Pós-Graduação em Computação Distribuída e Ubíqua

INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software

[Software Visualization]
Stephan Diehl. Capítulos 1, 2, 3, 4 e 5.

Sandro S. Andrade
sandroandrade@ifba.edu.br

Objetivos

- Apresentar os conceitos fundamentais e objetivos da área de Visualização de Software
- Discutir exemplos e aplicações na área

Pós-Graduação em Computação Distribuída e Ubíqua

INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software - Fundamentos

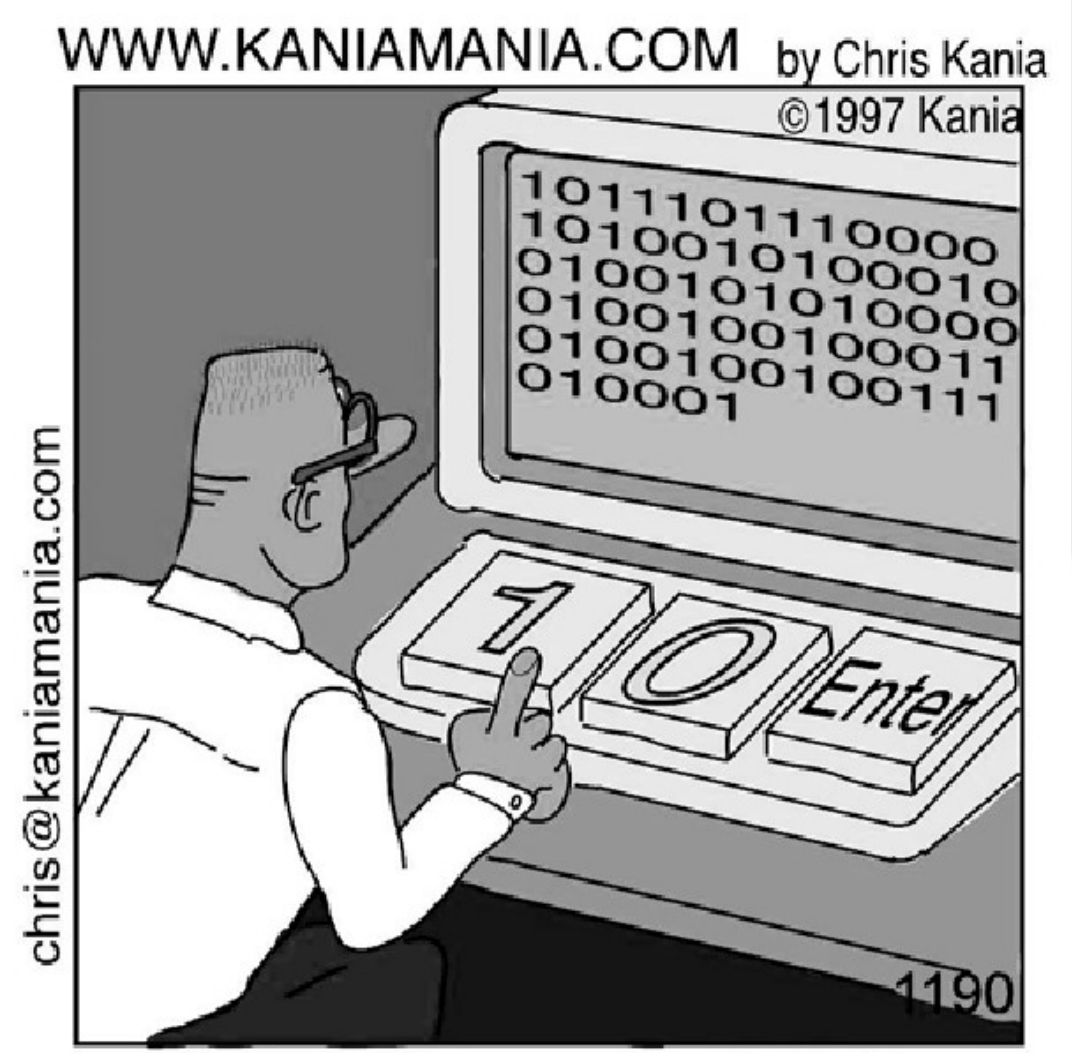
[Software Visualization]
Stephan Diehl. Capítulos 1, 2, 3, 4 e 5.

Sandro S. Andrade
sandroandrade@ifba.edu.br

Introdução

- Visualização é o processo de transformação de informação para um formato visual
- A exibição visual resultante possibilita que o cientista ou engenheiro perceba visualmente características implícitas nos dados, entretanto fundamentais para as atividades de análise e exploração
- Amplamente utilizada na Engenharia Mecânica, Química, Física e Medicina
- É, entretanto, ainda pouco utilizada como uma ferramenta para projeto, implementação e manutenção de *software*

Introdução



Real programmers code in binary.

Introdução

- Áreas:
  - Visualização Científica: processa dados físicos
  - Visualização de Informação: processos dados abstratos quaisquer

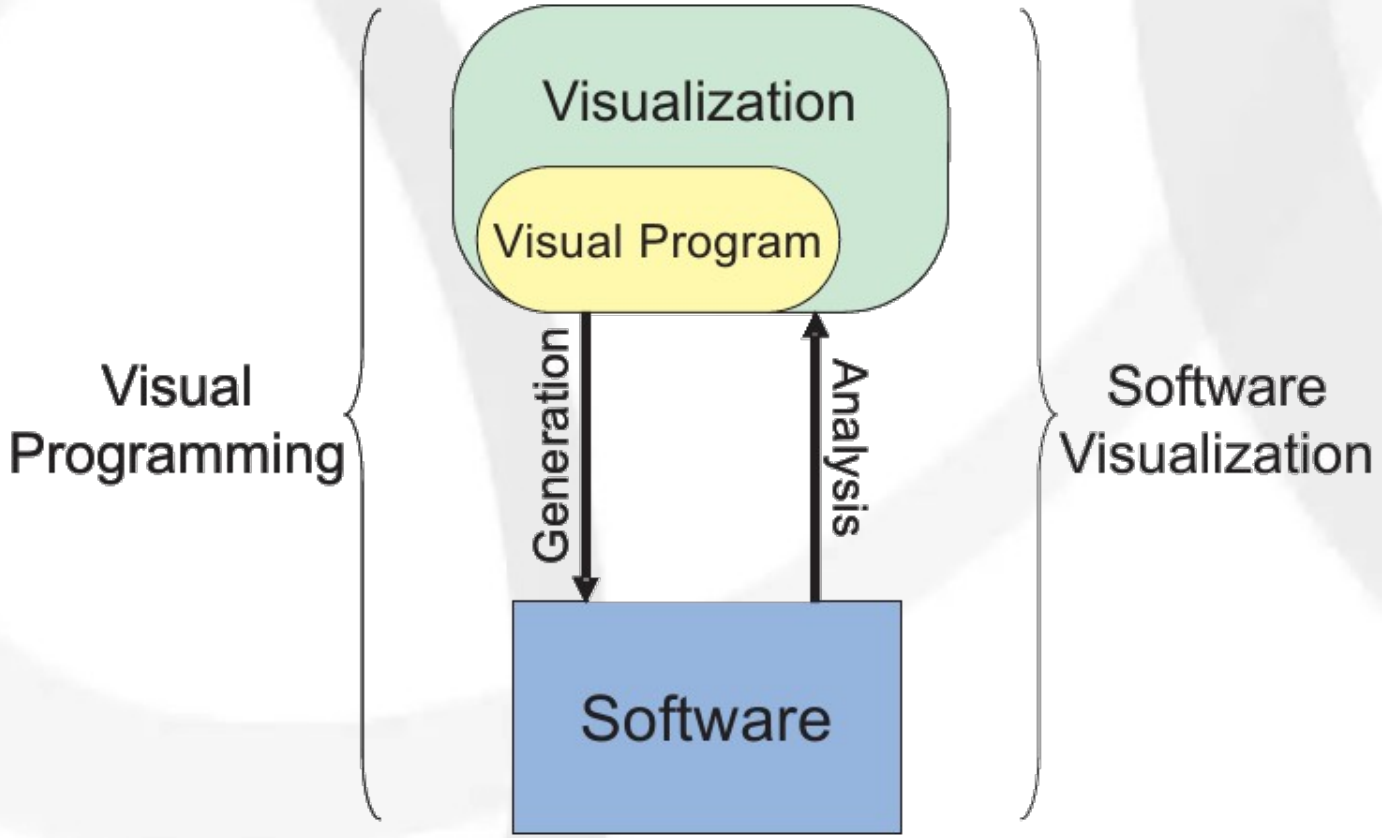
Visualização de Software é a visualização dos artefatos relacionados ao software e ao seu processo de desenvolvimento

Introdução

- Isto inclui a visualização de:
  - Estrutura:
    - Partes e relacionamentos estáticos do sistema, inferidas sem a necessidade de execução do sistema. Ex: grafo de chamadas estáticas e organização do *software* em módulos
  - Comportamento:
    - Refere-se à execução do programa, utilizando dados reais ou abstratos. É uma sequência de estados, onde cada estado contém o código e dados atuais do sistema
  - Evolução:
    - Refere-se ao processo de desenvolvimento e, em particular, enfatiza o fato de que o sistema muda com o tempo, com o objetivo de corrigir *bugs* ou adicionar novas funcionalidades

Introdução

- Visualização de Software x Programação Visual:

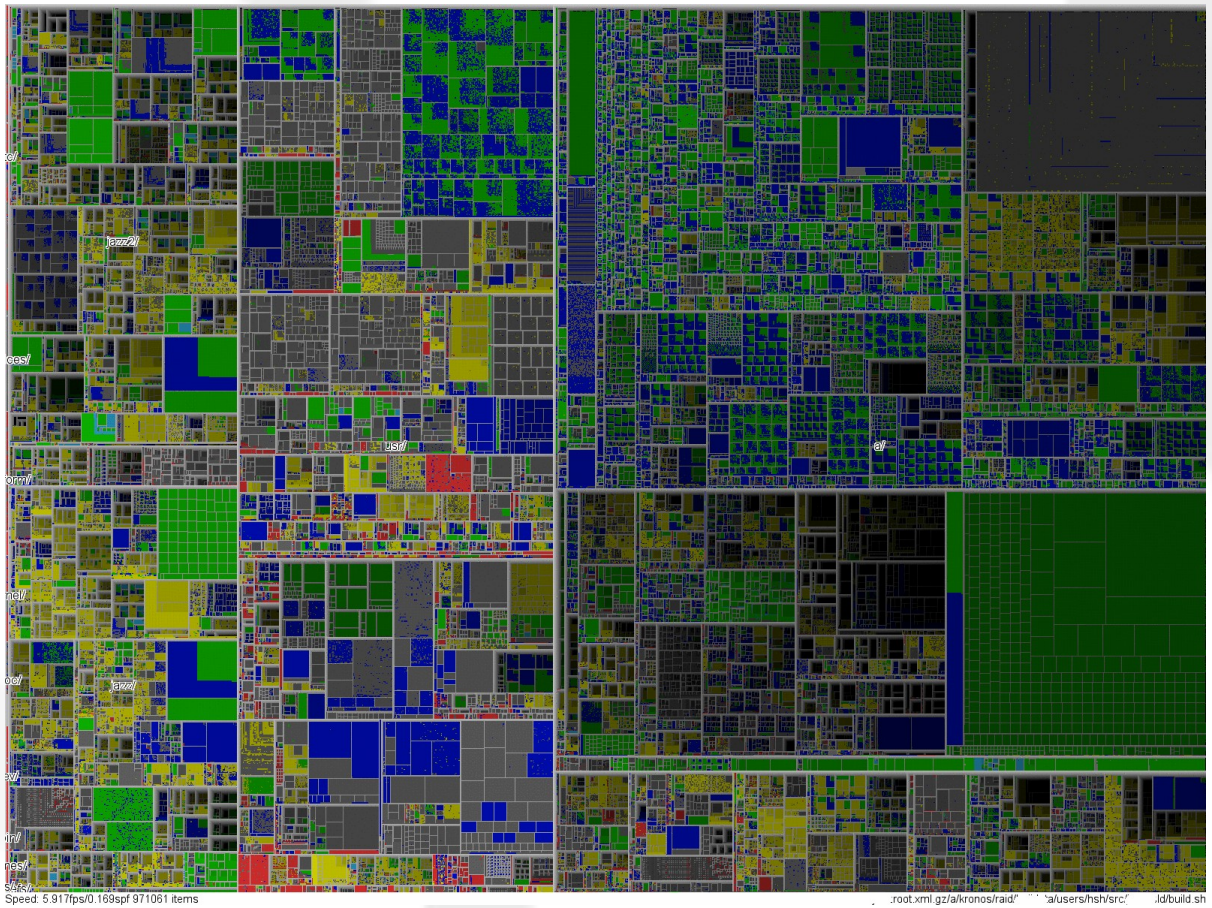


Fundamentos de Visualização

- Técnicas gerais para Visualização de Informação:
  - *Interactive Exploration*:
    - Permite que o usuário primeiro obtenha um *overview* dos dados e depois, através de operações do *zoom* e *filter* obtenha os detalhes sob demanda
  - *Focus + Context*:
    - Uma visão detalhada de alguma parte da informação (*focus*) é inserida dentro da visualização do contexto (informações menos detalhadas sobre partes relacionadas ao foco)

