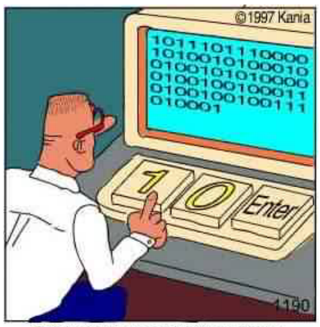
# Software Visualization



Real programmers code in binary.

## Lecture
## WS 02/03

Introduction

## Introduction

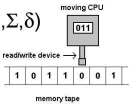
- What is „Software Visualization"?
- visual metaphors
- application domains
- relevance for software engineering
- taxonomies
- visualization pipeline

## Visualization

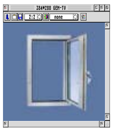
- „The senses cannot think. The understanding cannot see. By their union only can knowledge be produced."
  *[Emanuel Kant]*
- „Intelligence Amplification"
  *[Frederick Brooks ]*

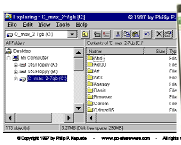
## Metaphors

$M=(Q,q_0,F,\Sigma,\delta)$



- Automata, Machines, Tapes
- Trees, Leaves, Queues
- Files, Folder, Archive
- Window




Exploring - C_max_2-7gb (C:)
© 1997 by Philip P.
File Edit View Tools Help
C_max_2-7gb (C:)
All Folders
Contents of 'C_max_2-7gb (C:)'
Desktop
My Computer
3½ Floppy (A:)
5¼ Floppy (B:)
C_max_2-7gb (C:)
Name
Size
Type
Andi
Aol30
Art
Art3
Aseasy
Basic
Browser
Cdrom
Cdrom95
113 object(s)
3.27MB (Disk free space: 290MB)
© Copyright 1997 by Philip P. Kapusta - www.pc-shareware.com - All rights reserved

## Software Visualization

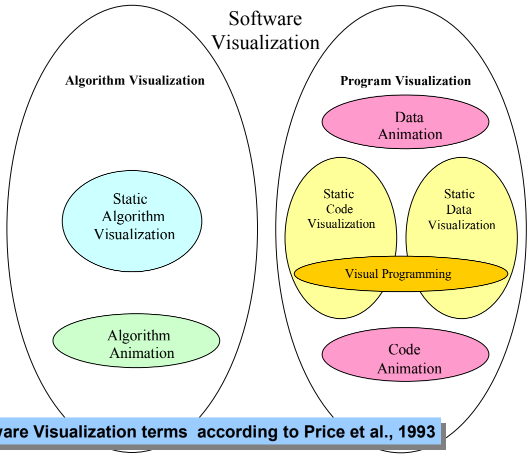
- Narrow Definition
  - **Visualization of algorithms and programs.**
- Wide Definition
  - **Visualization of artefacts related to software and its development process.**
- Short
  - **Visualizing Structure, Behavior and Evolution of Software**

## Algorithms and Science

- Algorithms/Programs are an essential part of nature
  - Genes, proteins, enzymes
  - conditioning, learntbehavior
- Information is neither matter nor energy
  - Einstein 1905: $E=mc^2$
  - Wiener 1950: Information is information, neither matter nor energy
- Algorithms are a kind of information
  - Wirth 1970: Programs = Algorithms + Data structures
  - Kowalski 1975: Algorithms = Logic + Control

## Some Taxonomies

- Several researchers proposed taxonomies to classify software visualization research and tools.
- Myers:90 *(→ next slides)*
- Price et al:93 *(→ next slides)*
- Roman and Cox:93 (similar to Price et al.)
  - Scope (code, data state,control state, behavior), abstraction, specification method, interface, presentation
- Oudshoorn et al: 96 *(→ next slides)*



**Software Visualization terms according to Price et al., 1993**

## Taxonomy [Price et al:93]

- Scope = range of programs used as inputs for the visualization
- Content = what information about the software is visualized
- Form = characteristics of output of system (e.g. medium)
- Method = how is the visualization specified
- Interaction = how does the user control the system
- Effectiveness = how well does the system communicate information to the user

