



# Aspectos Relevantes da Qualidade do **Software Estrutural** e Sua Influência na Elaboração de Projetos

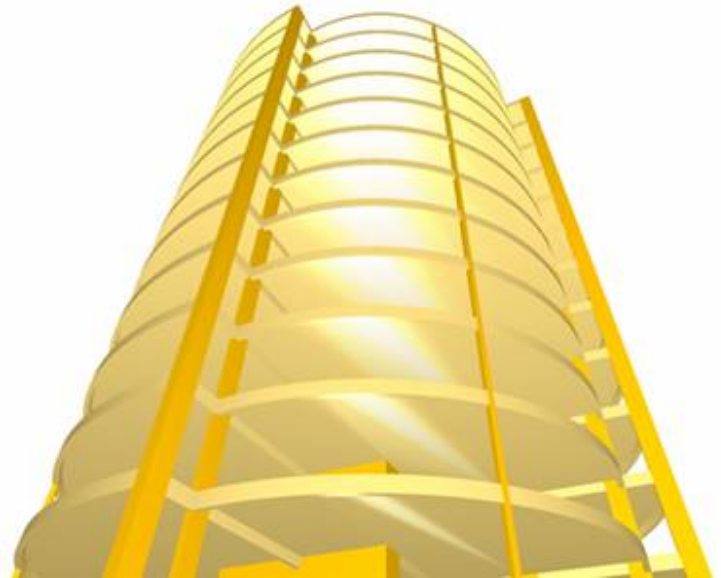
1. Introdução
2. Relação Software / Projeto Estrutural
3. Abrangência dos Softwares (Individuais e Integrados)
4. Dificuldades na Produção de Software – Porquê a maioria “morre”?
5. Dificuldades na Aplicação (Software Integrado)
6. Características de Qualidade de um Software Integrado (Usuários)
7. Metodologia para Garantia de Qualidade dos Softwares (Empresas)
8. Providências para Melhorar a Qualidade do Software
9. Recomendações - Melhoria da Qualidade do Projeto Estrutural

# 1 | Introdução

# Introdução

## Hipótese Básica

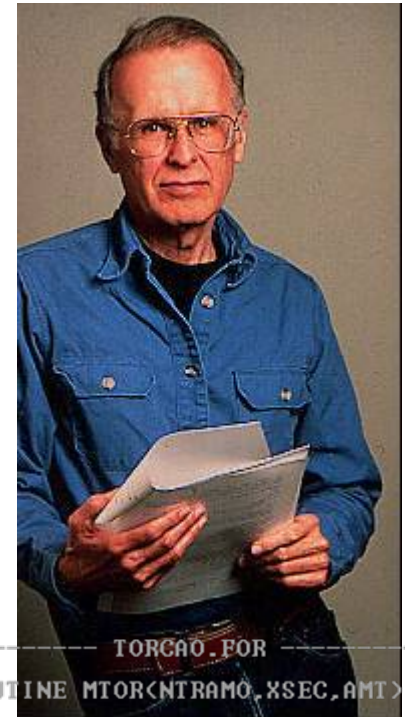
- Software para elaboração de projeto de estruturas de concreto armado e protendido
  - Segmento de edificações



# Introdução

## Homenagem ao Eng. John Backus

- Criador da linguagem Fortran
  - Ferramenta pioneira para o desenvolvimento de softwares por engenheiros
  - Computador IBM na década de 1950



```

C----- TORCAO.FOR
C
C      SUBROUTINE MTOR(NTRAMO,XSEC,AMT)
C
C  $INCLUDE: 'LINCIS.CMM'
C
C      EXCENF=0.001
C      IF(NTRAMO.EQ.NAPOIO)EXCENF=DEPFE*100.
C      IF(EXCENF.LT.0.001)EXCENF=0.001
C
C      XXX(1)=0.
C      XXX(2)=EXCENI
C      DT=COMPUA(NTRAMO)*100./12.
C      DO 100 I=3,14
C          XXX(I)=XXX(I-1)+DT
100  CONTINUE
C      XXX(15)=XXX(14)+EXCENF
C
C      YYY(1)=MT(NTRAMO,1)
C      DO 200 I=1,13
C          YYY(I+1)=MT(NTRAMO,I)
200  CONTINUE
C      YYY(15)=YYY(14)

```

# Introdução

## Informatização do mercado brasileiro

- Pioneirismo em estruturas de concreto armado
- Pioneirismo em softwares integrados para estruturas de concreto armado
- Cerca de 99% das empresas de projeto estão informatizadas

# Introdução

## Informatização do mercado brasileiro

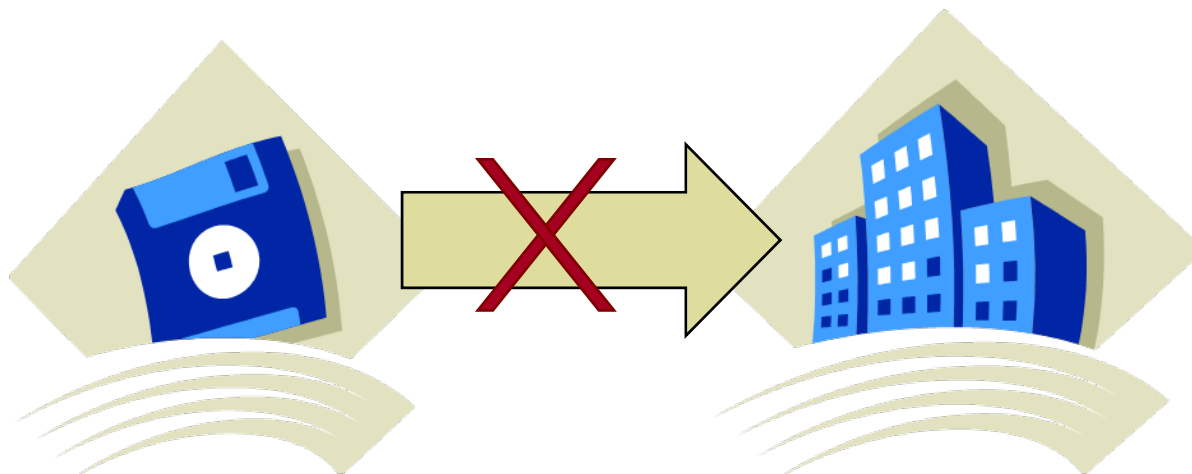
- Não existe competitividade sem um bom software
- Grande dependência entre projetista - software
- Competem empresas de software nacionais e do exterior (USA, Espanha, França, Israel, Coréia, Finlândia, etc.)

# 2

## Relação Software e Projeto Estrutural

# Relação entre Software e Projeto Estrutural

- Software não elabora projetos automaticamente
- Software é apenas uma “ferramenta” de projeto



# Relação entre Software e Projeto Estrutural

- Software apenas realiza cálculos, emite relatórios alfanuméricos, gráficos e desenhos
- O engenheiro estrutural deve analisar, verificar e validar os resultados emitidos tornando-os um projeto estrutural executável

# Relação entre Software e Projeto Estrutural

Exemplos de cláusulas contratuais para fornecimento de software:

# Relação entre Software e Projeto Estrutural

Exemplos de cláusulas contratuais:

“ **Limitação da Responsabilidade:** Em nenhuma hipótese a responsabilidade da CEDENTE por indenizações, perdas, ..... decorrentes direta ou indireta, da aquisição e/ou utilização dos SISTEMAS ocasionados à CESSIONÁRIA ou a terceiros ultrapassará o valor pago pela CESSIONÁRIA pelos SISTEMAS.

Em qualquer caso, devido ao estágio atual dos conhecimentos de informática, não é possível garantir totalmente a inexistência de erros nos SISTEMAS, a responsabilidade da CEDENTE pelos danos imputáveis diretamente aos SISTEMAS, **limitar-se-á ao preço pago pela CESSIONÁRIA** pela obtenção da presente Licença de Uso do SISTEMA. ”

# Relação entre Software e Projeto Estrutural

Exemplos de cláusulas contratuais:

“ 10. Limitações: a LICENCIADORA e seus distribuidores não se responsabilizam por quaisquer danos ou prejuízos, diretos ou indiretos, advindos do uso incorreto do programa. A LICENCIADORA não se responsabiliza pela perda de informações decorrente da utilização do programa ou danos ou prejuízos daí advindos. Todas as garantias descritas neste contrato limitam-se exclusivamente ao **valor pago pelo LICENCIADO ao adquirir o programa.** ”

# Relação entre Software e Projeto Estrutural

## Exemplos de cláusulas contratuais:

### “ Garantia Limitada do Software

A YYYYYYYY não se responsabiliza pelas garantias implícitas, incluindo garantias de comercialização, de desempenho e adequação a uma finalidade específica. A YYYYYYYY não será responsável por qualquer falha, erro, omissão, defeito ou não conformidade em qualquer software. Como resultado, o software é vendido no seu estágio atual de desenvolvimento e o comprador assume todo o risco quanto à sua qualidade e desempenho.

### Limitação de Responsabilidade

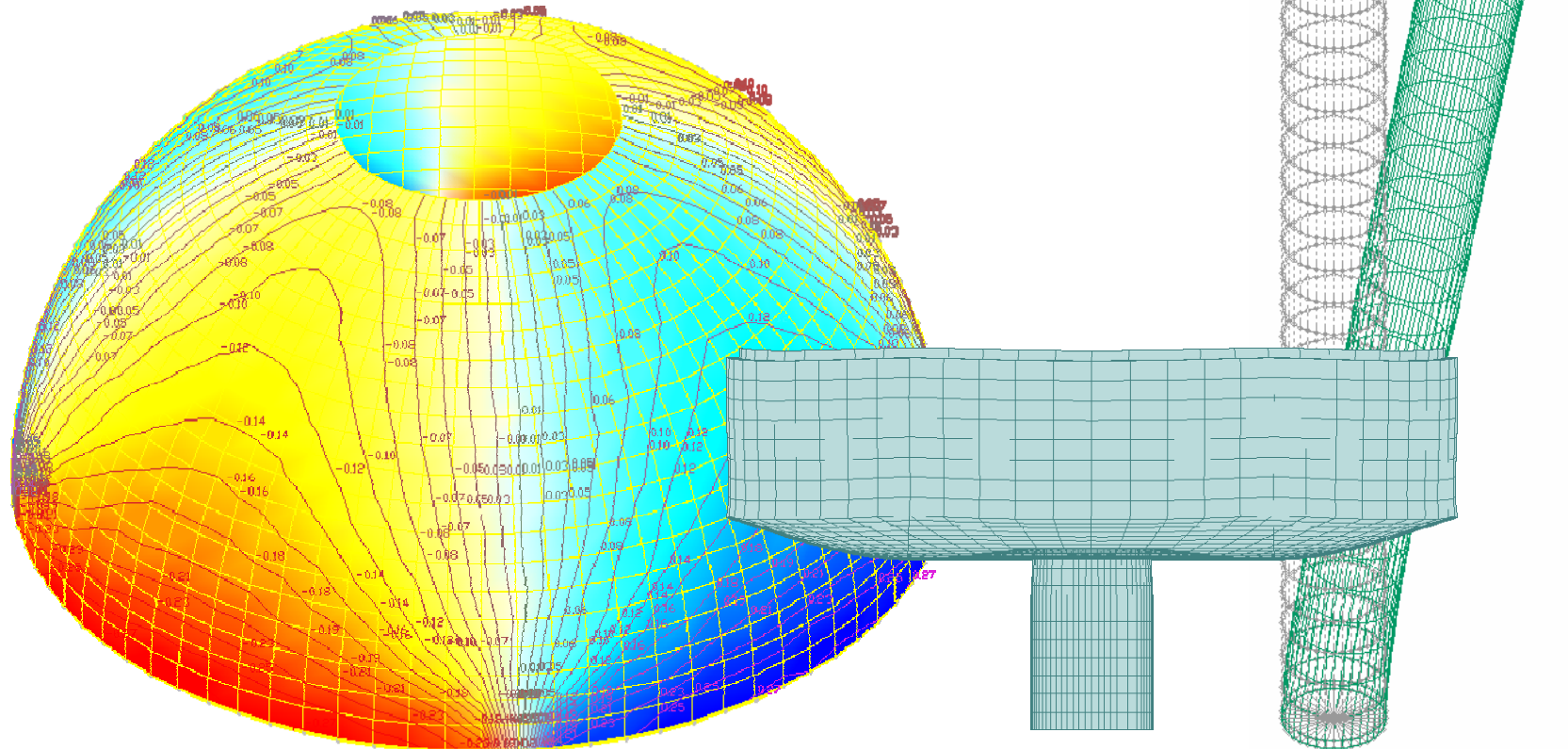
A YYYYYYYY em nenhum caso será responsabilizada por danos diretos, indiretos, ....., resultantes de qualquer defeito no software ou sua documentação, ..... Você concorda que nossa responsabilidade decorrente do contrato, negligência, estrita responsabilidade civil extracontratual ou garantia, **não deve exceder um montante pago por você por este produto.** ”