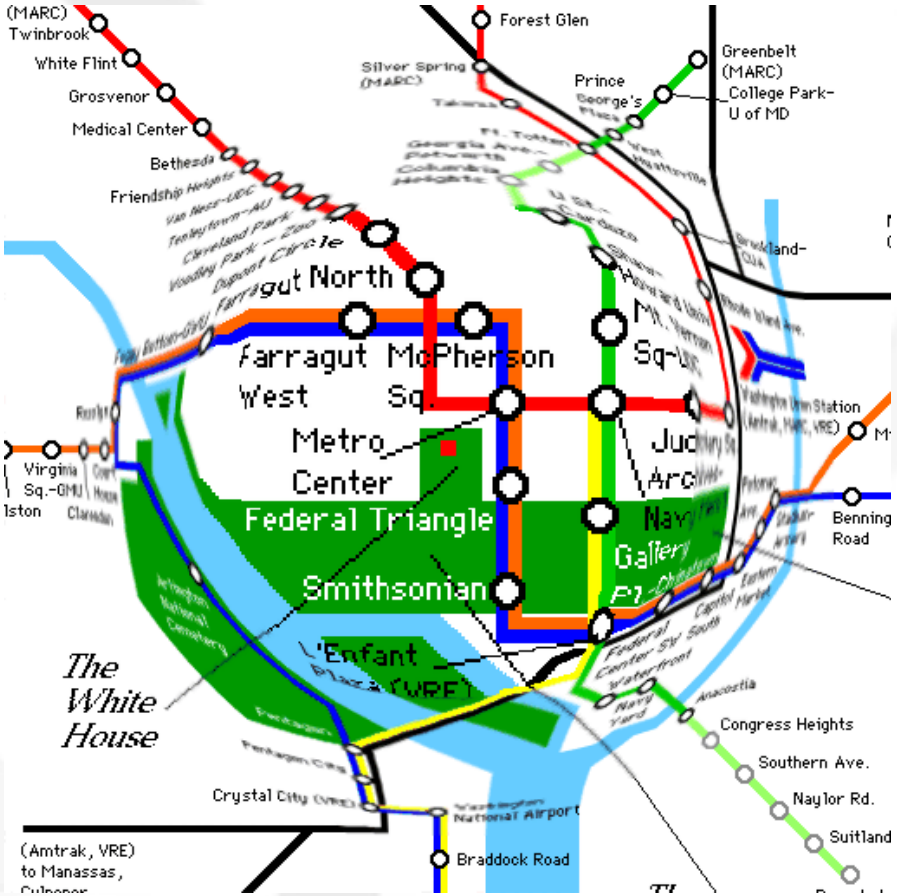
Fundamentos de Visualização

- *Interactive Exploration*:



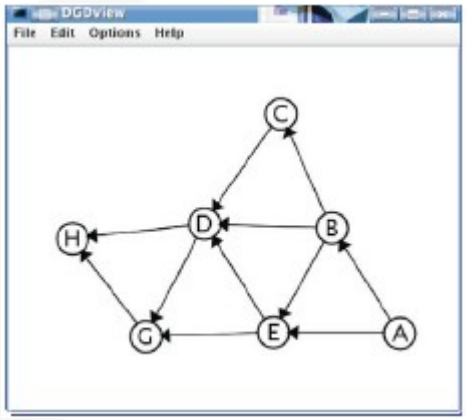
Fundamentos de Visualização

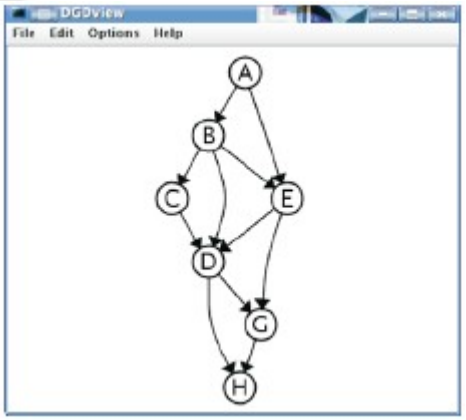
- *Focus + Context*:

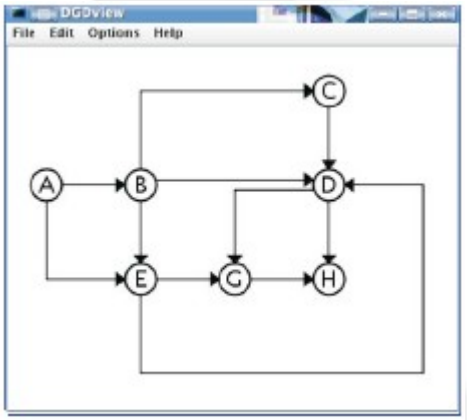


Fundamentos de Visualização

- Grafos e algoritmos de *layout*:



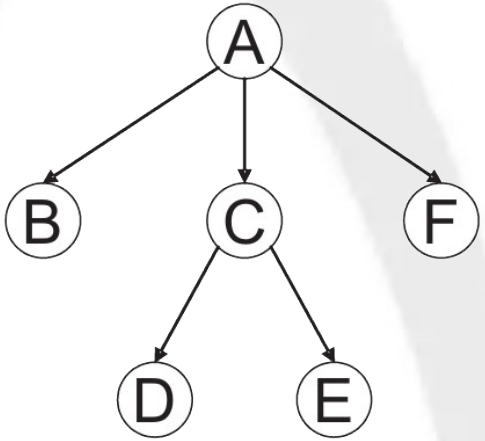


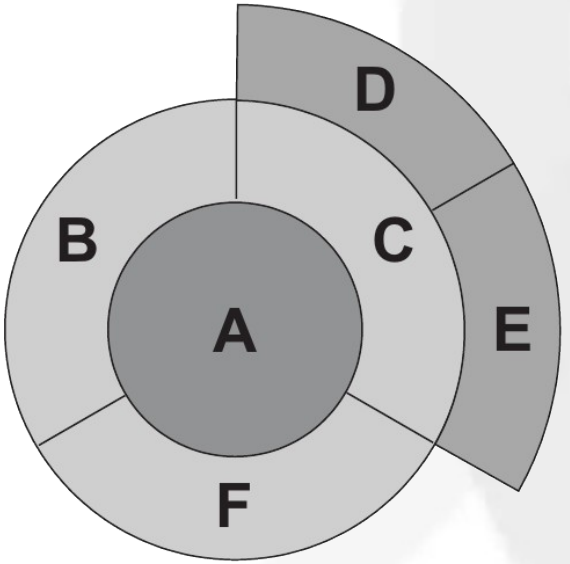


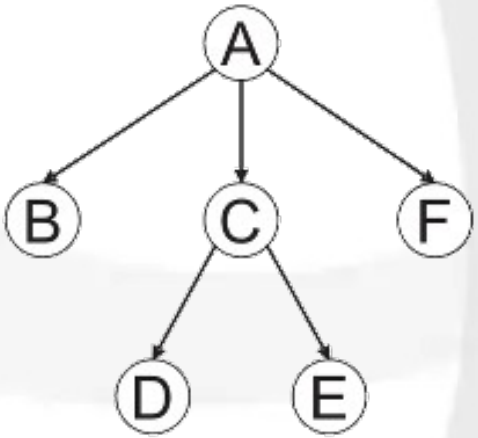
- Soluções:
  - Graphviz, Boost Graph Library

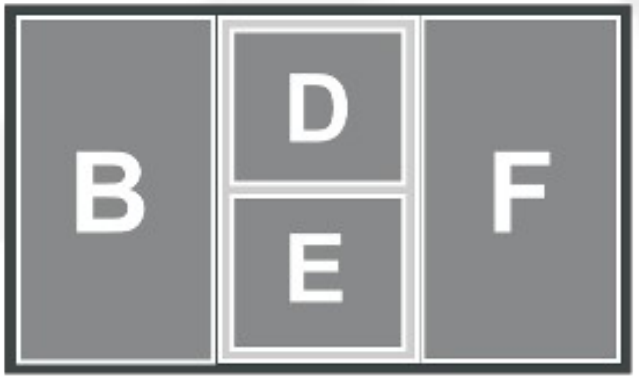
Fundamentos de Visualização

- Hierarquias:



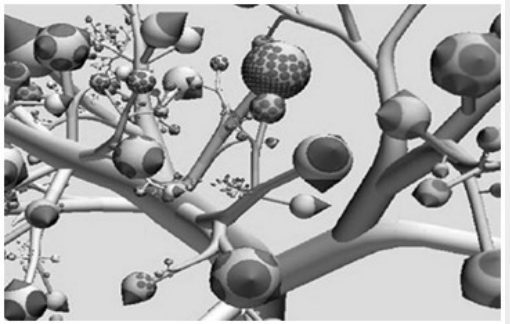


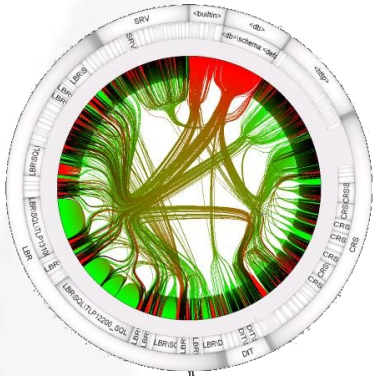


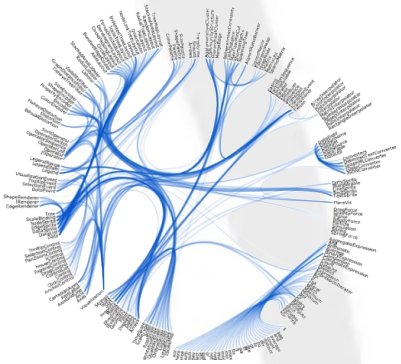


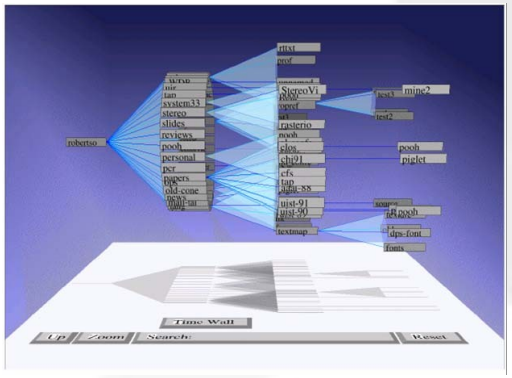
Fundamentos de Visualização

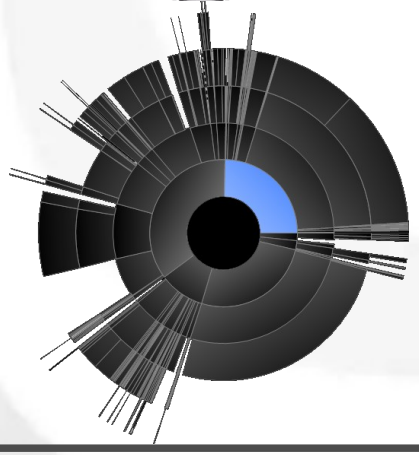
- Hierarquias:

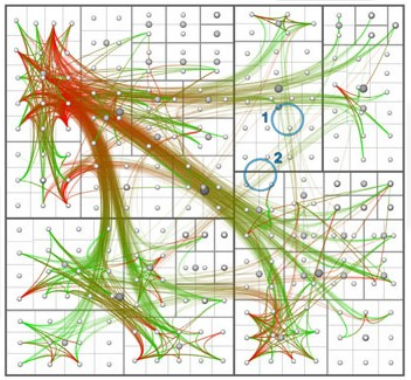


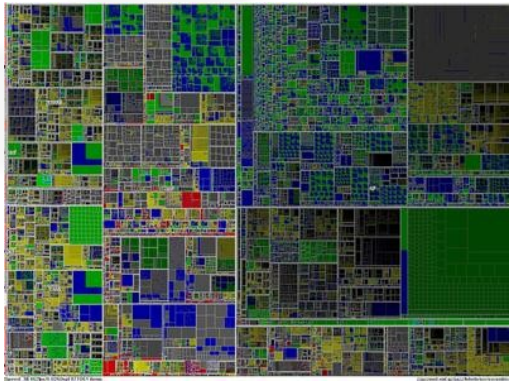












Pós-Graduação em Computação Distribuída e Ubíqua

INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software - Visualização Estrutural

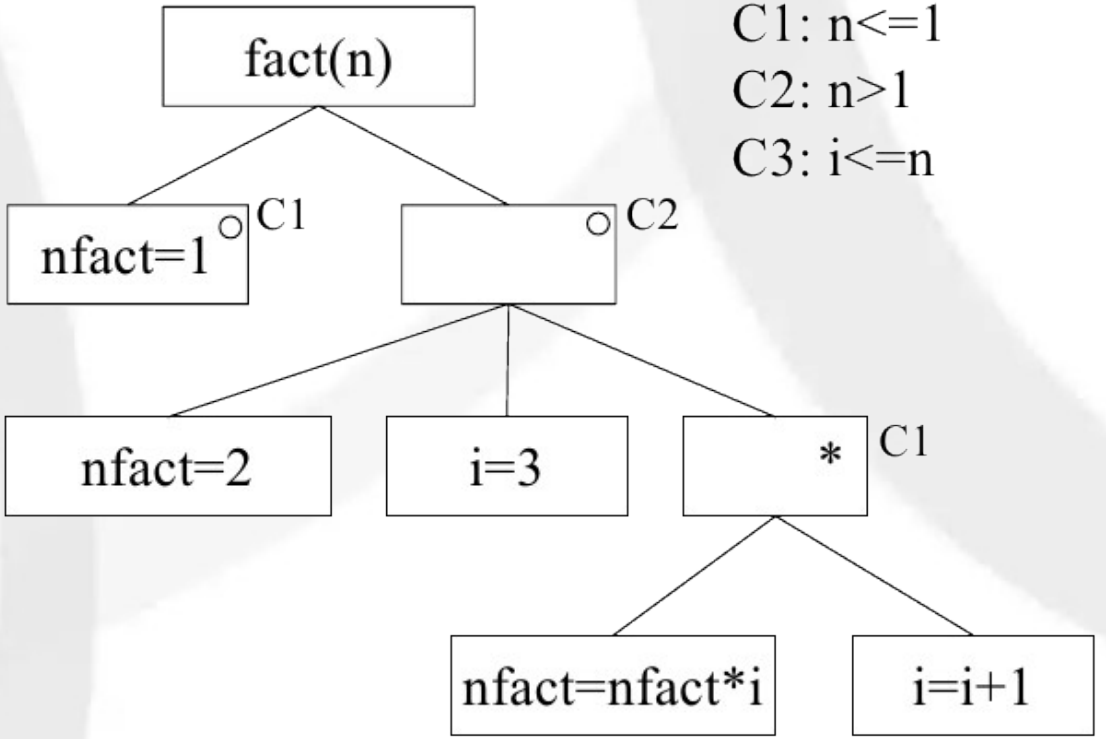
[Software Visualization]
Stephan Diehl. Capítulos 1, 2, 3, 4 e 5.

Sandro S. Andrade
sandroandrade@ifba.edu.br

Visualização Estrutural

- Representações diagramáticas:
  - Diagramas de Jackson

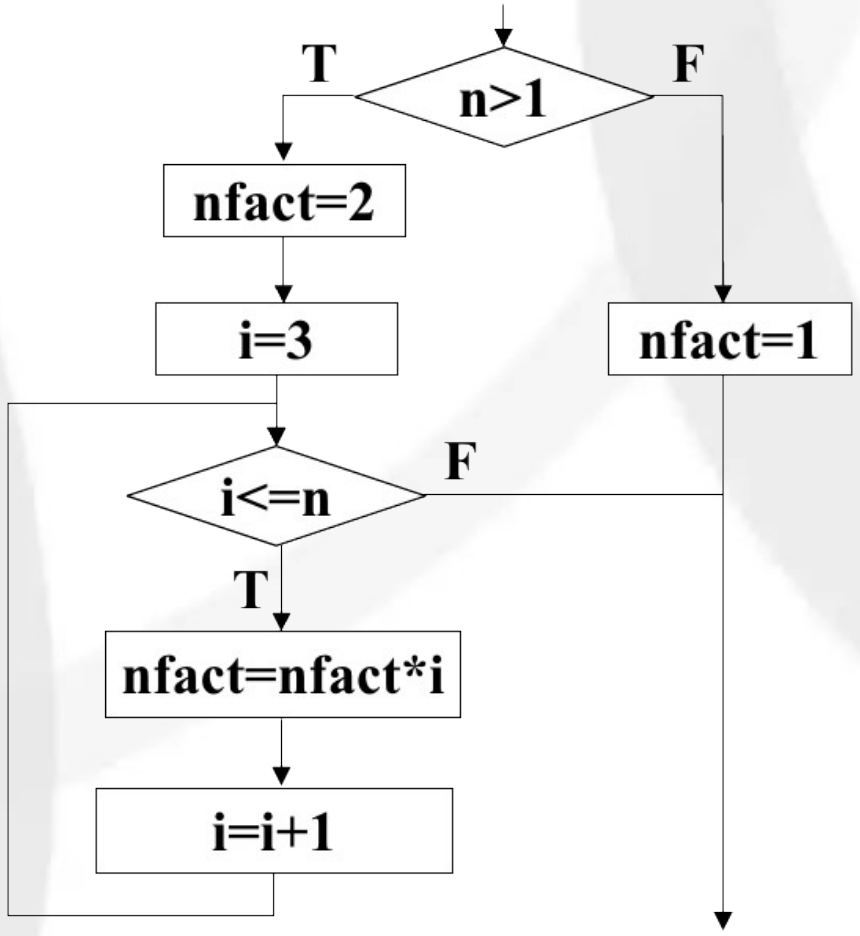
```
int fact(n) {
  if (n>1)
   { nfact=2;
     for(int i=3;i<=n;i++)
       nfact=nfact*i;
   }
  else
   { nfact=1;
   }
  return nfact;
}
```



Visualização Estrutural

- Representações diagramáticas:
  - *Control Flow Graphs*

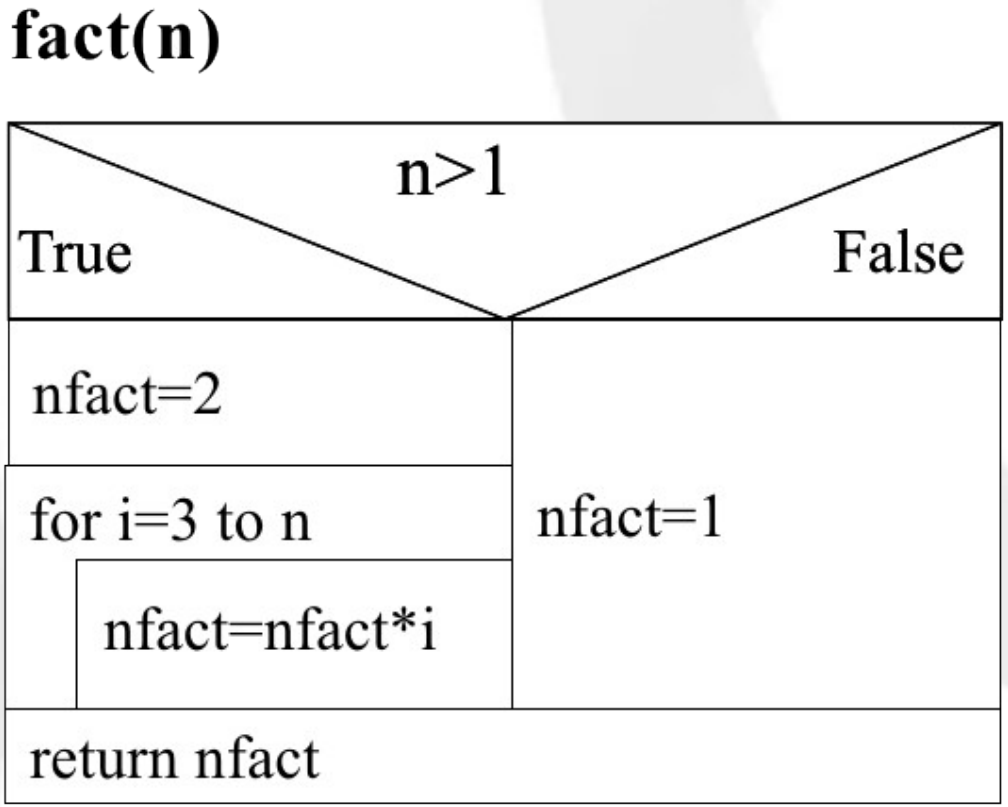
```
int fact(n) { if (n>1)
   { nfact=2;
     int i=3;
     while(i<=n)
      { nfact=nfact*i;
        i=i+1;
      }
   }
  else
   { nfact=1;
   }
  return nfact;
}
```



Visualização Estrutural

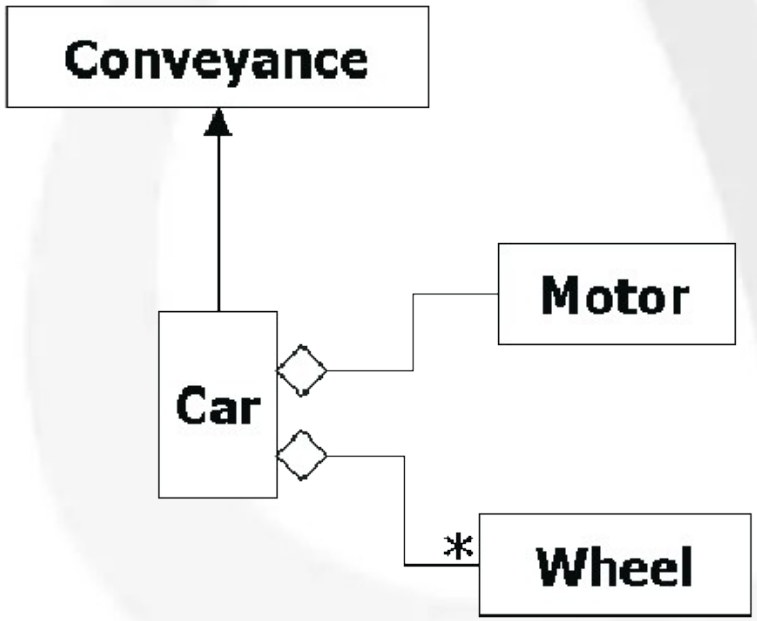
- Representações diagramáticas:
  - Diagramas de Nassi-Schneiderman