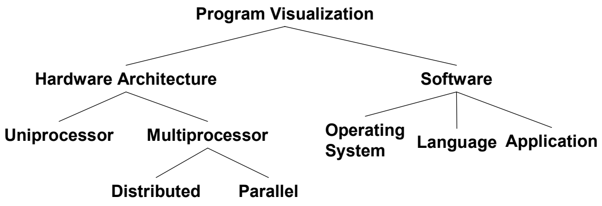
## Taxonomy [Myers:90]

- Program Visualization Taxonomy
- 6 regions

|  | Code | Data | Algorithm |
|---|---|---|---|
| Dynamic | 1 | 2 | 3 |
| Static | 4 | 5 | 6 |

## Taxonomy [Oudshoorn_et_al:96]

- Program Visualization Taxonomy



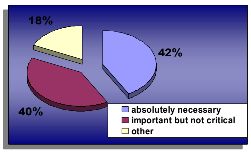
## Literature Survey [Diehl:02]

|  | Static Structure | Concrete Execution | Abstract Execution | Evolution of Static Structure |
|---|---|---|---|---|
| Systems | ■ | ▒ | □ | □ |
| Algorithms/ Programs | ■ | ■ | □ | □ |
| Abstract Machines | ▒ | ▒ | □ | □ |
| Real Machines | ■ | ▒ | □ | □ |

■ more than 100 ▒ more than 10 □ up to 10

**[see LNCS2269, 2002]**

## Importance of Software Visualization in Software Engineering



- Rainer Koschke's Survey
  - 100 researchers from software maintainance, re-engineering and reverse engineering
  - **40 %** find Software Visualization absolutely necessary for work
  - **42 %** find it important but not critical

**[see LNCS2269, 2002]**

## Survey [Bassil&Keller:01]

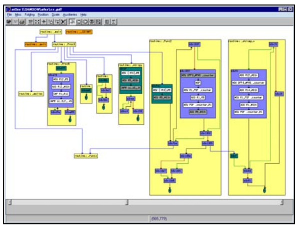
- 107 participants mostly from industry
- Benefits of SV tools
  - Savings in time and money
  - Better comprehension of software
  - Increase in productivity and quality
  - Management of complexity
  - To find errors
  - ...
- Whishlist
  - Integration of SV tools into other (third-party) tools
  - Better import/export of data and visualizations

## Examples of SV

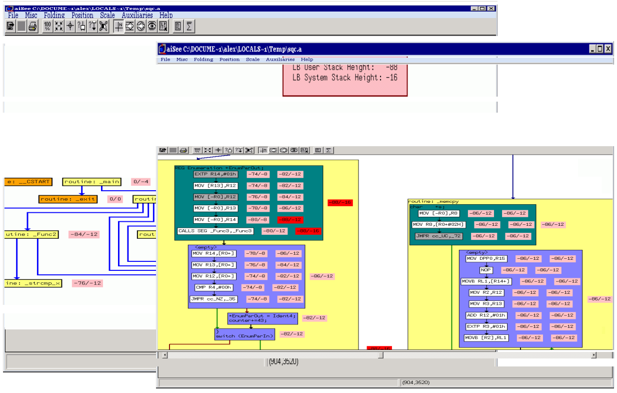
- CallGraph → Static Program Visualization
- X-Tango → Algorithm Animation
- SeeSoft → Software Evolution

## Development Tool: aiCall

- Visualization of Control Flow Graph of Embedded Applications
  - Results of Program Analyses: here stack usage
  - to prevent runtime errors due to stack overflow