# 3

## Abrangência dos Softwares (Individuais e Integrados)

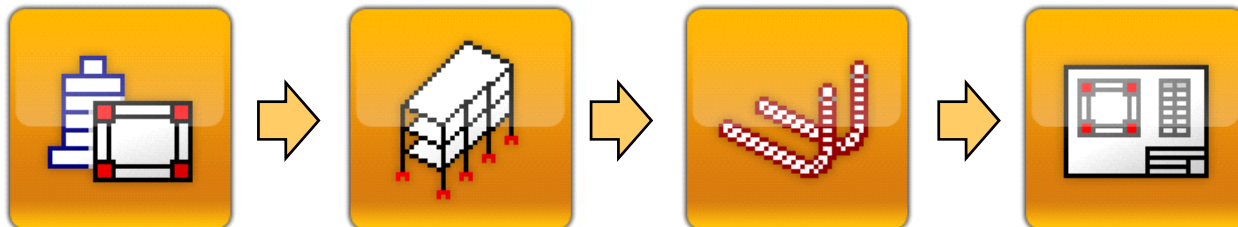
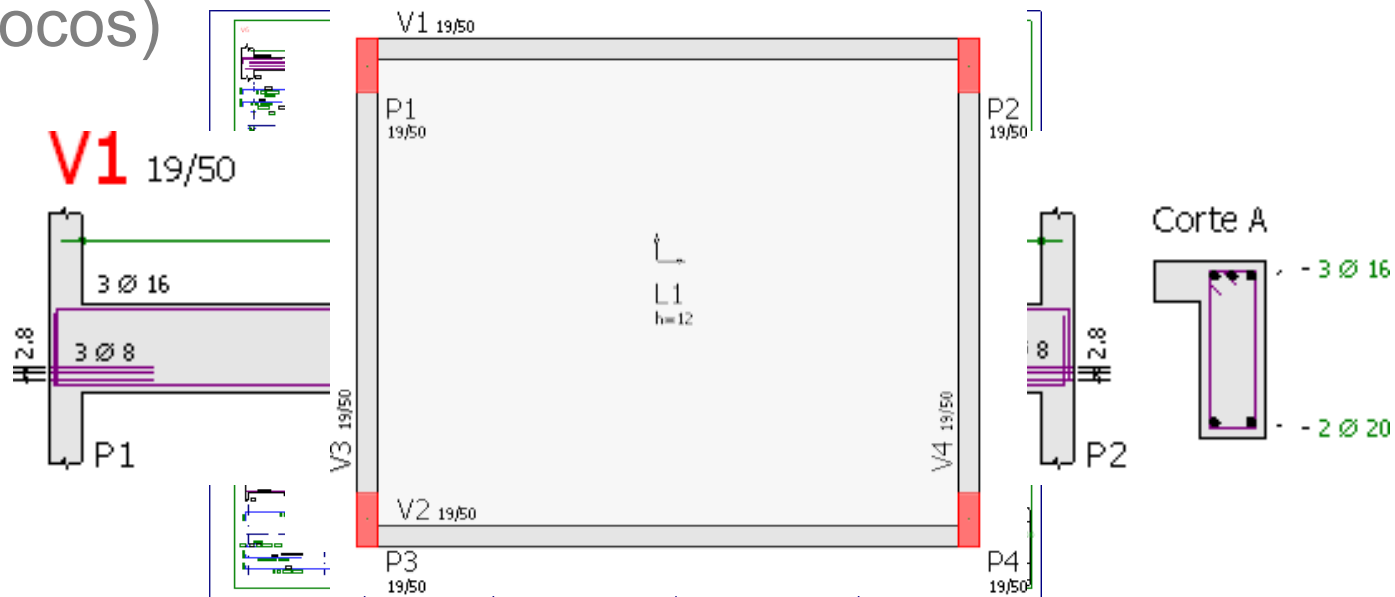
# Abrangência dos Softwares

- Individuais: Somente esforços solicitantes (estruturas genéricas)



# Abrangência dos Softwares

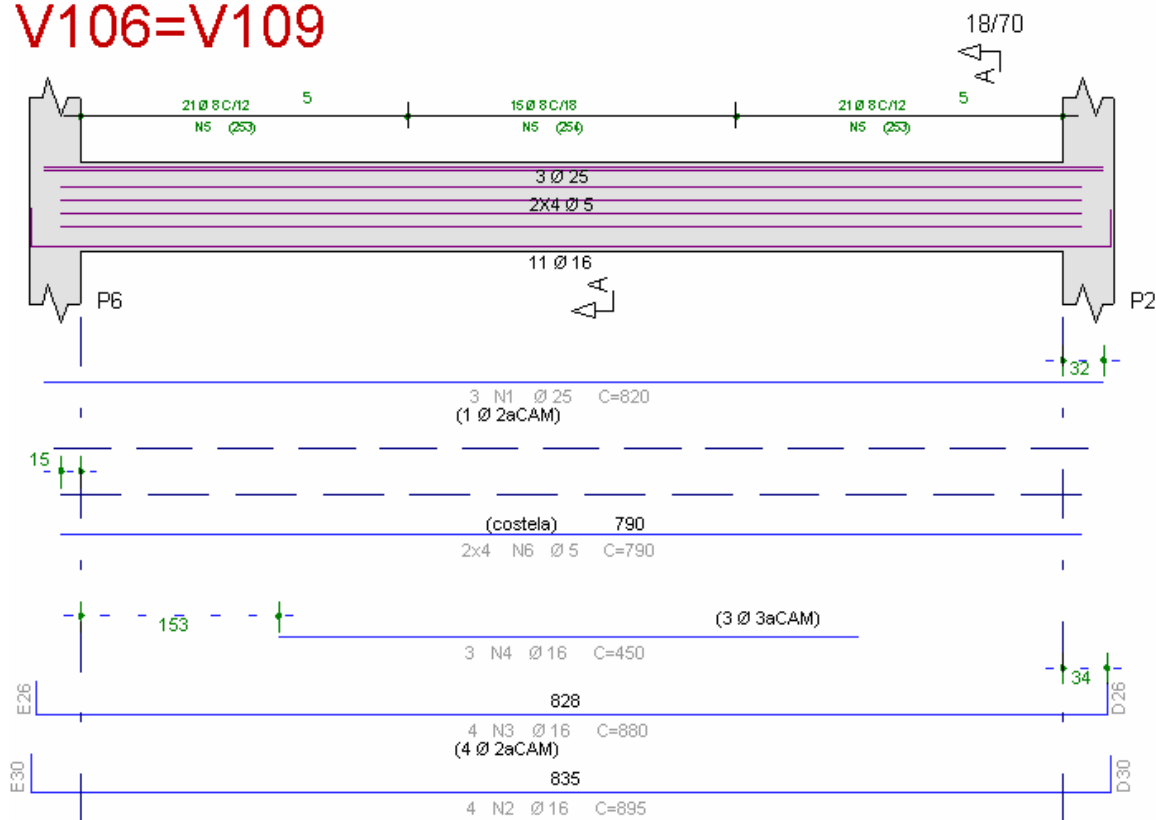
- Integrados: Modelagem, esforços, dimensionamento, detalhamento e desenho (vigas, lajes, pilares, sapatas e blocos)



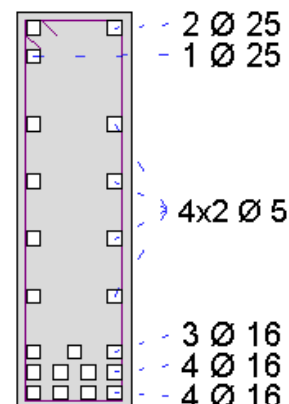
# Abrangência dos Softwares

- Detalhamento influi na fase de dimensionamento (alojamento de armaduras)

V106=V109



Corte A



# Abrangência dos Softwares

- O elevado grau de automação implica em maiores cuidados e responsabilidades com os resultados obtidos. Perigoso pecado da omissão – **enviar para a obra os desenhos gerados automaticamente**
- Softwares integrados: mais abrangentes, mais automáticos e mais difíceis de assegurar a qualidade

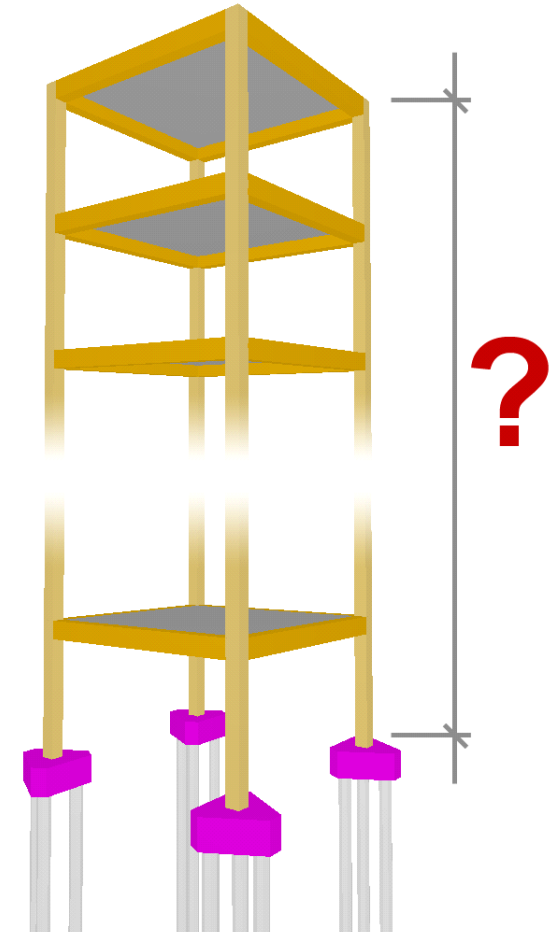
# 4

## Dificuldades na Produção de Software

Por que a maioria “morre”?

