

How To Give Strong Technical Presentations

Markus Püschel

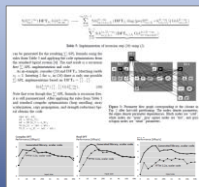
Department of Computer Science

ETH

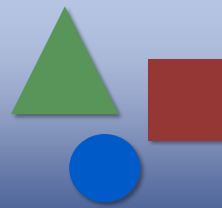
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich



Basics

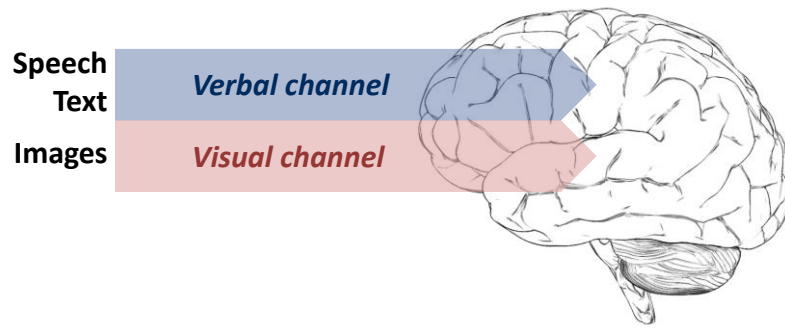


Content



Design

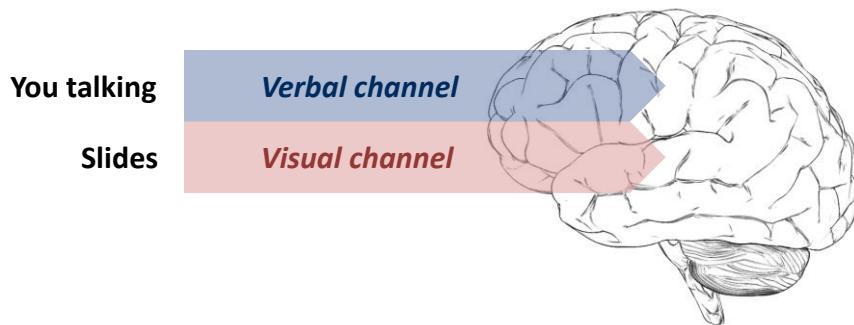
Biological Fact I: Text Versus Images



You cannot read and listen at the same time

3

Ideally



4

Handling the Medium

"Teleprompter"



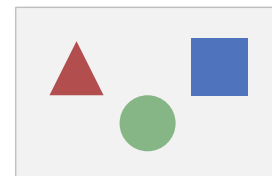
Really bad

"Slideument"



Better

Presentation



Ideal
(not always possible)

5
Nancy Duarte, *Slide:ology*

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Minimize Text

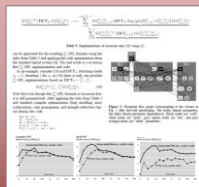
6

Good presentation slides
are not self-contained

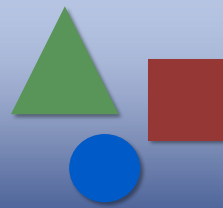
7



Basics



Content



Design

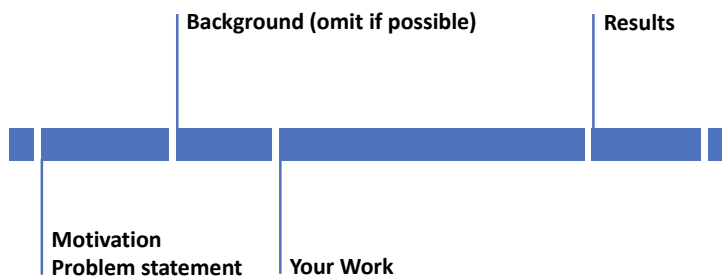
8

Be Clear About the Goal

Goal 1: ~~In these 30 minutes explain the entire approach and technology including all relevant details~~

Goal 2: In these 30 minutes explain what main problem the technology addresses, one or two key ideas in the approach, and one or two key results.
Get people excited to learn more.