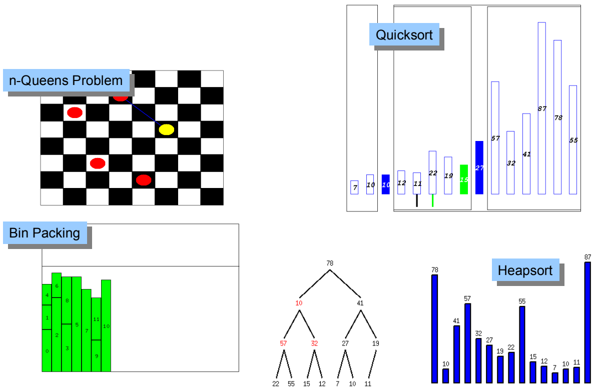


## Stack Usage Visualization

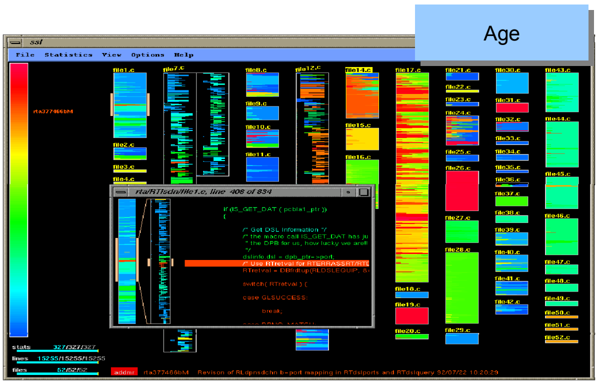

aiSee C:\DOCUME~1\alex\LOCALS~1\Temp\sqr.a
File Misc Folding Position Scale Auxiliaries Help

L8 User Stack Height: -88
L8 System Stack Height: -16

## Algorithm Animation: X-Tango



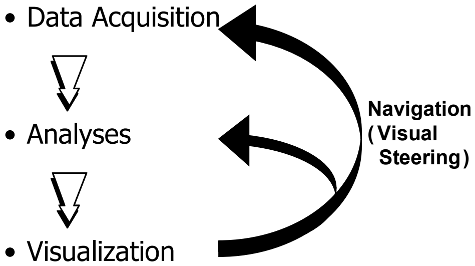
## Software Evolution: SeeSoft


Age
ssf
File Statistics View Options Help
rta/RTisdn/file1.c, line 408 of 834
addmr

```
if (IS_GET_DAT ( pcbla1_ptr ))
{

        /* Get DSL Information */
        /* the macro call IS_GET_DAT has ju
         * the DPB for us, how lucky we are!!
         */
        dslinfo.dsl = dpb_ptr->port;
        /* Use RTretval for RTERRASSRT/RTD
        RTretval = DBfrdtup(RLDSLEQUIP, &

        switch( RTretval ) {

        case GLSUCCESS:

                break;
```

rta377466bM Revison of RLdpnsdchn b-port mapping in RTdslports and RTdslquery 92/07/22 10:20:29

## Visualization Pipeline



- Data Acquisition
- Analyses
- Visualization

## Application domains and the visualization pipeline

| | Data Acquisition | Analysis | Visualization |
|---|---|---|---|
| Algorithm Animation | | | |
| Program Analysis | | | |
| (Visual) Debugging | | | |
| Software Design | | | |
| Software Evolution | | | |
| ■ ■ ■ | | | |

## Preliminary schedule

1. Introduction (today)
2. Static Program Representations
3. Program Analyses
4. Algorithm Animation
5. Case Studies (in Education)
6. Information Visualization
7. Visualization of Software Architectures
8. Software Evolution and Metrics
9. Visual Debugging
10. Visual Programming
11. Visualization Pipeline -- revisited

## Organizational Issues

- Exercises (Übungen)
  - Practical programming projects (electronic submission)
  - Written exercises (e.g. drawing of diagrams)
  - One short presentation (5-10 minute)
- → 50% of points required for admission to Final Exam (Klausur) on February 12th.
- Infos on the web:
  - **http://www.cs.uni-sb.de/~diehl/SoftVisVorles**
  - Slides and additional information (e.g. download of software)
- Questions:
  - **diehl@cs.uni-sb.de**
  - **Room 432, Building 45**