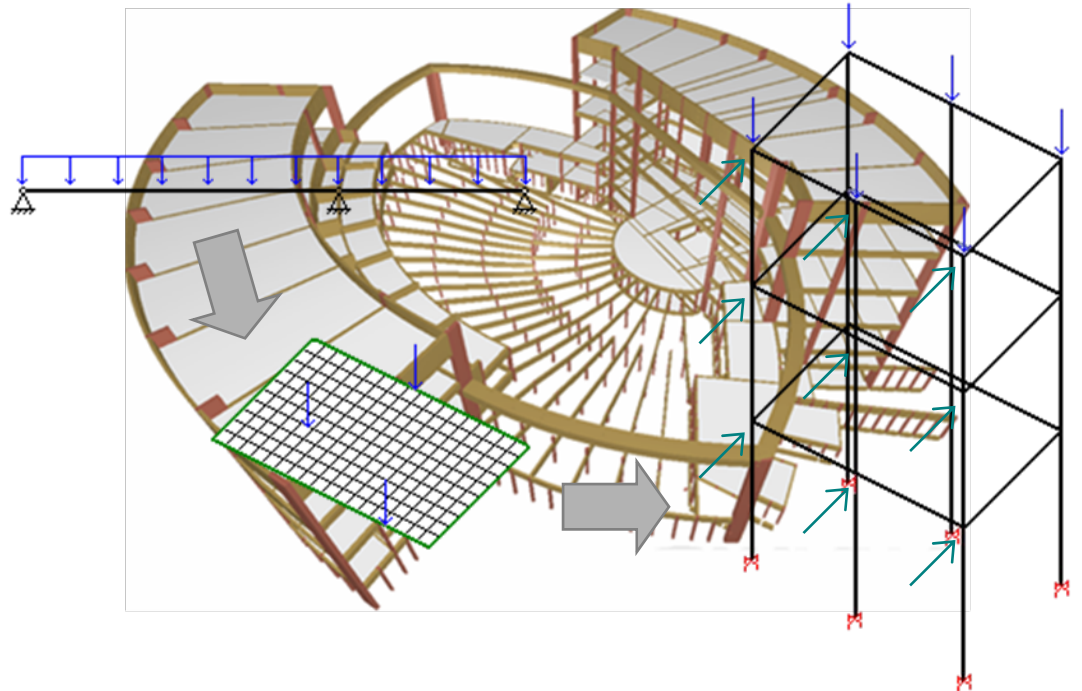
# Por que a maioria dos softwares “morre”?

- Dificuldades de planejamento
  - Geralmente a 1ª. versão do software é  $\leq 20\%$  da final
  - Analogia: projetar a fundação sem conhecer o porte da edificação



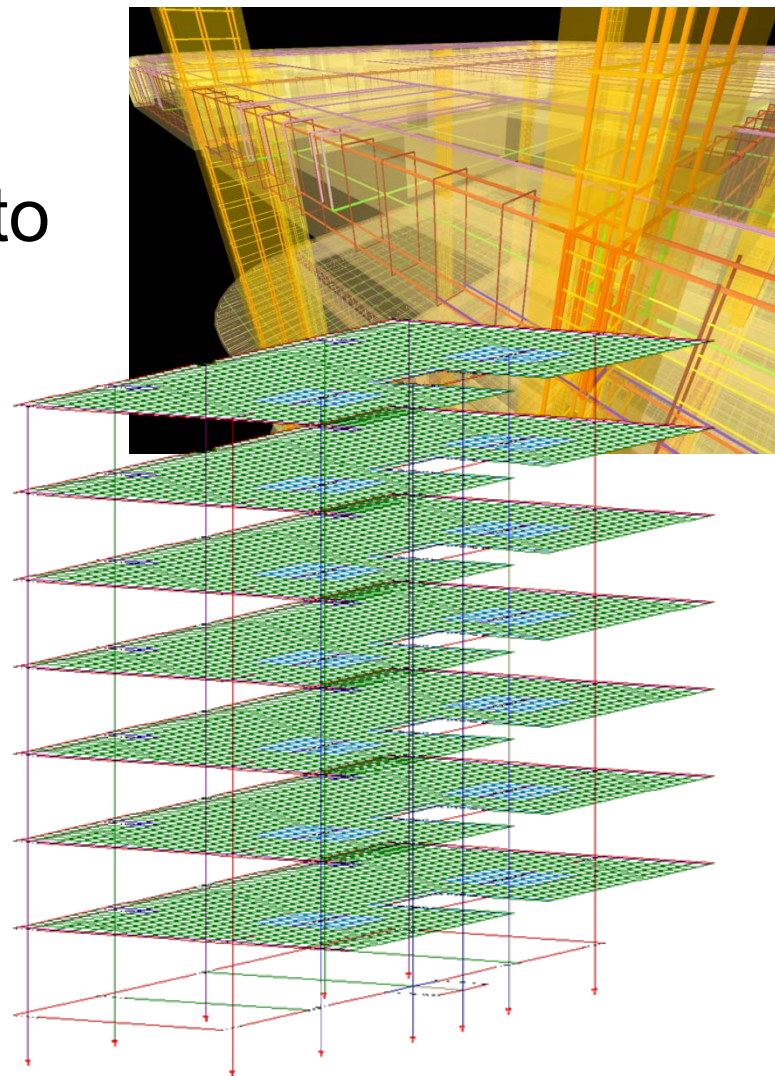
# Por que a maioria dos softwares “morre”?

- Dificuldades de planejamento
  - Evolução nos últimos 25 anos
    - Viga contínua
    - Grelha
    - Pórtico espacial
    - Estruturas mais arrojadas



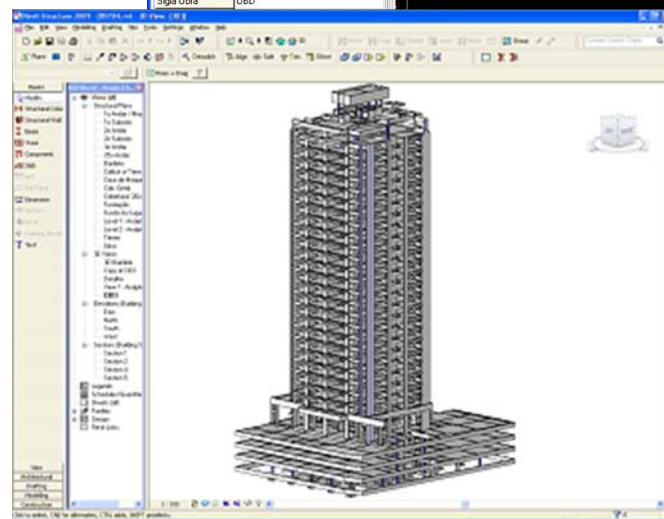
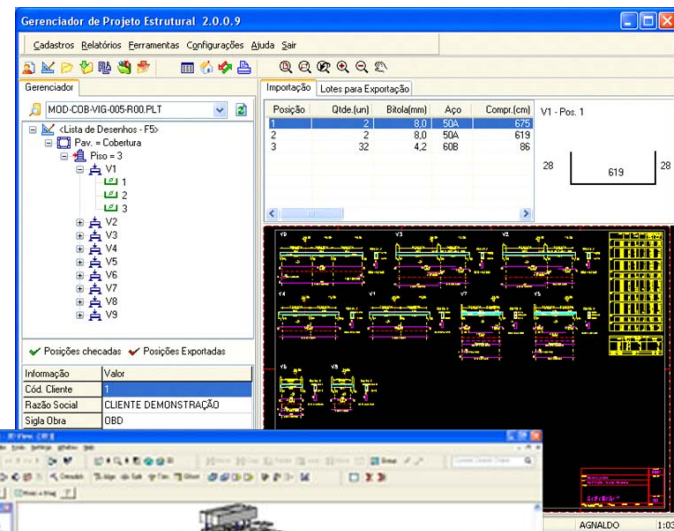
# Por que a maioria dos softwares “morre”?

- Dificuldades de planejamento
  - Evolução na próxima década
  - Abrangência de elementos tratados, efeitos construtivos, interação solo estrutura, armadura integração em 3D, etc.
  - Novos modelos estruturais



# Por que a maioria dos softwares “morre”?

- Dificuldades de planejamento
  - BIM – Building **Information** Modeling
- Será a maior exigência no futuro próximo
- Geometria 3D – Quantitativo
- Armaduras – Integração Projeto – Corte e Dobra

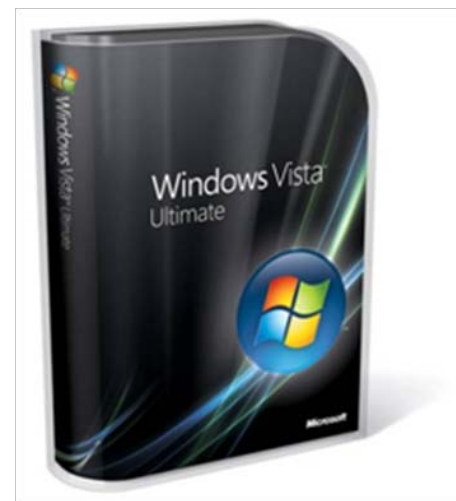


# Por que a maioria dos softwares “morre”?

- Principais causas:
  - Software pessoal – abandono pelo desenvolvedor
  - Falta de investimentos – desenvolver custa caro
  - Dificuldades técnicas de programação
    - Relação com o porte do software – Obstáculos para emissão de novas versões

# Por que a maioria dos softwares “morre”?

- Principais causas:
  - Suporte técnico
  - Mudança de plataforma de hardware e software básico (Windows)
  - Concorrência e competitividade (globalização) / pirataria



# Por que a maioria dos softwares “morre”?

- Conclusões
  - Software é um produto, de certa forma, volátil, frágil, sem valor material agregado e em constante evolução.
  - Centenas de softwares para engenharia - 95% já abandonados
  - Qual o custo do software que morreu?
  - Influência direta na qualidade do software

