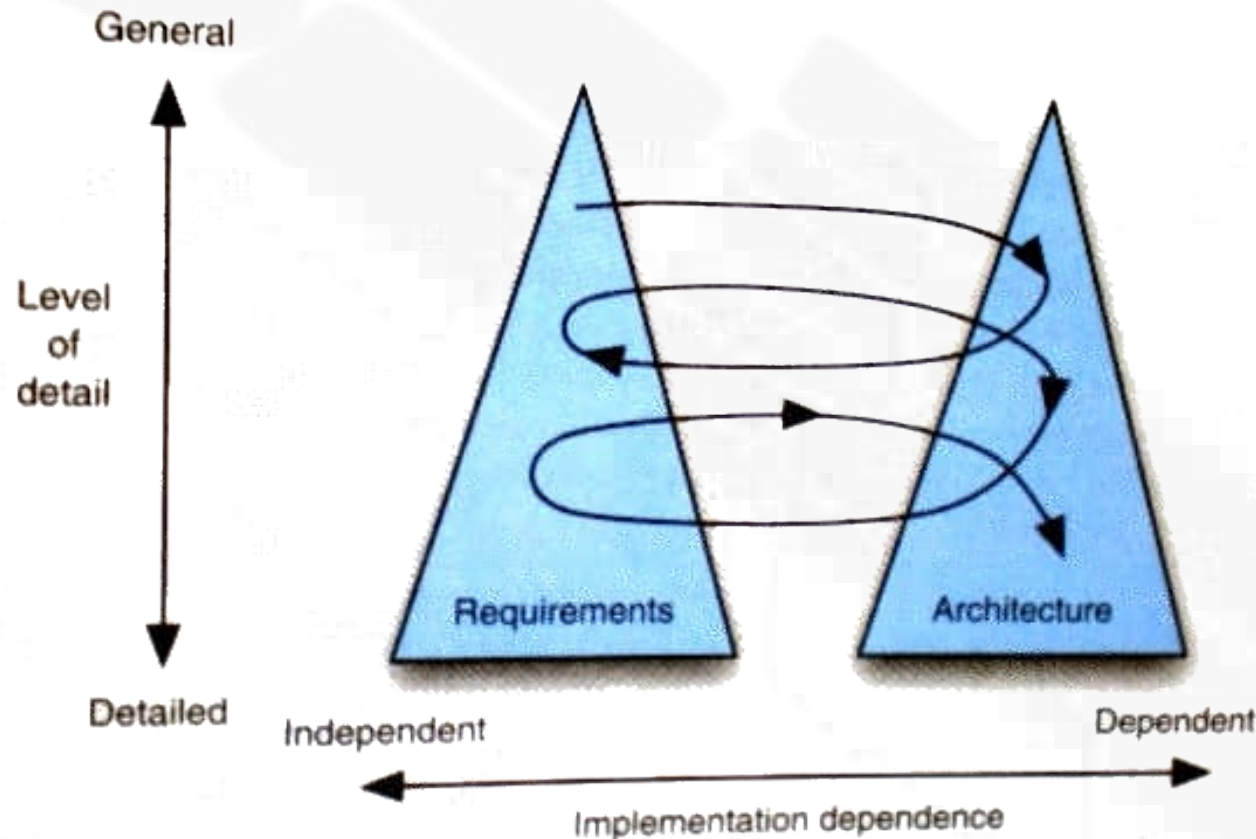
- EHe # &\$o de des%rição de &ro%esso,
  - / esen)o\$)i# ento 8gi\$



# Processos de Desenvolvimento

- Outros modelos de processo,

- *Twin Peaks* [Nusei et al 2001]



# INF016 – Arquitetura de Software

## 02 – Reorientação da Engenharia de Software

**Sandro Santos Andrade**  
sandroandrade@ifba.edu.br

**Instituto Federal de Educação, Ciência e Tecnologia da Bahia**  
**Departamento de Tecnologia Eletro-Eletrônica**  
**Graduação Tecnológica em Análise e Desenvolvimento de Sistemas**