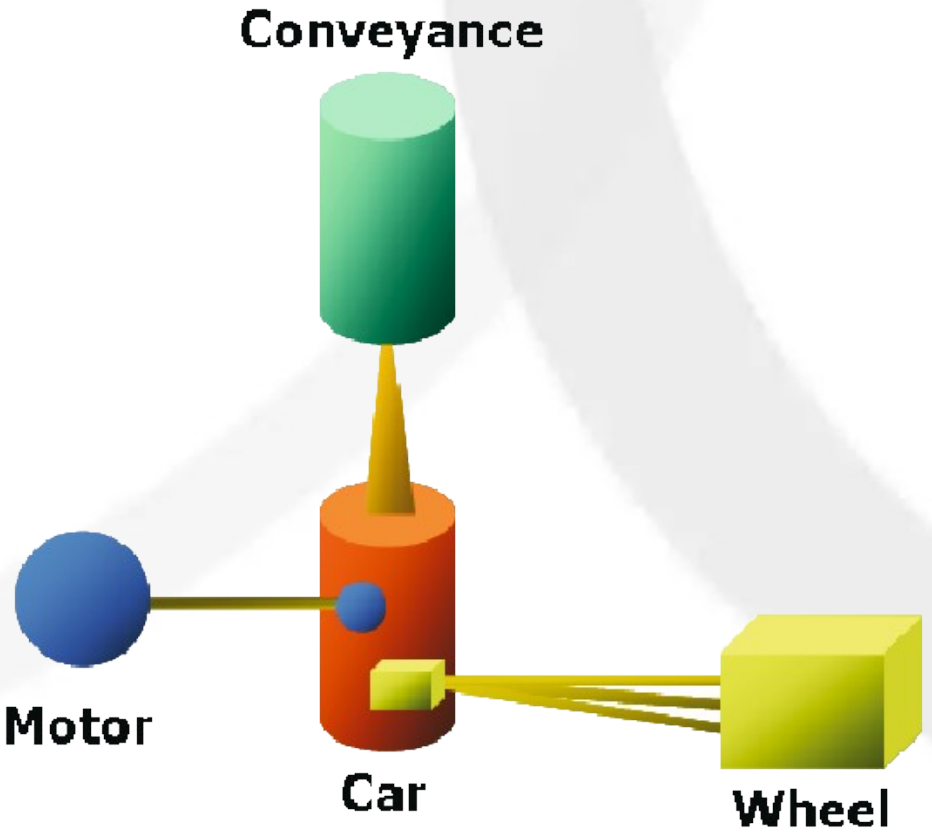
```
int fact(n) { if (n>1)
   { nfact=2;
     for(int i=3;i<=n;i++)
       nfact=nfact*i;
   }
  else
   { nfact=1;
   }
  return nfact;
}
```



Visualização Estrutural

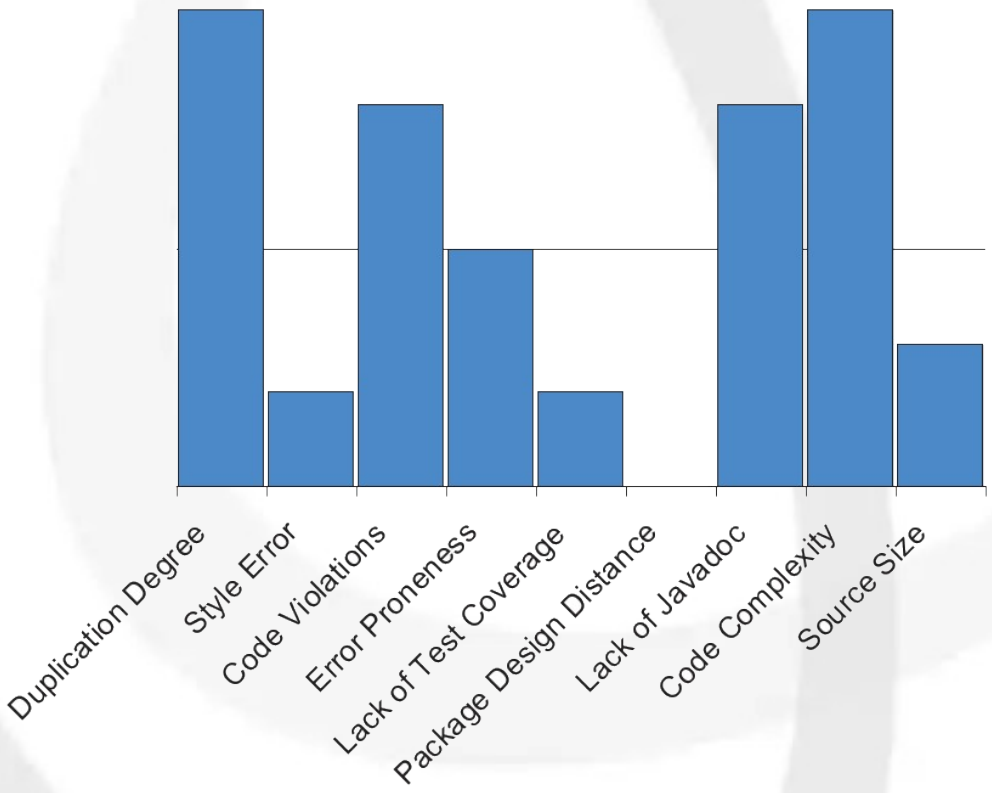
- Representações diagramáticas:
  - UML x *Geon Diagrams*

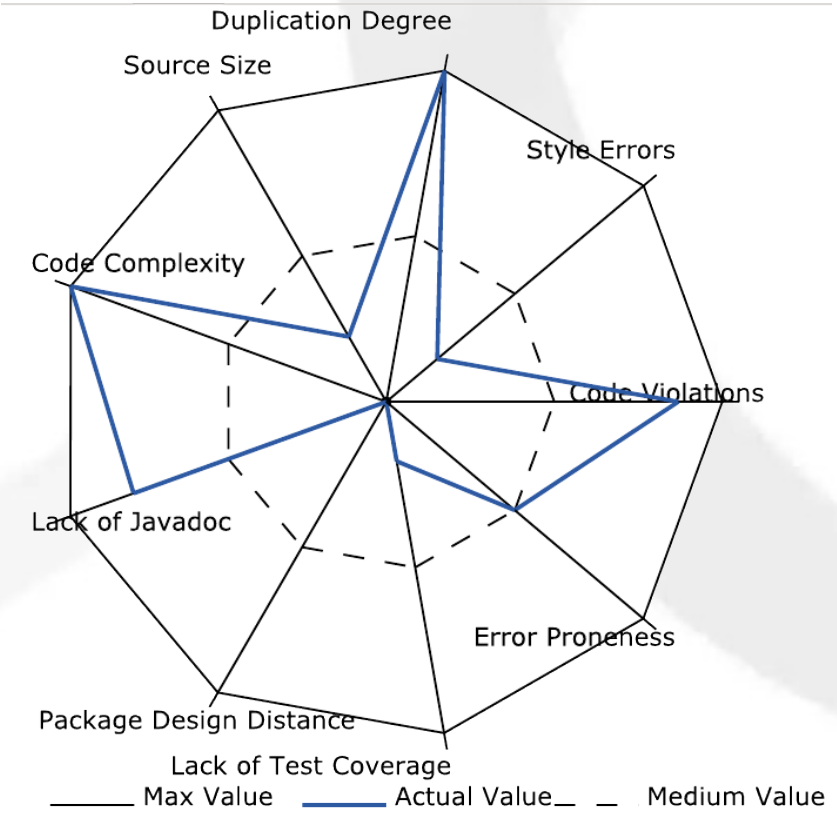




Visualização Estrutural

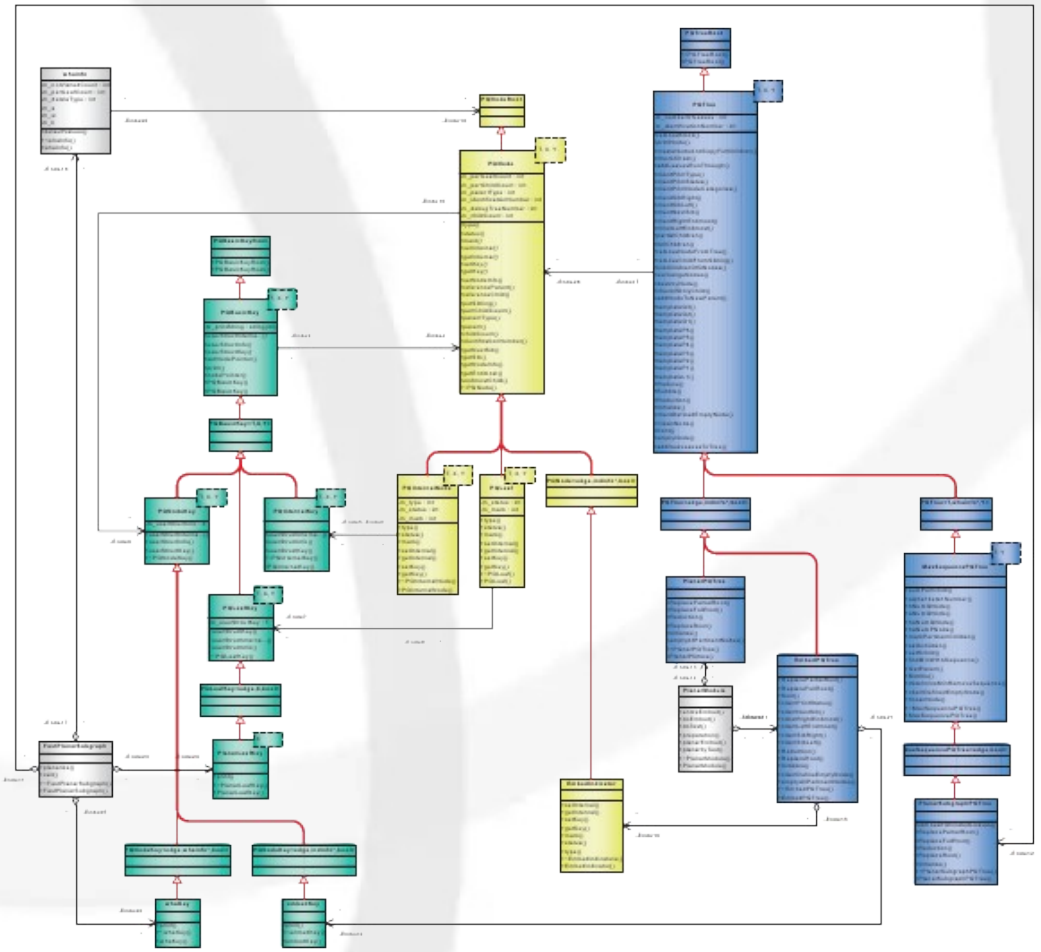
- Representações diagramáticas:
  - Visualizando métricas