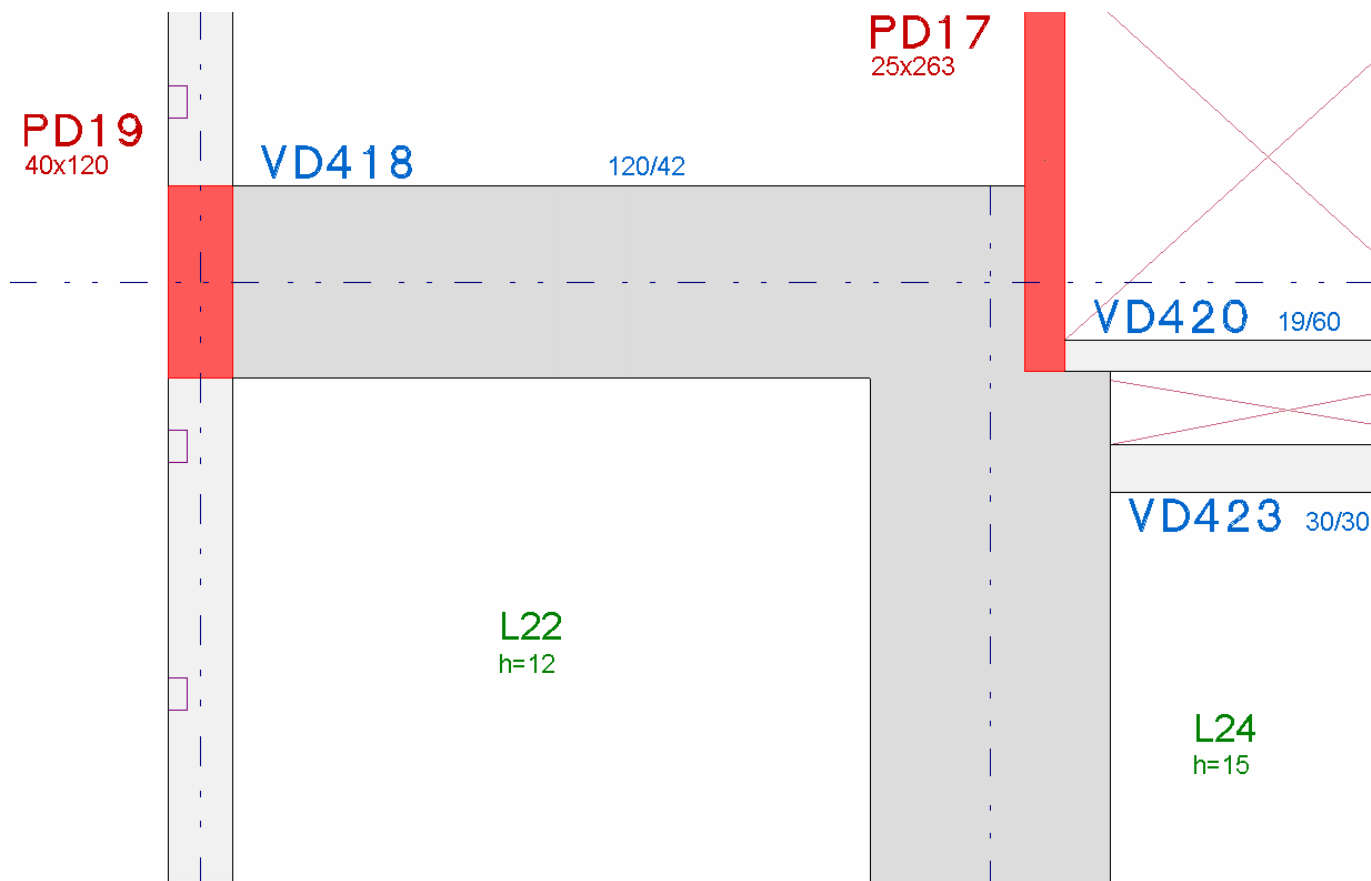


# 5

## Dificuldades na Aplicação (Software Integrado)

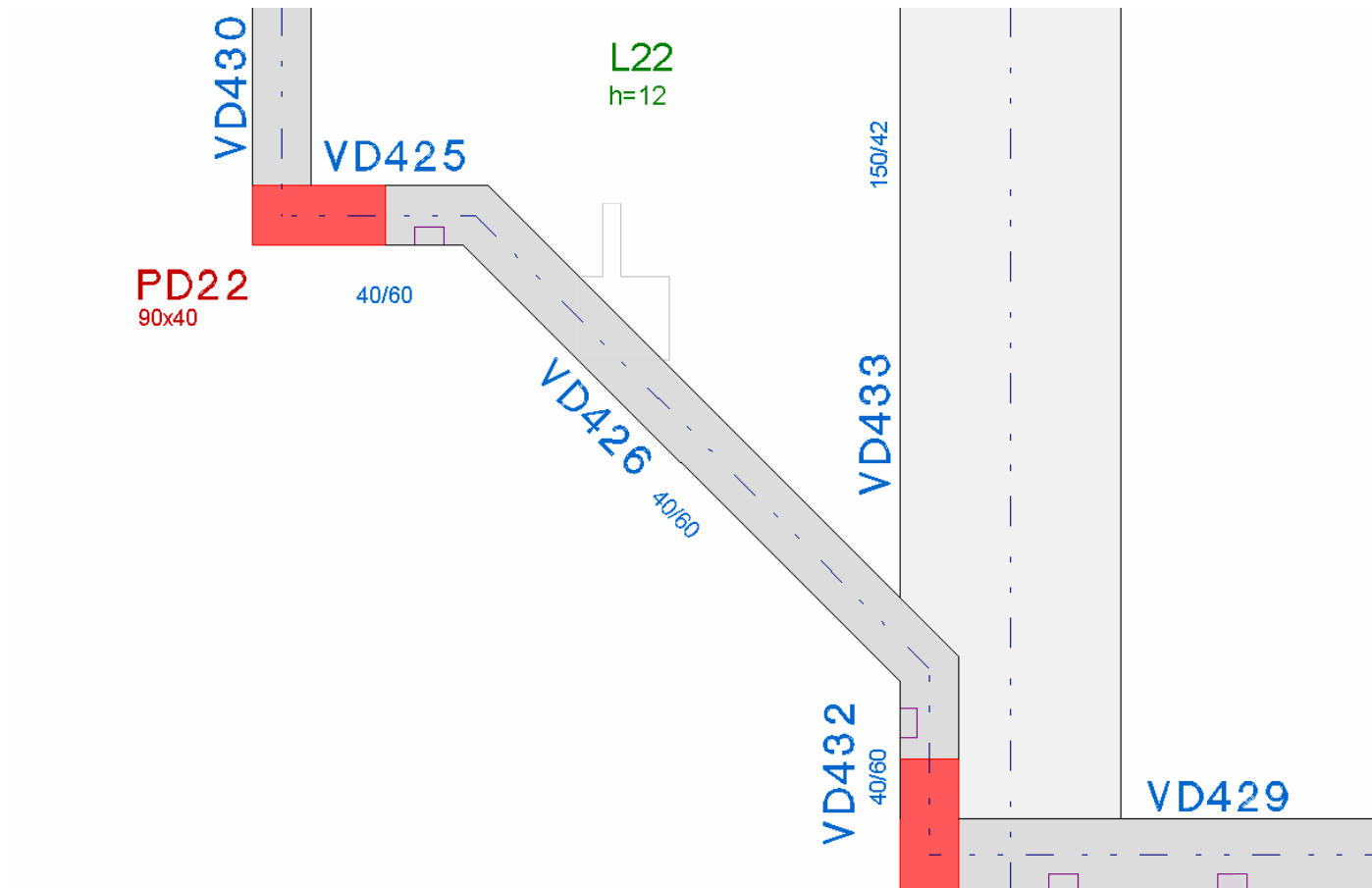
# Exemplos de aplicação de software integrado

- Vigas: apoio de vigas em pilares



# Exemplos de aplicação de software integrado

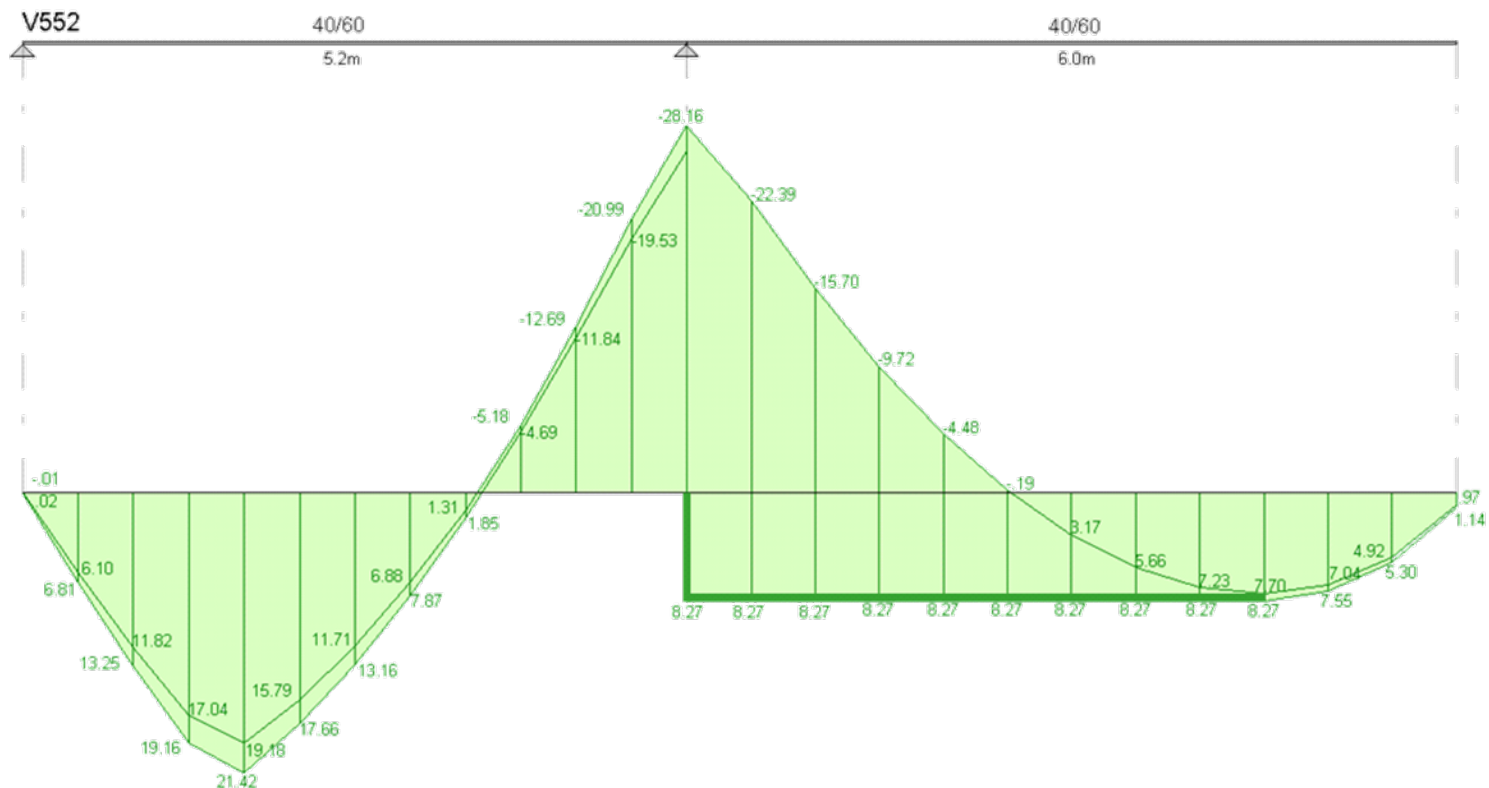
- Vigas: apoio de vigas em pilares





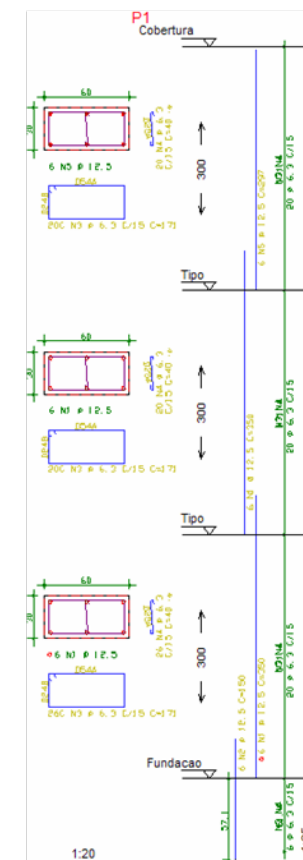
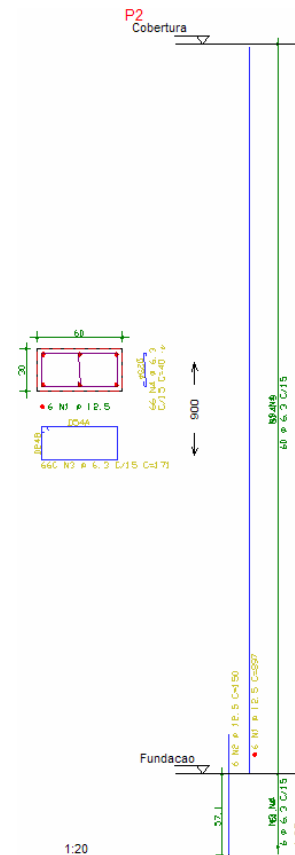
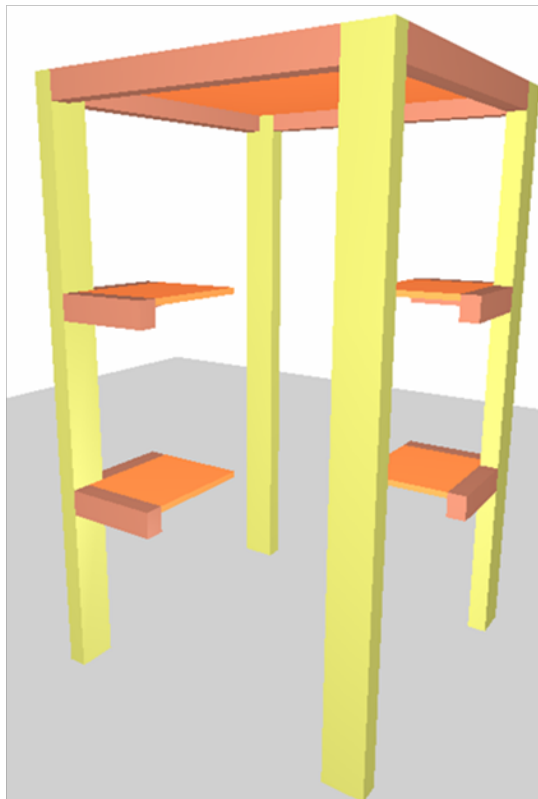
# Exemplos de aplicação de software integrado

- Vigas: detalhamento de vão e apoio
  - Momento positivo no balanço (?)



# Exemplos de aplicação de software integrado

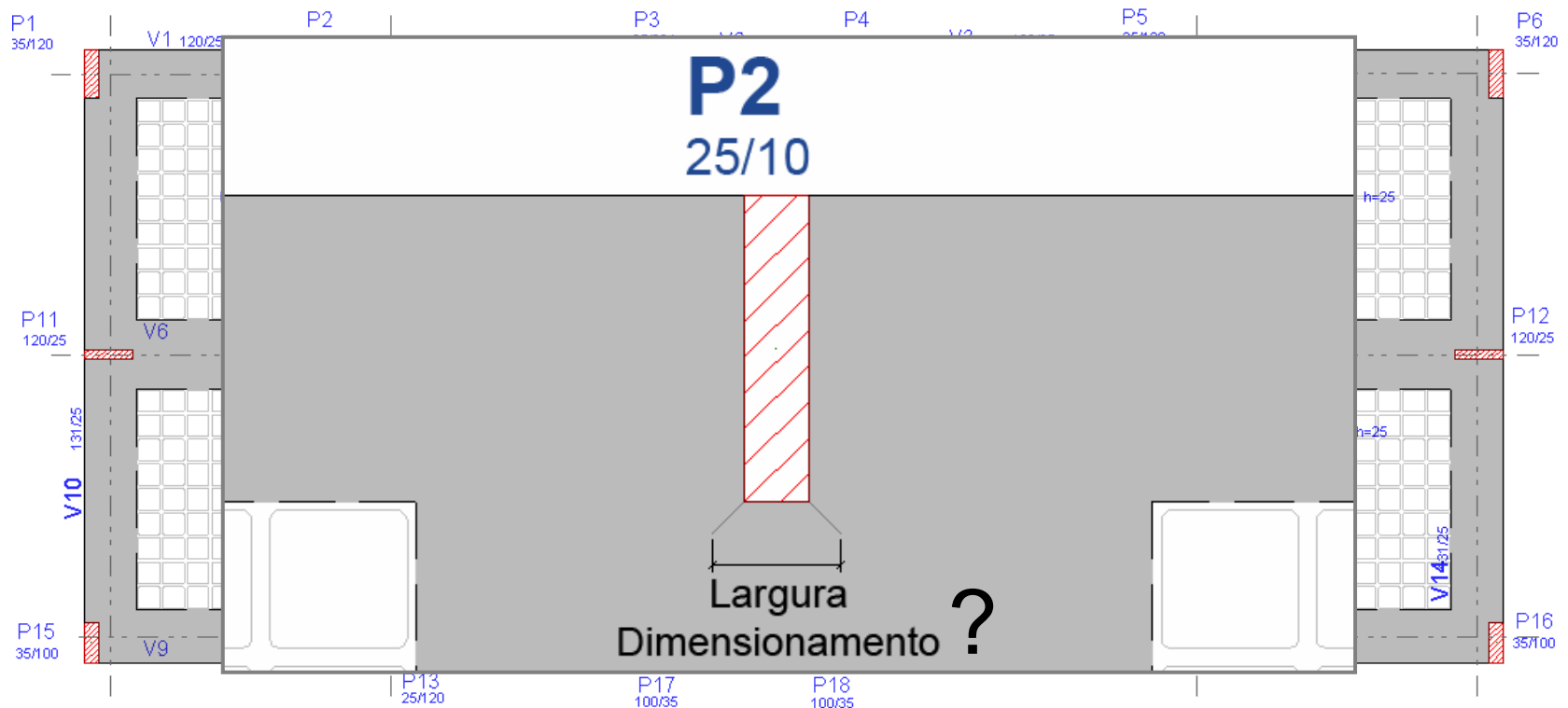
- Pilares: efeitos de segunda ordem. Travamento?

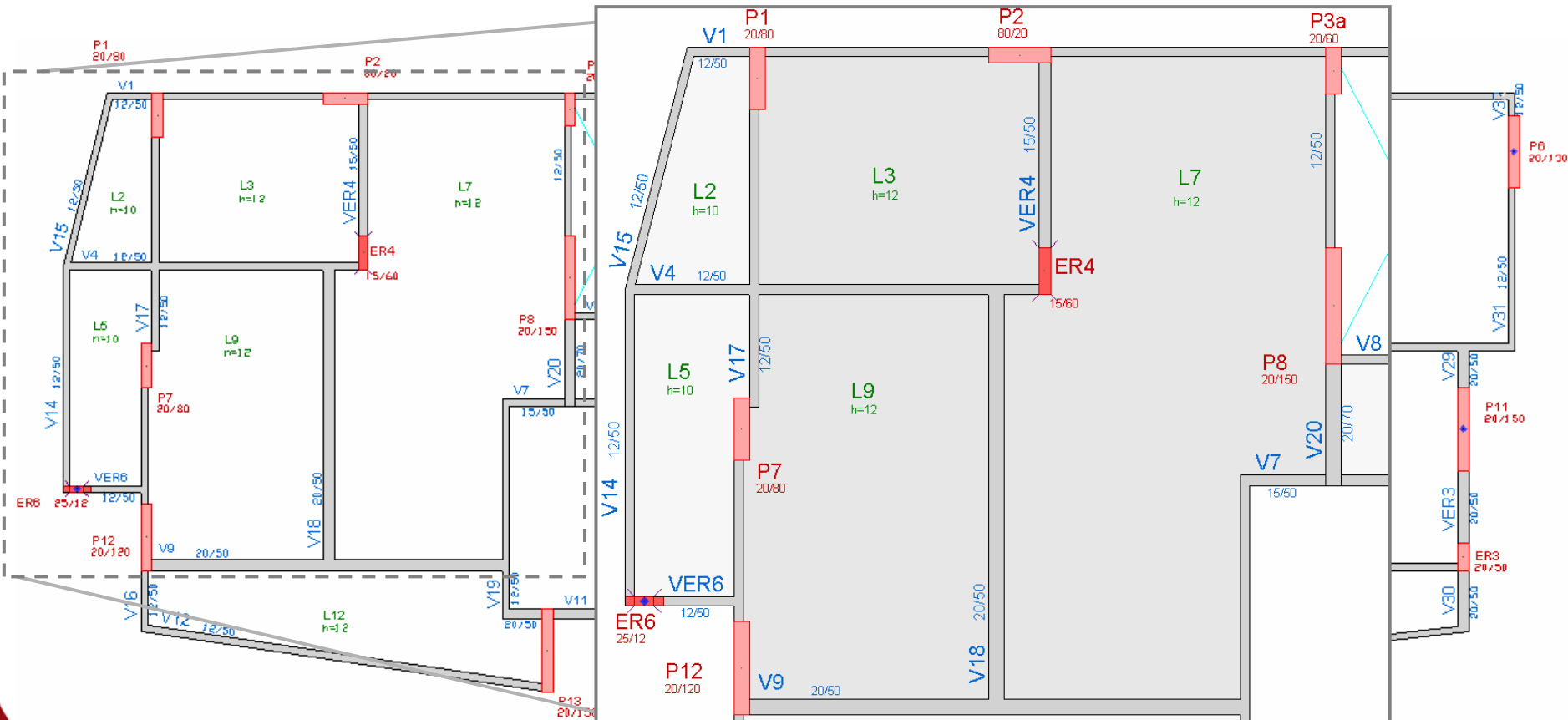




# Exemplos de aplicação de software integrado

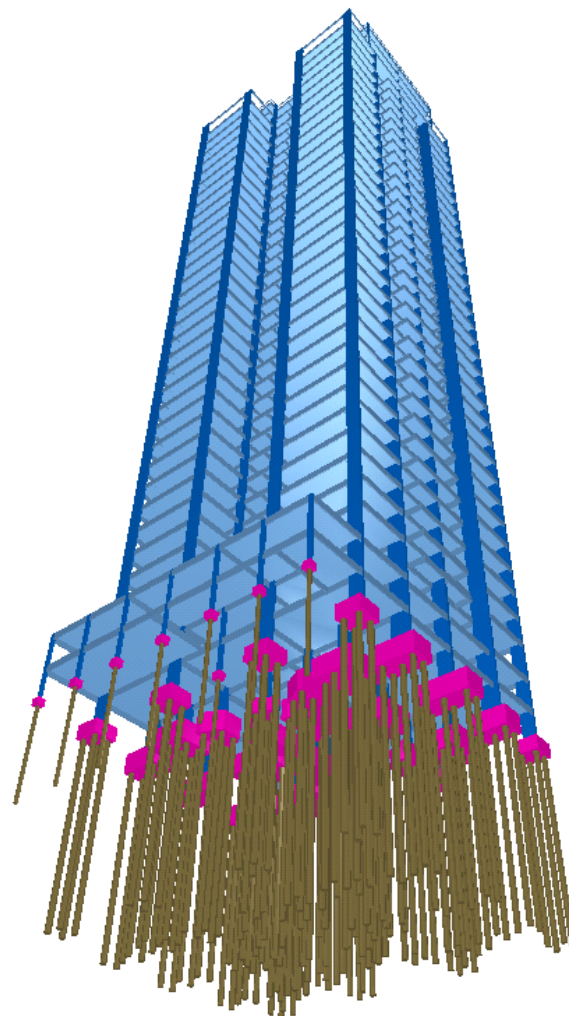
- Lajes: vigas faixas suportando cargas horizontais.  
E o dimensionamento / detalhamento?  
É possível?