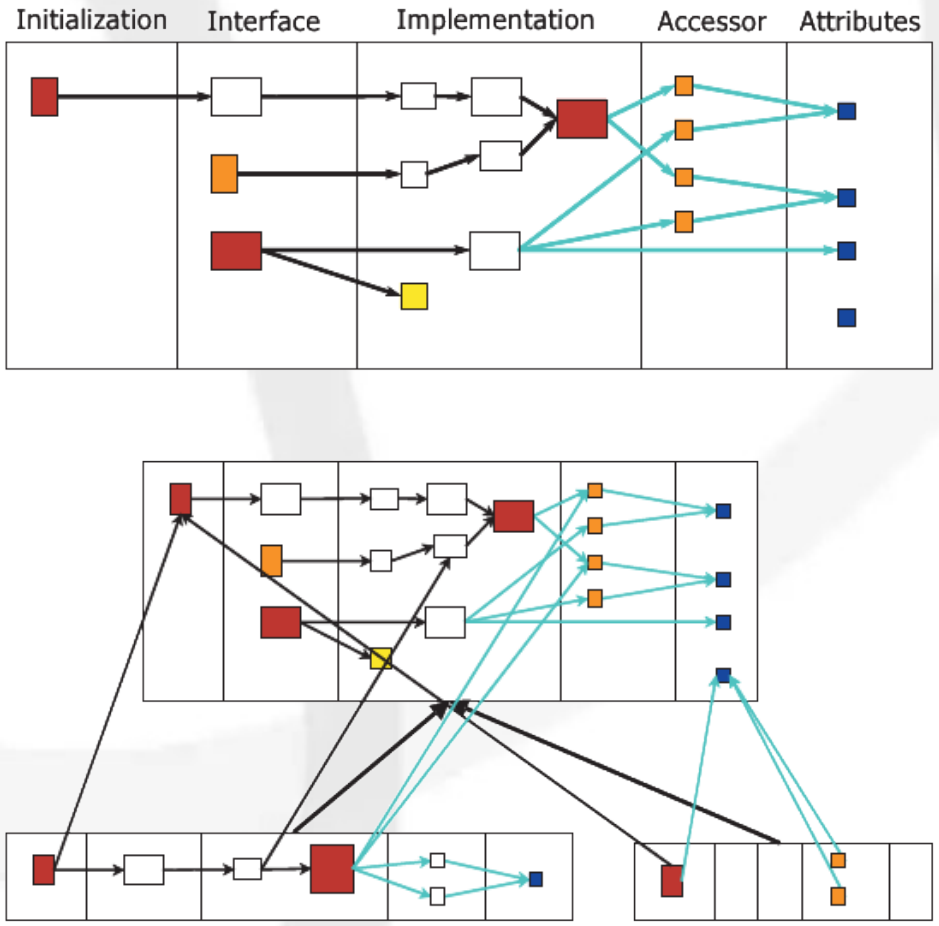
Visualização Estrutural

- Visualização e Engenharia Reversa:



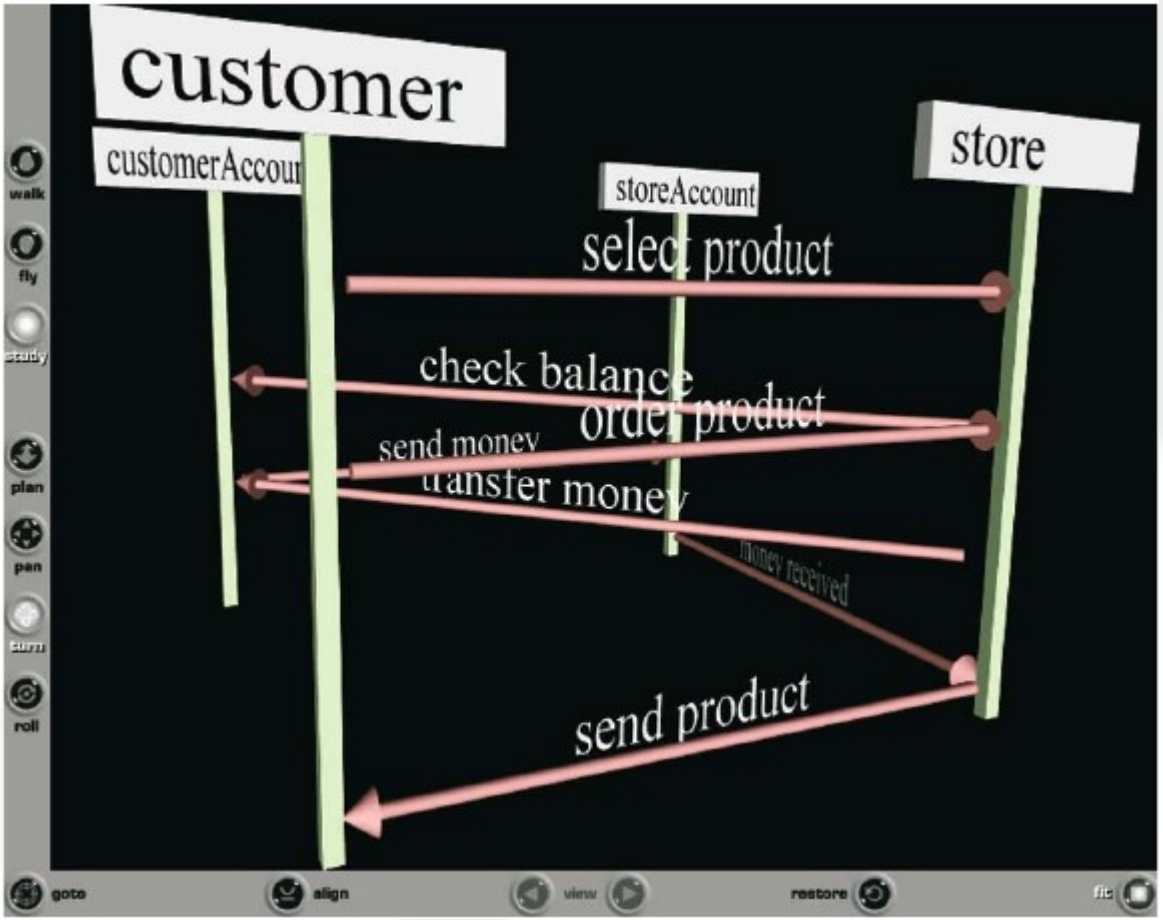
Visualização Estrutural

- Visualização e Engenharia Reversa:



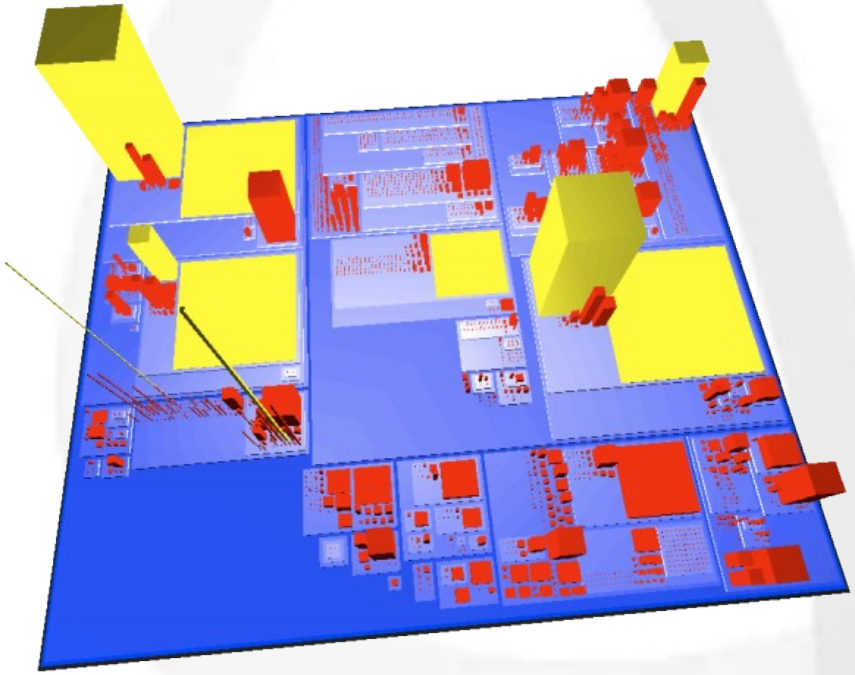
Visualização Estrutural

- Arquitetura de Software e Visualização 3D:

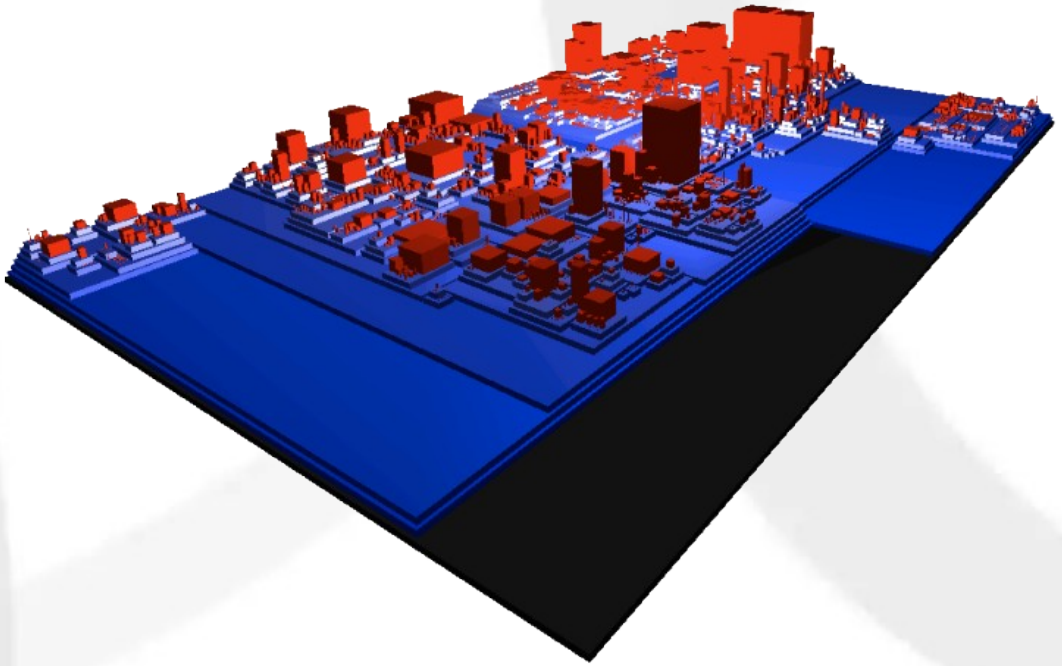


Visualização Estrutural

- Arquitetura de Software e Visualização 3D:



ArgoUML 0.24



Azureus 2.2.0.2

Pós-Graduação em Computação Distribuída e Ubíqua

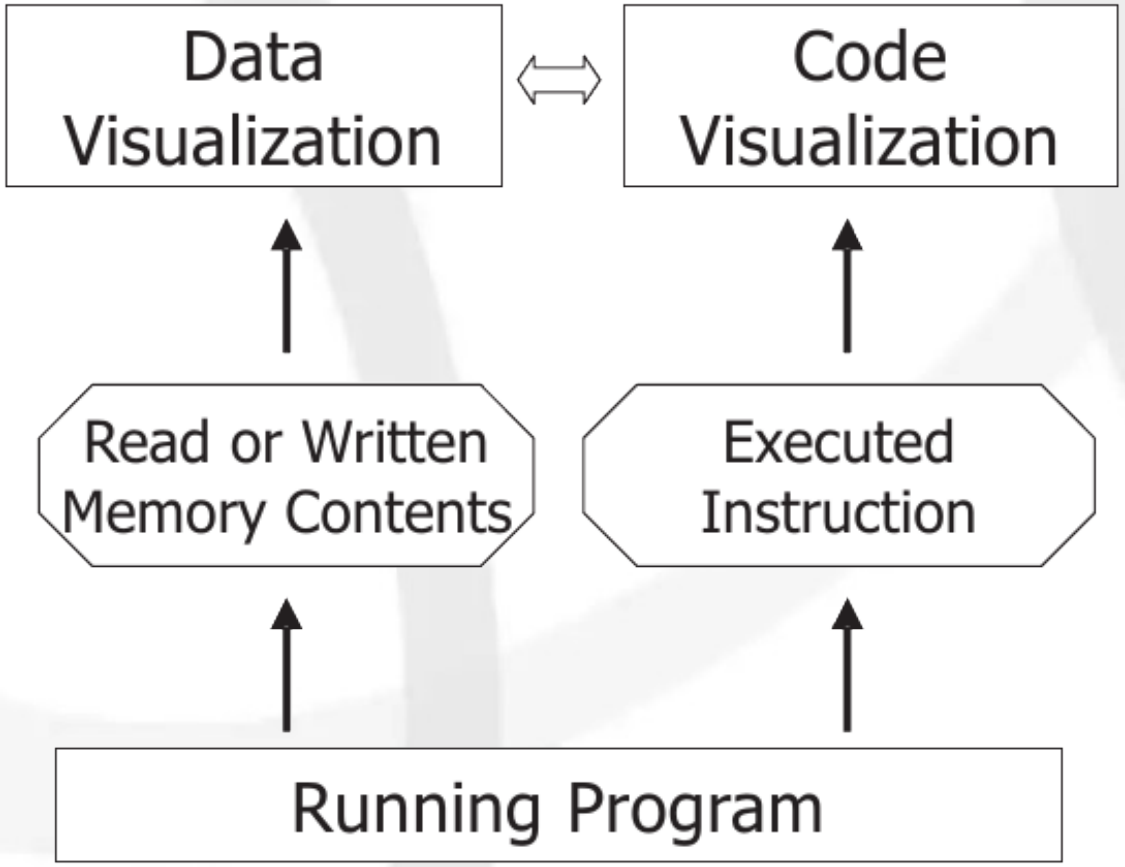
INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software - Visualização Comportamental

[Software Visualization]
Stephan Diehl. Capítulos 1, 2, 3, 4 e 5.

Sandro S. Andrade
sandroandrade@ifba.edu.br

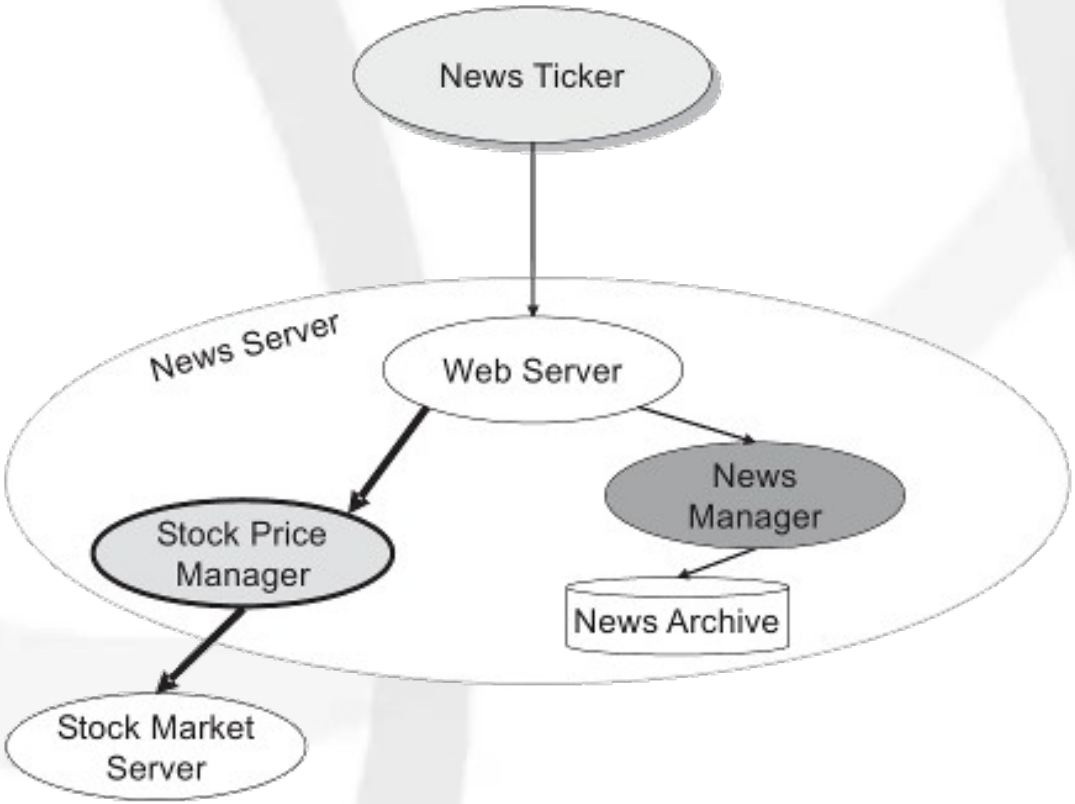
Visualização Comportamental

- Visão geral:



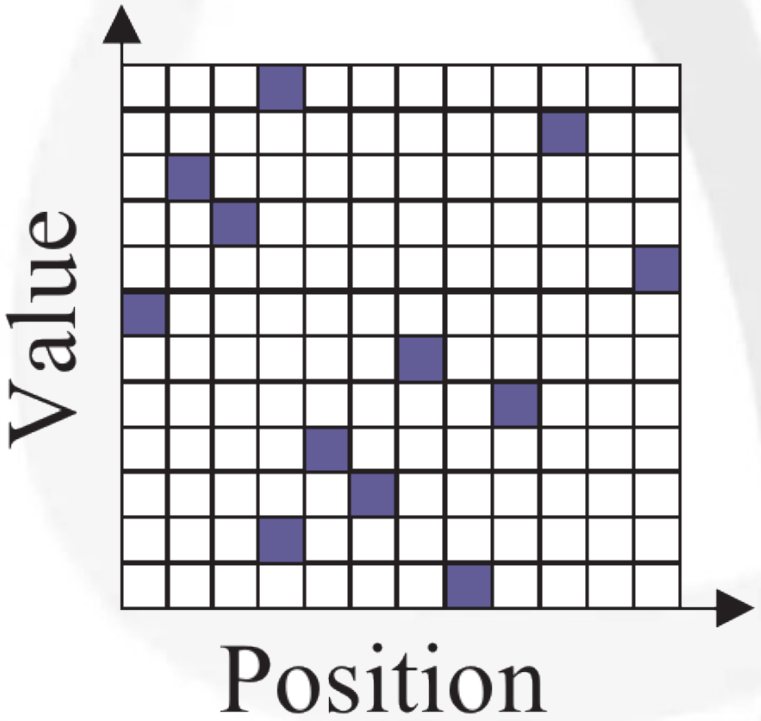
Visualização Comportamental

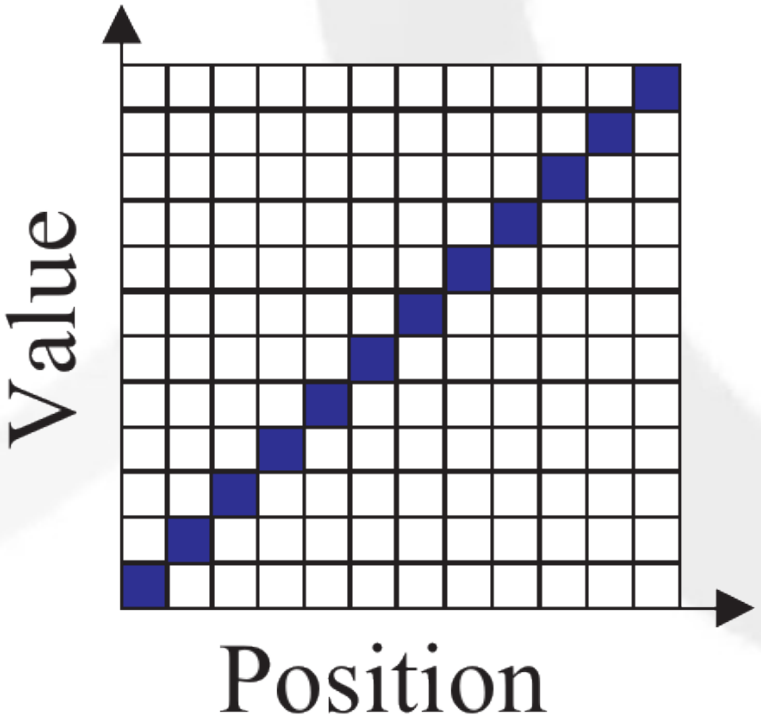
- Métodos:
  - Diagramas estáticos aumentados:



Visualização Comportamental

- Métodos:
  - Animação de algoritmos (não só para propósitos didáticos):



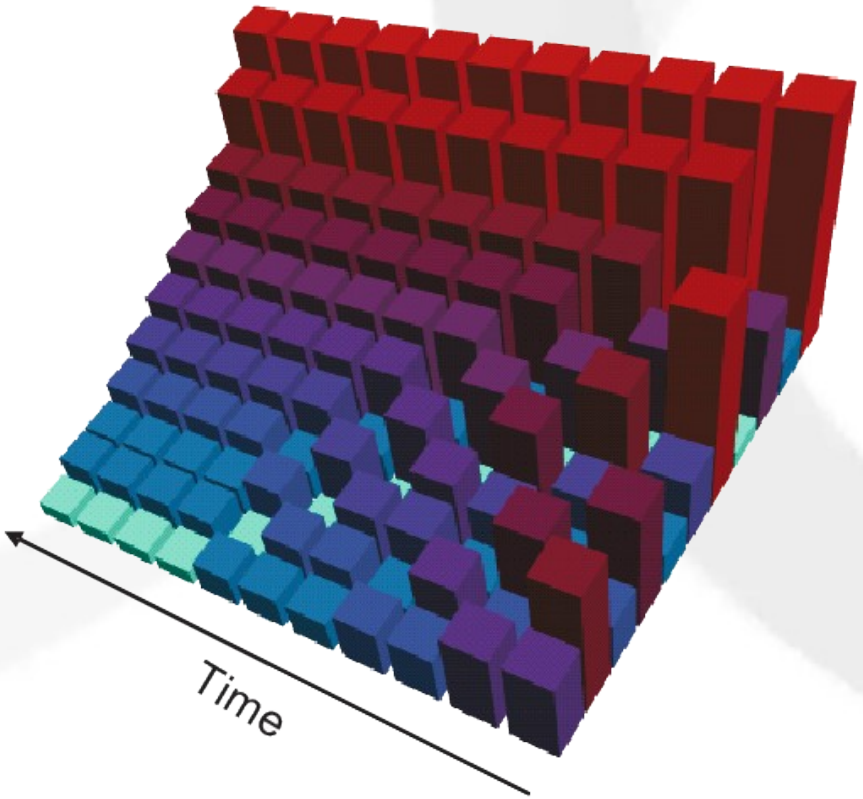


Visualização Comportamental

- Métodos:
  - Animação 3D do *Bubble Sort*

8 3 7 2 4 11 1 6 2 13
3 7 2 4 8 1 6 2 11 13
5 2 4 7 1 6 2 8 11 13
2 4 5 1 6 2 7 8 11 13
3 4 1 5 2 6 7 8 11 13
3 1 4 2 5 6 7 8 11 13
1 3 2 4 5 6 7 8 11 13
2 2 3 4 5 6 7 8 11 13
2 2 3 4 5 6 7 8 11 13
2 2 3 4 5 6 7 8 11 13
2 2 3 4 5 6 7 8 11 13