# Exemplos de aplicação de software integrado

- Algumas Conclusões
  - Todo software integrado tem limitações importantes para tratamento de determinados elementos
  - Todo software tem simplificações em relação ao comportamento do modelo real. Simplificação relevante atual: interação solo-estrutura



## Exemplos de aplicação de software integrado

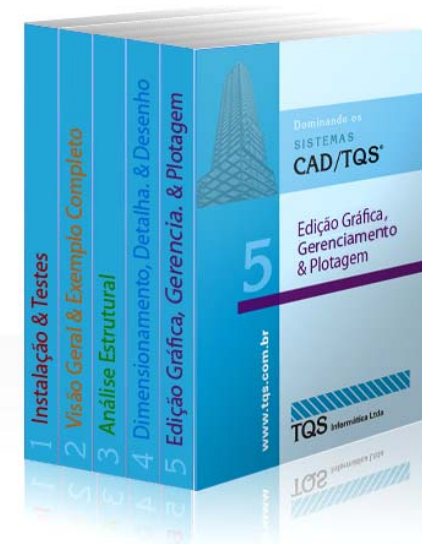
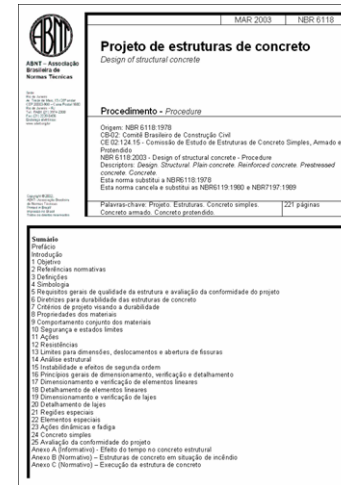
- Algumas Conclusões
  - É praticamente impossível especificar casos onde o software não é aplicável
- Qualidade do software em sistemas integrados está intimamente ligada a sua abrangência e a tomada de decisões em elementos estruturais não triviais

# 6

## Características de Qualidade de um Software Integrado (Usuários)

# Características da qualidade de um software (Usuários)

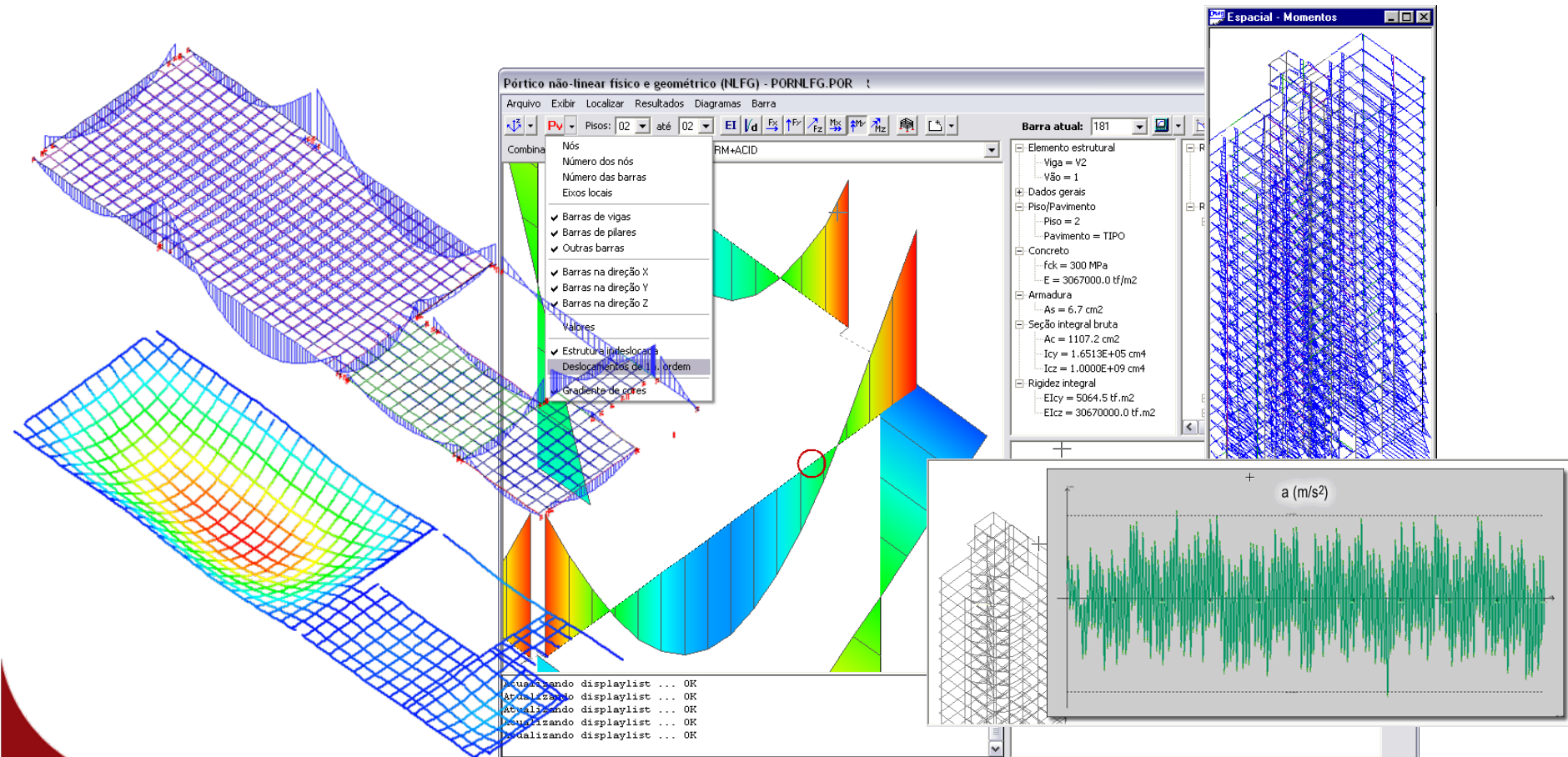
- Atendimento às normas técnicas
- Documentação completa (operacional e fundamentos teóricos)
- Disponibilidade de suporte técnico



# Características da qualidade de um software

(Usuários)

- Apresentação de resultados gráficos

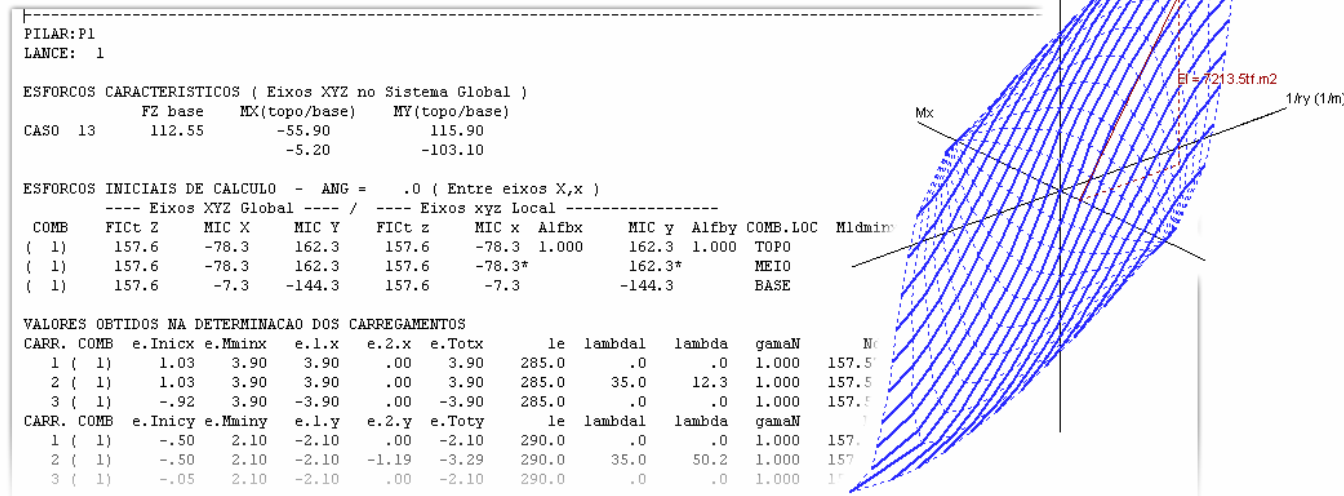


# Características da qualidade de um software

(Usuários)

- Apresentação de todas as etapas de cálculos realizadas

– Ex. Pilares: complexidade inevitável



100 (comb.) x 3 (pontos) x 2 (direc.) x 30 (pvts)  
x 20 (pilares) = 360.000 linhas → Consulta

# Características da qualidade de um software

(Usuários)

- Diálogo do software com o usuário:
  - Apresentação das anomalias, alertas, advertências

T Q S INFORMATICA LTDA / RUA DOS PINHEIROS, 706 / SAO PAULO SP 05422-001 / 56.555.212/0001-90 2008-TQS

Edifício CMAQ 0018 D:\TQS\MODPLA\CMAQ D:\TQS\MODPLA

Pavimento CMAQ Pasta [D:\TQS\MODPLA\CMAQ]

Gravando o arquivo [D:\TQS\MODPLA\CMAQ\CMAQ.LDF]

AVISO: Viga PAR.1 com rebaixo/elevação muito grande. Os rebaixos não são modelados nos sistemas Grelha e Pórtico-TQS (no pórtico podem ser modelados com restrições). Para levar este e outros rebaixos em consideração, verifique a possibilidade de criar novos pavimentos ou pisos auxiliares

TESTE DE VALIDAÇÃO DA INSTALAÇÃO / TQS Informatica LTDA

CAD/Formas CARPIL.DWG

Editar

Limpar

T Q S INFORMATICA LTDA / RUA DOS PINHEIROS, 706 / SAO PAULO SP 05422-001 / 56.555.212/0001-90 2008-TQS

Edifício CMAQ 9003 Pav TERREO 0011 D:\TQS\MODPLA\TERREO

AVISO: Dimensão da viga V9 (10.00) menor que a mínima recomendada pela NBR-6118:2003 (12.00). Será necessário verificar o alojamento de armaduras e se o lançamento e vibração do concreto obedece a NBR-14931

AVISO: O pilar P10 nasce na viga V5 de maneira excêntrica com a viga. Para que não surjam momentos excêntricos irreais, defina um nó mais próximo do CG do pilar.

TESTE DE VALIDAÇÃO DA INSTALAÇÃO / TQS Informatica LTDA

CAD/Formas

Editar

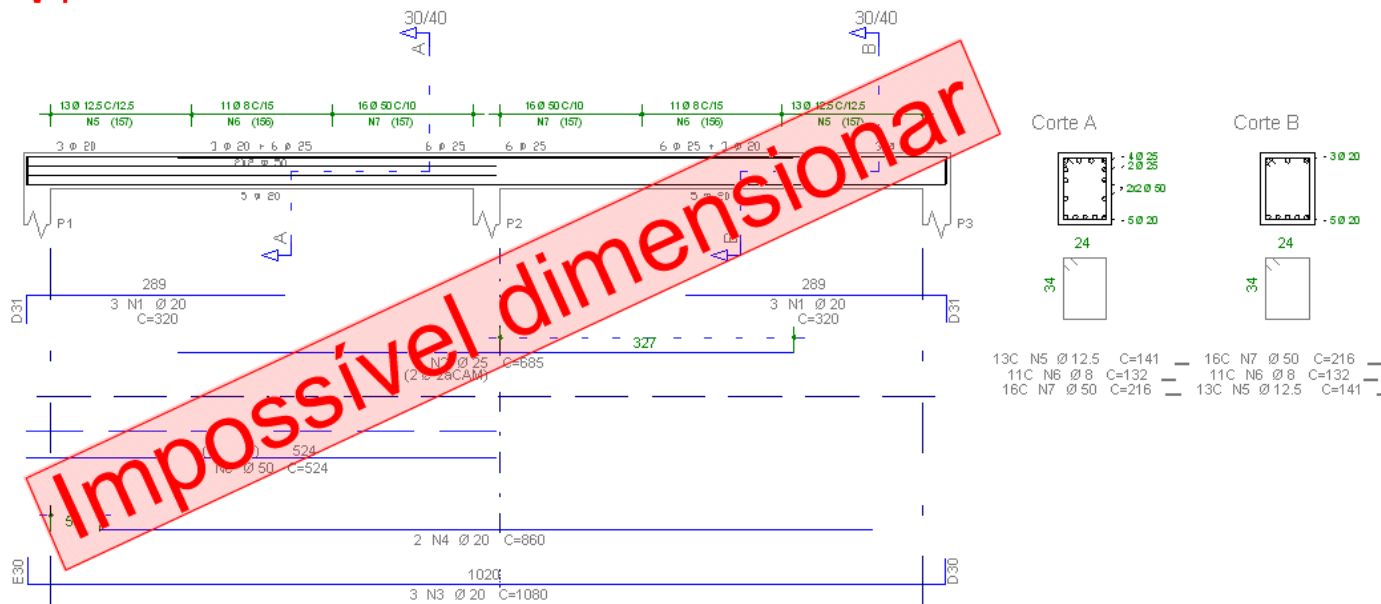
Limpar

# Características da qualidade de um software

(Usuários)