Time



Pós-Graduação em Computação Distribuída e Ubíqua

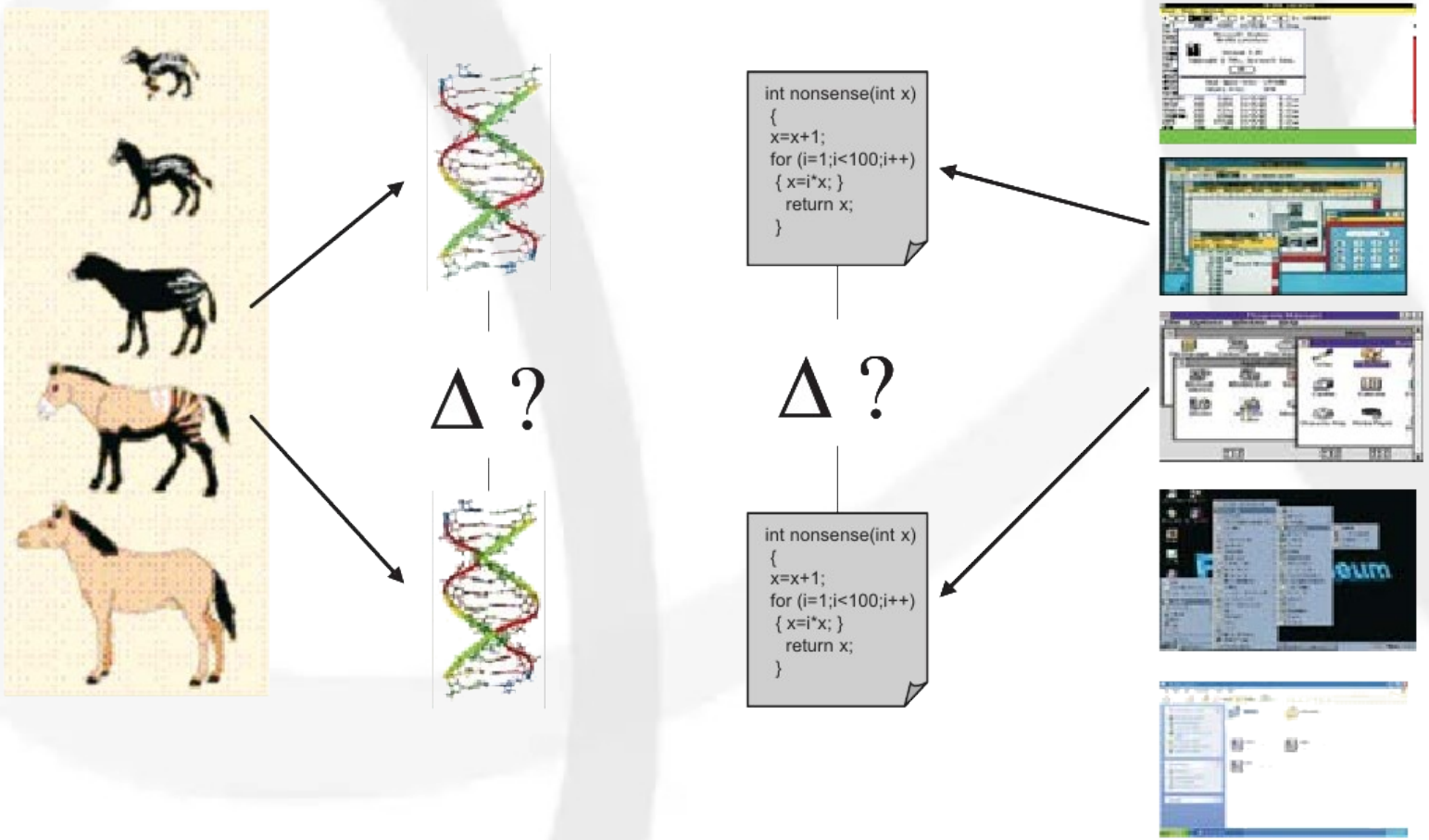
INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software - Visualização de Evolução

[Software Visualization]
Stephan Diehl. Capítulos 1, 2, 3, 4 e 5.

Sandro S. Andrade
sandroandrade@ifba.edu.br

Visualização de Evolução

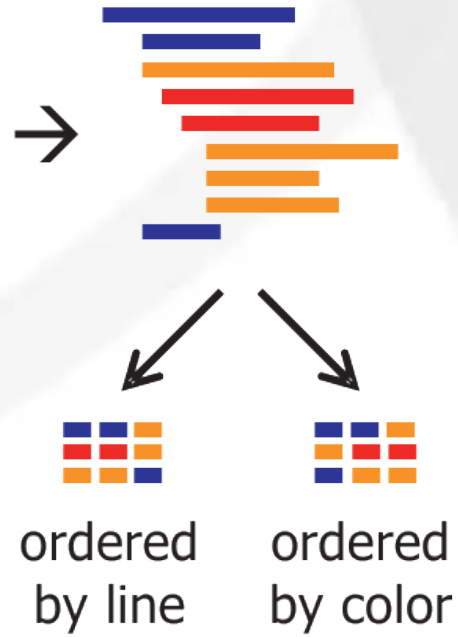
- Visão geral:



Visualização de Evolução

- Exemplos:

```
void bubblesort(int a[]) {
  int len=a.length;
  for(int i=0; i<len; i++) {
    for(int j=0; j<len; j++) {
      if ( a[j]>a[j+1] ) {
        int tmp = a[j]; // swap
        a[j] = a[j+1];
        a[j+1] = tmp; } } }
  return; }
```

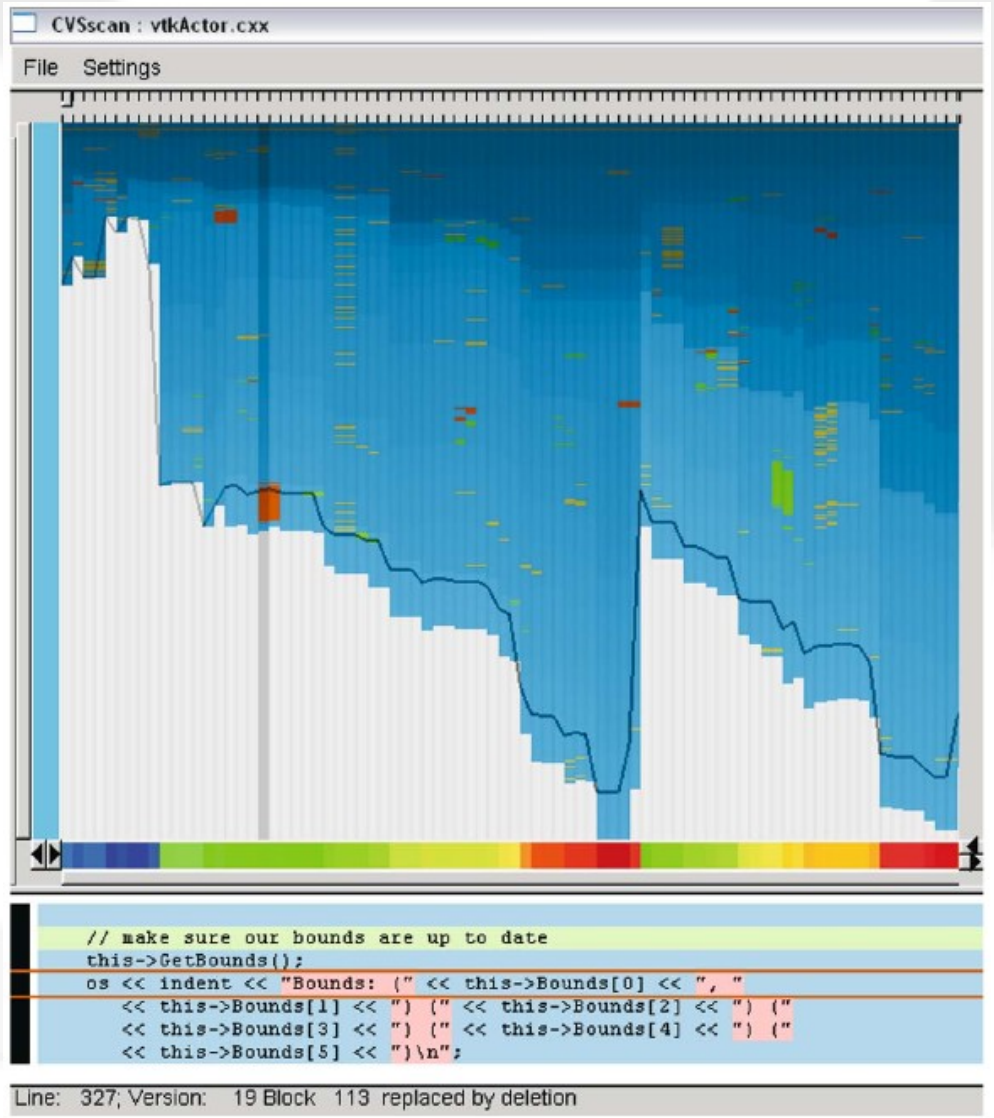


Visualização de Evolução

- Exemplos:

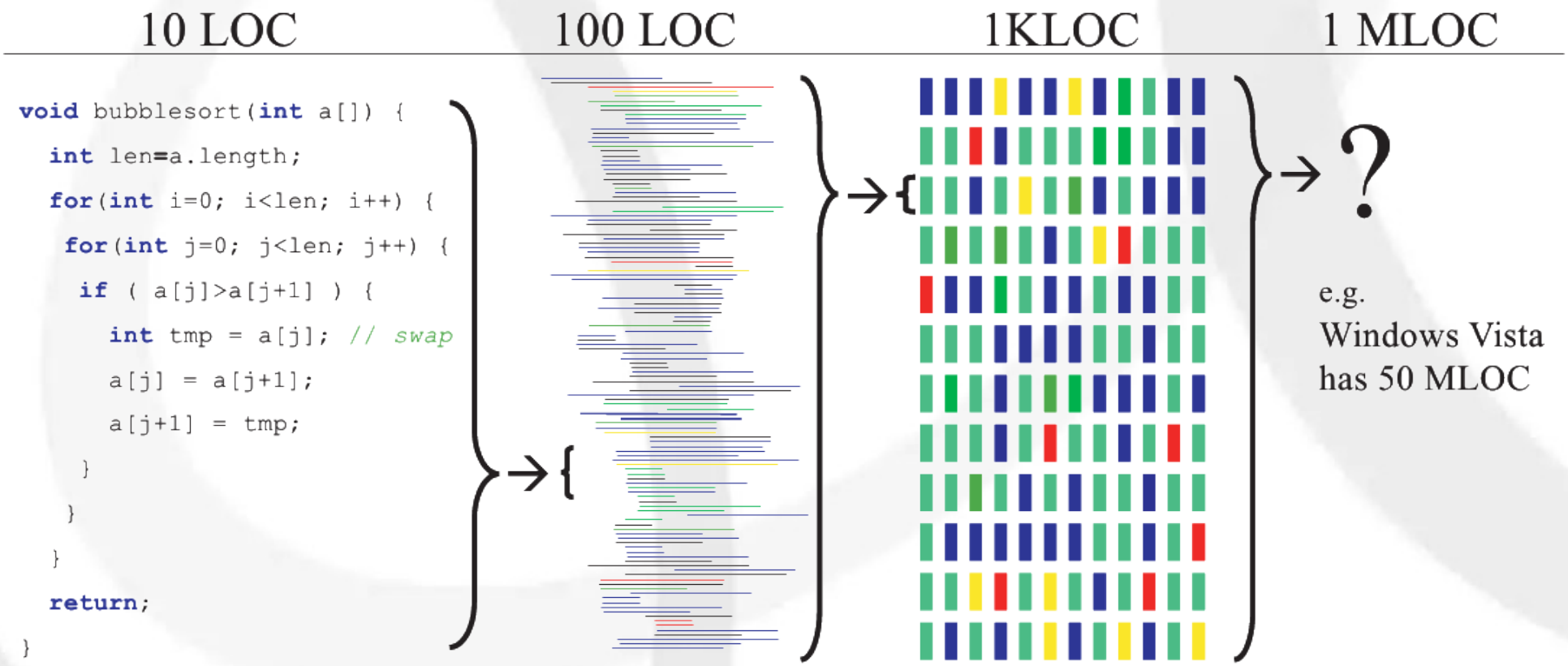
Visualização de Evolução

- Exemplos:



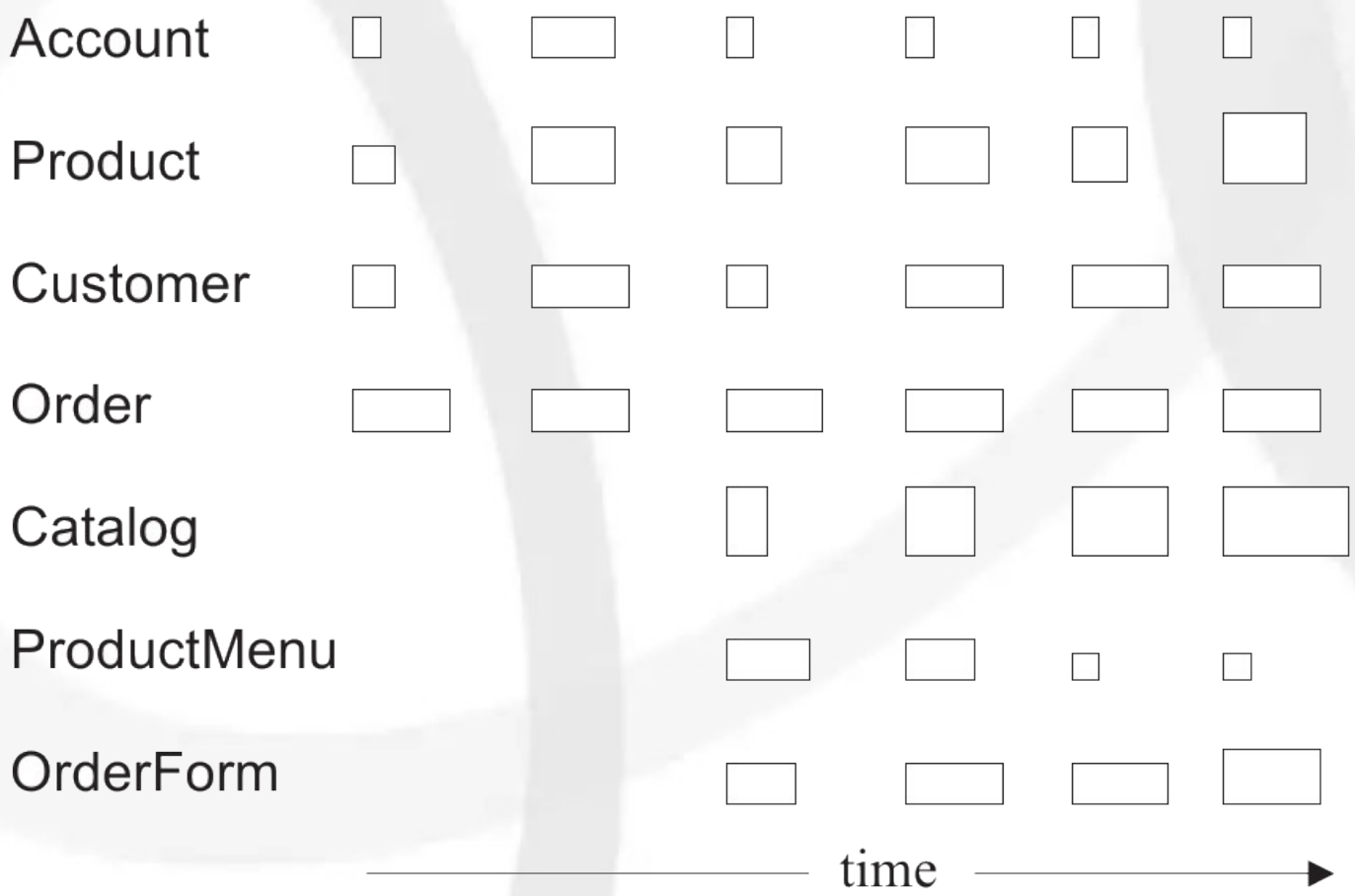
Visualização de Evolução

- Desafios:



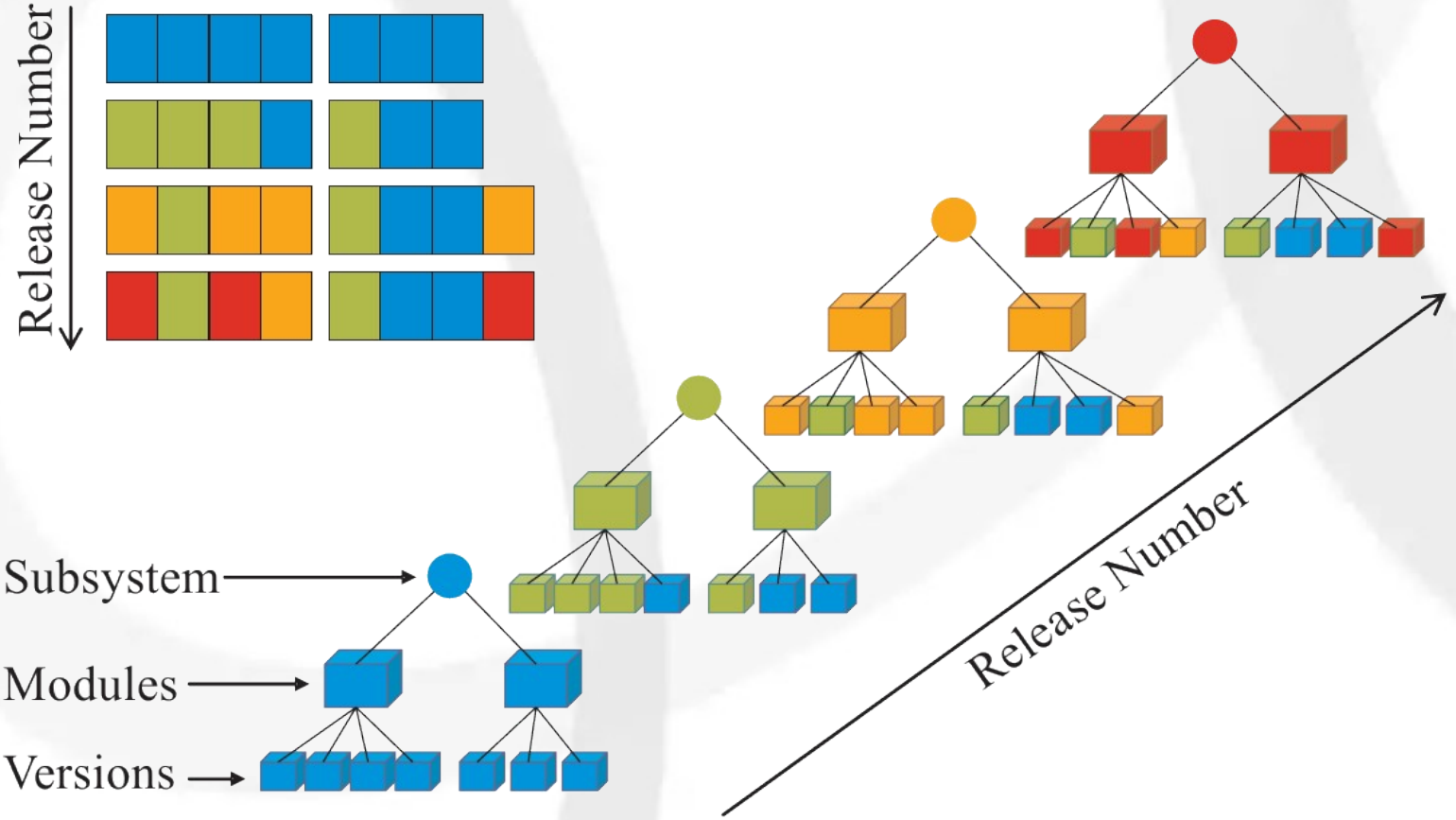
Visualização de Evolução

- Matriz de Evolução:



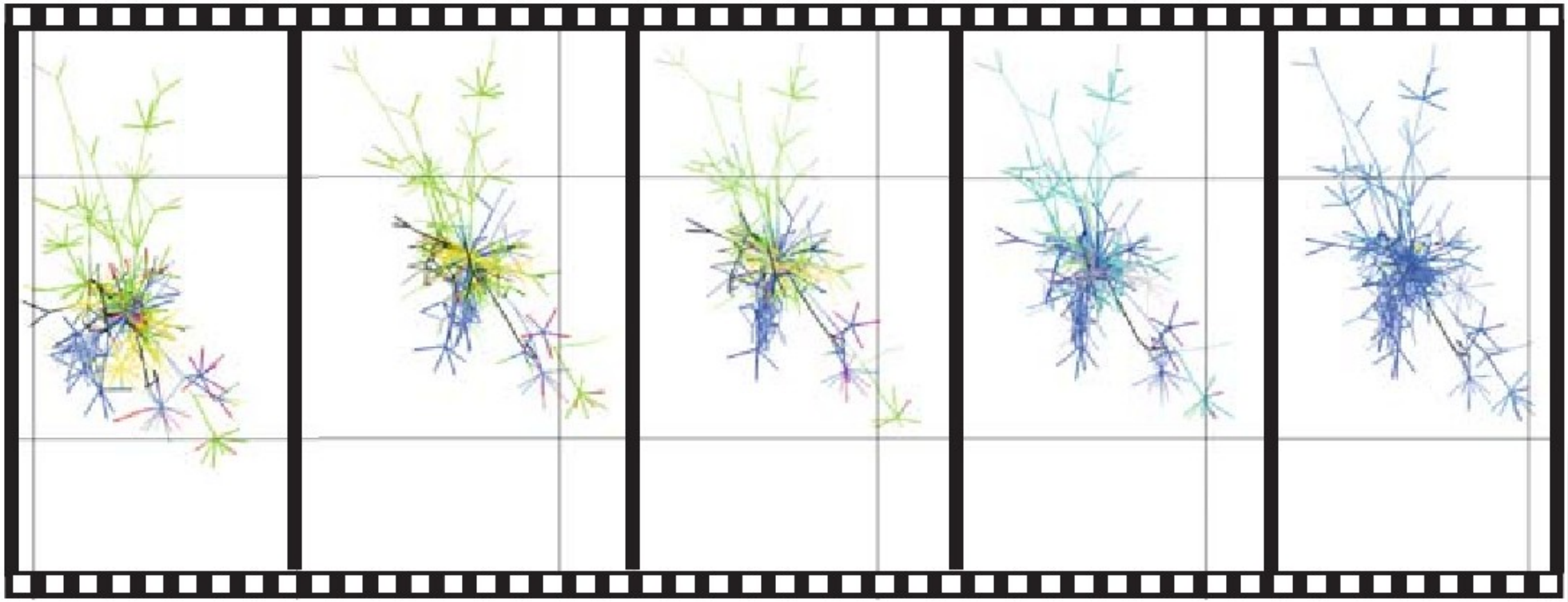
Visualização de Evolução

- Visualizando mudanças estruturais:



Visualização de Evolução

- Visualizando mudanças em árvores de herança:



Conclusões

- Visualização de Software como tecnologia importante de auxílio a todas as fases de desenvolvimento
- Fundamental para o desenvolvimento e evolução de sistemas complexos
- É desejável que esteja integrada à IDE sendo utilizada
- Novos paradigmas de visualização continuam surgindo

Pós-Graduação em Computação Distribuída e Ubíqua

INF612 - Aspectos Avançados em Engenharia de Software
Visualização de Software

Sandro S. Andrade
sandroandrade@ifba.edu.br