- Diálogo do software com o usuário:
  - Aviso de impossibilidade de dimensionamento, detalhamento

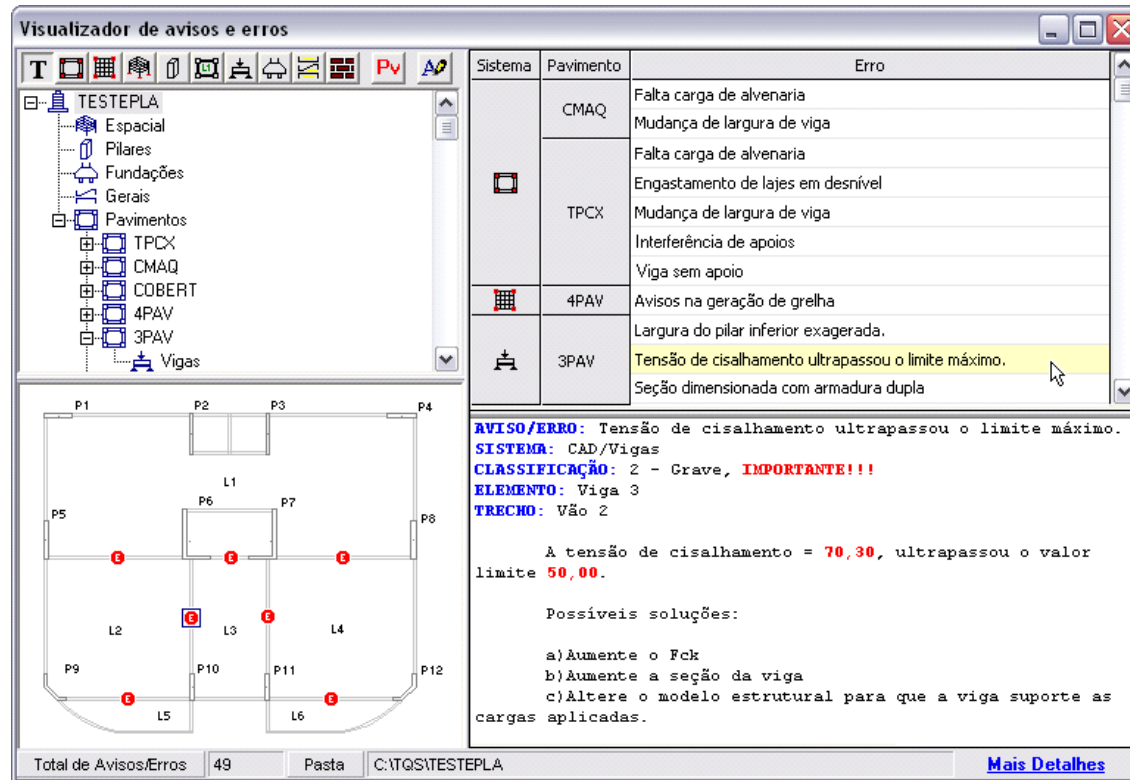
V1



# Características da qualidade de um software

(Usuários)

- Diálogo do software com o usuário:
  - Instruções para tomada de decisão



**Visualizador de avisos e erros**

| Sistema | Pavimento | Erro                                                                                                                                     |
|---------|-----------|------------------------------------------------------------------------------------------------------------------------------------------|
|         | CMAQ      | Falta carga de alvenaria<br>Mudança de largura de viga                                                                                   |
|         | TPCX      | Falta carga de alvenaria<br>Engastamento de lajes em desnível<br>Mudança de largura de viga<br>Interferência de apoios<br>Viga sem apoio |
|         | 4PAV      | Avisos na geração de grelha<br>Largura do pilar inferior exagerada.                                                                      |
|         | 3PAV      | <b>Tensão de cisalhamento ultrapassou o limite máximo.</b><br>Seção dimensionada com armadura dupla                                      |

**AVISO/ERRO:** Tensão de cisalhamento ultrapassou o limite máximo.  
**SISTEMA:** CAD/Vigas  
**CLASSIFICAÇÃO:** 2 - Grave, **IMPORTANT!!!**  
**ELEMENTO:** Viga 3  
**TRECHO:** Vão 2

A tensão de cisalhamento = 70,30, ultrapassou o valor limite 50,00.

Possíveis soluções:

- a) Aumente o Fck
- b) Aumente a seção da viga
- c) Altere o modelo estrutural para que a viga suporte as cargas aplicadas.

Total de Avisos/Erros: 49    Pasta: C:\TQS\TESTEPLA    [Mais Detalhes](#)

# Características da qualidade de um software

(Usuários)

- Diálogo do software com o usuário:
  - Resumo das informações do processamento

## Resumo estrutural

### Conteúdo

[Dados do edifício](#)  
[Parâmetros de durabilidade](#)  
[Modelo estrutural](#)  
[Ações](#)  
[Estabilidade global](#)  
[Análise em serviço - ELS](#)  
[Parâmetros qualitativos](#)  
[Parâmetros quantitativos](#)  
[Consumo](#)  
[Avisos e erros](#)

### Estabilidade global

#### Parâmetros de instabilidade

| Parâmetro | Valor máximo |
|-----------|--------------|
| GamaZ     | 1.09         |
| FAVt      | 1.25         |
| Alfa      | .94          |

#### Avaliação e classificação da estrutura

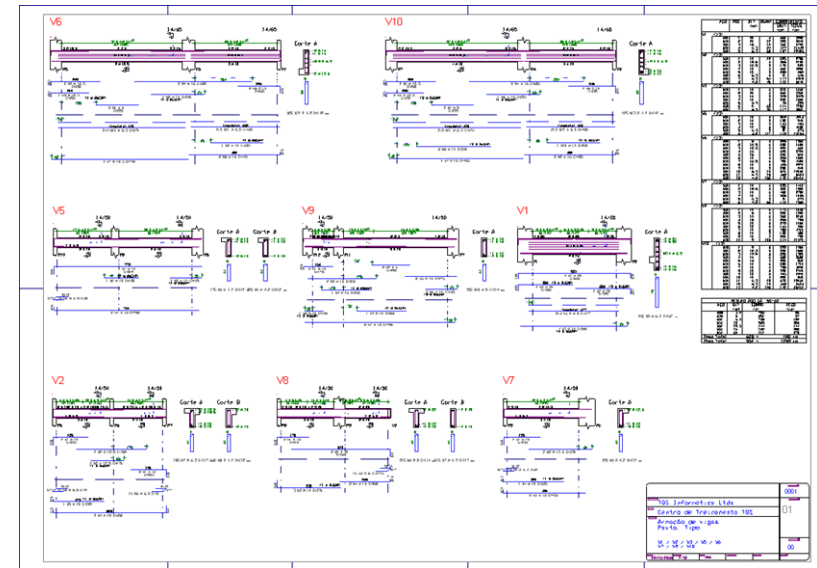
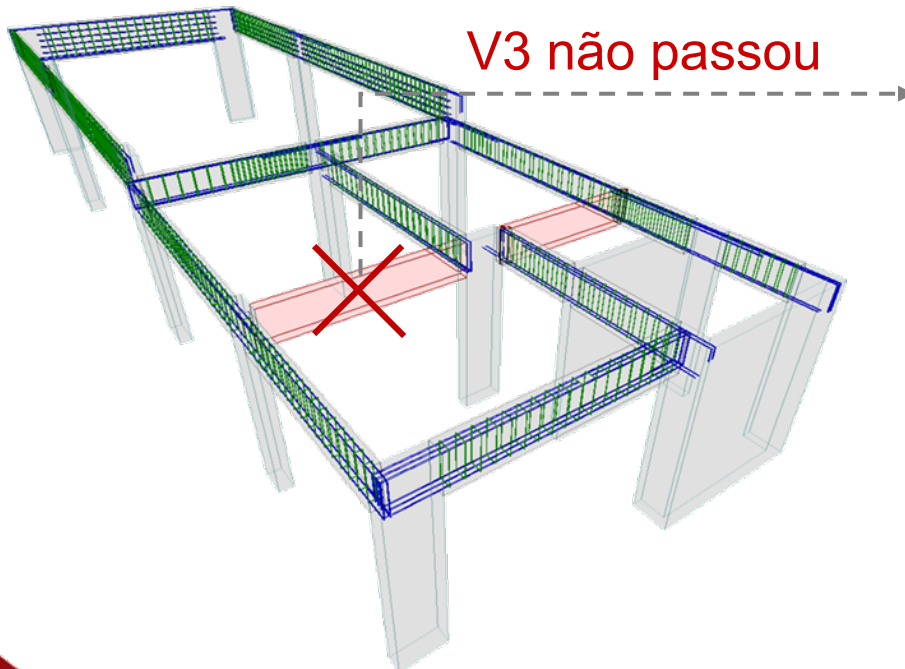
Parâmetro adotado na análise do edifício ..... 1.25 (Acima do limite)  
 Valor limite de referência ..... 1.20  
 Tipo da estrutura ..... Nós móveis

Verifique este item.

# Características da qualidade de um software

(Usuários)

- Diálogo do software com o usuário:
  - Exemplo indevido: elementos que não “passam” não são desenhados

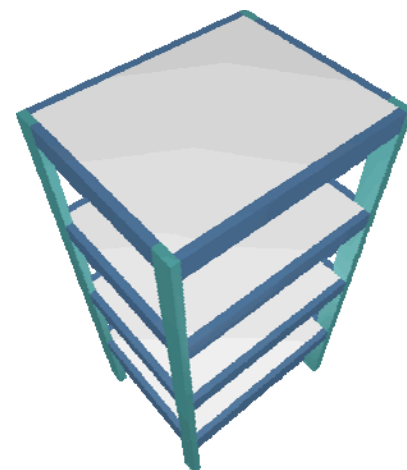


✗ Planta enviada sem a V3

# Características da qualidade de um software

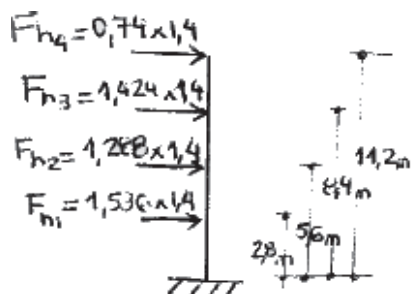
(Usuários)

- Controle de qualidade: testes teóricos padronizados



Parâmetro de estabilidade (GamaZ) para os carregamentos simples de vento

| Caso | Ang  | CTot  | M2  | CHor | M1   | Mig | GamaZ | Alfa | Obs |
|------|------|-------|-----|------|------|-----|-------|------|-----|
| 5    | 90.  | 147.0 | .4  | 6.6  | 42.0 | 3.0 | 1.017 | .237 |     |
| 6    | 270. | 147.0 | .4  | 6.6  | 42.0 | 3.0 | 1.017 | .237 |     |
| 7    | 0.   | 147.0 | 1.9 | 5.0  | 31.8 | 3.0 | 1.121 | .604 | B   |
| 8    | 180. | 147.0 | 1.9 | 5.0  | 31.8 | 3.0 | 1.121 | .604 | B   |



$$M_{1d} = (1,536 \times 1,4) \times 28 + (1,288 \times 1,4) \times 56 + (1,424 \times 1,4) \times 84 + (0,74 \times 1,4) \times 112$$

$$M_{1d} = 44,46 \text{ tf.m}$$

$$M_{2d} = (42,7 \times 1,4 \times 0,00807 \times 1,4) + (42,7 \times 1,4 \times 0,01811 \times 1,4) + (42,7 \times 1,4 \times 0,2509 \times 1,4) + (19 \times 1,4 \times 0,2883 \times 1,4) =$$

$$M_{2d} = 5,37 \text{ tf}$$

$$\delta_z = \frac{1}{1 - \frac{1}{1,1} \cdot \frac{5,37}{44,46}} = 1,12 \rightarrow \delta_z = 1,12$$

✓ OK

# Características da qualidade de um software

(Usuários)

- Controle de qualidade: testes com projetos reais, testes e mais testes

Planilha de Testes Automáticos

|            |         | Sistemas TQS |     |       |     |     |     |       |     |     |     |         |     |     |     |        |     |     |     |          |     |     |       |     |     |     |    |     |     |     |     |    |
|------------|---------|--------------|-----|-------|-----|-----|-----|-------|-----|-----|-----|---------|-----|-----|-----|--------|-----|-----|-----|----------|-----|-----|-------|-----|-----|-----|----|-----|-----|-----|-----|----|
| DATA       | USUÁRIO | VersãoTQS    | LDF | VIGAS |     |     |     | PILAR |     |     |     | PORTICO |     |     |     | GRELHA |     |     |     | Fundação |     |     | SISES |     |     |     |    |     |     |     |     |    |
|            |         |              |     | VIG   | VFC | VI3 | VI4 | PIL   | PIB | PIC | GER | PED     | POR | MIX | PFL | PNL    | SIS | GRE | GMX | GNL      | VIB | SAP | BLO   | ALV | GLO | BIG | AB | EST | SAP | TUB | P-p |    |
| 23/09/2008 | Ricardo | v14.00.332   | ER  | ER    | ER  | ER  | ER  | OK    | ER  | OK  | OK  | OK      | OK  | OK  | ER  | ER     | OK  | ER  | OK  | OK       | OK  | OK  | ER    | ER  | ER  | ER  | OK | ER  | ER  | OK  | ER  | ER |
| 24/09/2008 | Ricardo | v14.00.336   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | ER  | ER     | ER  | OK  | OK  | OK       | OK  | OK  | ER    | ER  | OK  | OK  | OK | OK  | OK  | OK  | OK  | ER |
| 25/09/2008 | Ricardo | v14.00.337   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | ER     | OK  | OK  | OK  | OK       | OK  | OK  | OK    | OK  | OK  | OK  | OK | OK  | OK  | ER  | OK  | ER |
| 25/09/2008 | Ricardo | v14.00.341   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | ER     | OK  | OK  | OK  | OK       | OK  | OK  | OK    | OK  | ER  | OK  | OK | OK  | OK  | OK  | OK  | OK |
| 26/09/2008 | Ricardo | v14.00.342   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | OK     | OK  | OK  | OK  | OK       | OK  | OK  | OK    | OK  | OK  | ER  | OK | OK  | OK  | OK  | OK  | OK |
| 26/09/2008 | Ricardo | v14.00.343   | OK  | ER    | ER  | ER  | ER  | OK    | OK  | OK  | OK  | OK      | OK  | ER  | OK  | OK     | OK  | OK  | OK  | OK       | OK  | ER  | ER    | ER  | ER  | OK  | OK | OK  | OK  | OK  | OK  | OK |
| 29/09/2008 | Ricardo | v14.00.346   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | OK     | OK  | ER  | ER  | OK       | OK  | OK  | OK    | OK  | OK  | OK  | OK | OK  | OK  | OK  | OK  | OK |
| 29/09/2008 | Ricardo | v14.00.349   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | OK     | OK  | OK  | OK  | OK       | OK  | ER  | ER    | OK  | ER  | ER  | OK | OK  | OK  | OK  | OK  | OK |
| 30/09/2008 | Ricardo | v14.00.351   | ER  | ER    | ER  | ER  | ER  | OK    | OK  | OK  | OK  | OK      | OK  | ER  | ER  | OK     | ER  | OK  | OK  | OK       | ER  | OK  | ER    | ER  | OK  | ER  | OK | OK  | OK  | OK  | OK  | OK |
| 01/10/2008 | Ricardo | v14.00.352   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | OK     | OK  | OK  | OK  | OK       | OK  | OK  | OK    | OK  | OK  | OK  | OK | OK  | OK  | OK  | OK  | OK |
| 02/10/2008 | Ricardo | v14.00.355   | OK  | OK    | OK  | OK  | OK  | OK    | OK  | OK  | OK  | OK      | OK  | OK  | OK  | OK     | OK  | OK  | OK  | OK       | OK  | OK  | OK    | OK  | OK  | OK  | OK | OK  | OK  | OK  | OK  | OK |

Todos

Filtrar por Data

de / / a / /

Filtrar por Usuário

Filtrar por Versão

v -

Legenda:

OK - Teste sem erros, Validado

ER - Há diferenças, Verificar

Todos

Filtrar por Data

Filtrar por Usuário

Filtrar por Versão

Legenda: OK - Teste sem erros, Validado

ER - Há diferenças, Verificar

"Você não pode dialogar com o software até a exaustão, porque as instruções de programação são finitas, mas os diálogos são infinitos.

É impossível testar um software para 100% dos casos possíveis."

# 7 | Metodologia para Garantia de Qualidade do Software (Empresas)

# Como melhorar a qualidade dos softwares?

- Preocupação das empresas desenvolvedoras com a qualidade
  - Qualidade não é custo
  - Todo investimento em qualidade é compensador
  - Sobreviverão apenas empresas que produzam software com certos padrões de qualidade

# Como garantir a qualidade dos softwares (Empresas)?

- Qualidade do **Processo** ou Qualidade do **Produto**?
- Qualidade do **Processo** de Produção de Software
  - Atendimento a classificação internacional CMM /ISO 9000-3
  - Adotado pelo Depto. de Defesa dos EUA – CMM
  - “Modelos de avaliação de qualidade de processo, baseado no conceito de maturidade.”

# Como garantir a qualidade dos softwares (Empresas)?

|                |              |                                                 |
|----------------|--------------|-------------------------------------------------|
| <b>Nível 5</b> | Otimizado    | Otimização do desempenho atual e futuro         |
| <b>Nível 4</b> | Estabelecido | Emprego de princípios de engenharia de software |
| <b>Nível 3</b> | Gerenciado   | Qualidade, recursos e cronograma são atingidos  |
| <b>Nível 2</b> | Executado    | Implementação atingiu o objetivo                |
| <b>Nível 1</b> | Incompleto   | Sem nenhuma implementação                       |

# Como garantir a qualidade dos softwares (Empresas)?

- Qualidade do **Processo** de Produção de Software
  - Qualificação técnica da equipe desenvolvedora (auditorias)
  - Procedimentos adotados no desenvolvimento (auditorias)
  - Atendimento às normas técnicas vigentes

# Como garantir a qualidade dos softwares (Empresas)?

- Qualidade do **Produto**
  - Procedimentos de validação: testes com projetos reais, testes e mais testes

“Não é possível assegurar que o software não tenha erros.”

*Leroy Emkin*

“Não é possível assegurar que o software não tenha erros sérios.”

*Leroy Emkin*

# Como garantir a qualidade dos softwares (Empresas)?

- Importante implantar ambas qualidades:  
**Processo** e **Produto**
- Pouquíssimas empresas no mundo com a qualidade do **Processo** de nível elevado (4-5)
- Qualidade do **Produto** é a mais difundida e aplicada

# 8

## Providências para Melhorar a Qualidade dos Softwares

# Como melhorar a qualidade dos softwares?

- Qualificação pelas associações de classe



- Qualificação por empresas de grande porte em projetos sensíveis
  - Área de petróleo, mineração, etc.

# Como melhorar a qualidade dos softwares?

- Regulamentação, através de órgãos públicos governamentais
  - CREAs, Ministérios, etc.
  - Exemplo dos EUA – Depto. Estradas da Califórnia
- Alcançar usuários qualificados
  - A qualidade técnica do software está diretamente ligada a qualidade técnica dos seus usuários

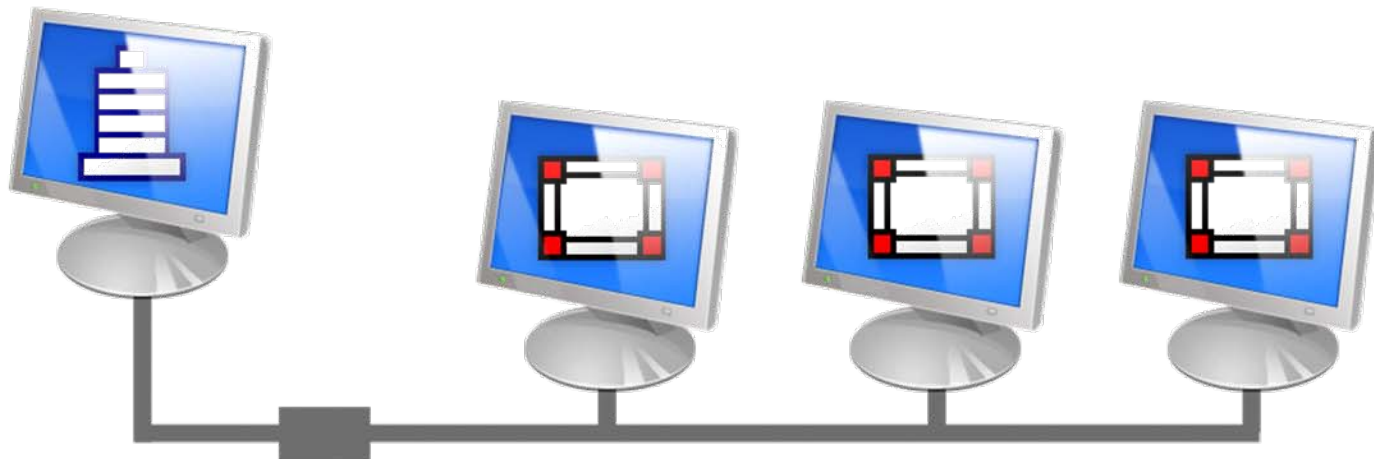
# 9 | Recomendações para Melhorar a Qualidade do Projeto Estrutural

# Como melhorar a qualidade do projeto com o emprego do software?

- Preocupação das empresas projetistas com Qualidade
- Supervisor de projeto – profissional experiente
- Selecionando um software de qualidade
  - Qualificação técnica dos desenvolvedores
  - Consulta aos usuários do mesmo ramo de atividade

# Como melhorar a qualidade do projeto com o emprego do software?

- Projetistas independentes lançando a mesma estrutura
- Utilização de dois softwares distintos para o mesmo projeto, envoltórias.



# Fim

Obrigado pela atenção