Typical Organization





Motivation

Problem statement

- *What? Why? Why important?*
- Exceptionally clear
- If possible, precise problem statement:
 - Given ..., we want to compute ...
 - Input: ..., Output: ...
 - Block diagram showing input/output
- Start interesting
 - example result
 - interesting fact plus source
 - anything that starts the story

A presentation is a story
that starts on the first slide

The Problem

- **Computers architectures have become more complex**
 - Memory hierarchies
 - Vector extensions
 - Multiple cores
- **Optimizing for software for these features is very difficult**
 - Compilers fail to do it
 - Hence the software developer has to do it
 - Requires a high level of algorithm expertise: expensive
- **Performance does not port**
 - Needs re-optimization for every new processor
 - Without optimization: often 10x performance loss
- **Particularly noticeable for computing functions**
 - Matrix multiplication
 - Discrete Fourier transform
 - others

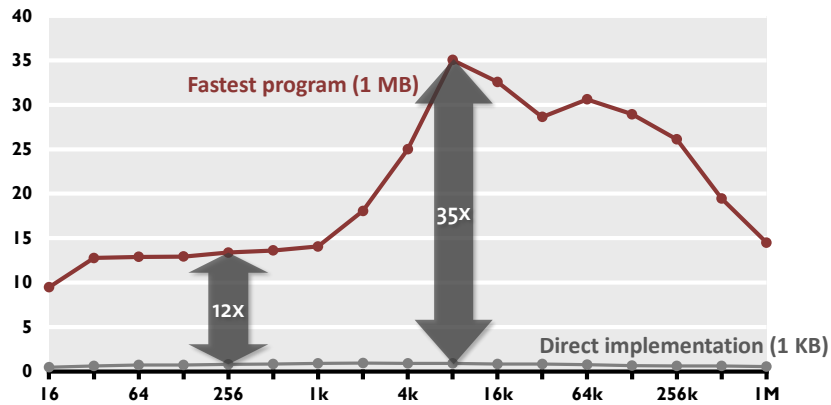
13

Don't start with a text-only bullet slide

The Problem: Example DFT

Discrete Fourier transform (single precision) on Intel Core i7 (4 cores)

Performance [Gflop/s]



■ Same operations count

■ Best compiler and flags

15



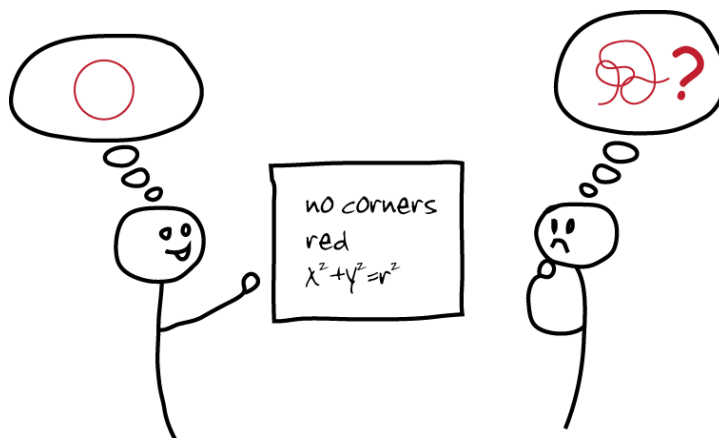
- Communicate main idea(s) and approach
- Do not (try to) communicate every detail of your work
- *How to explain technical work well?*

Explaining well

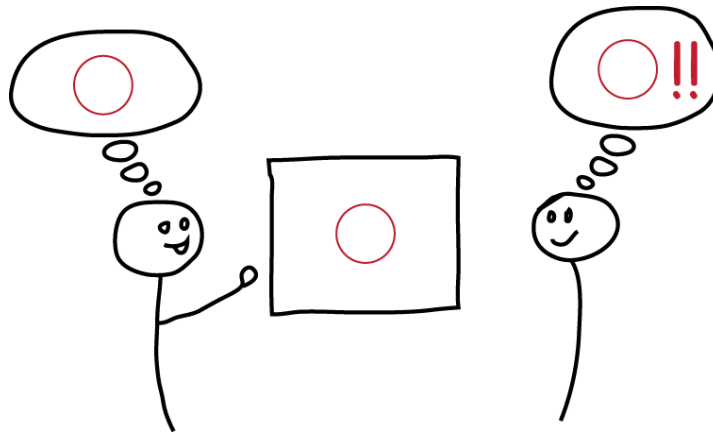
- *Visualize*
- *Use examples not generic explanations*
- *Small example, full truth*

17

Don't just talk about it



... show it!



Example: not so good

Statistical Classification: C4.5

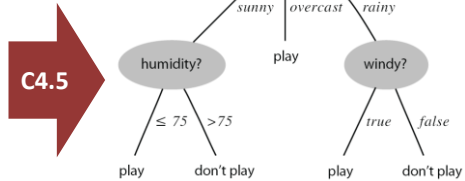
- C4.5 generates decision trees from training data
- The trees can be used for classification
- Formally:
 - Input: Training set of size m ; each member has n features

	Features				Classes
	$x_{1,1}$	$x_{1,2}$	$\dots$	$x_{1,n}$	c_1
Samples	$x_{2,1}$	$x_{2,2}$	$\dots$	$x_{2,n}$	c_2
	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
	$x_{m,1}$	$x_{m,2}$	$\dots$	$x_{m,n}$	c_m

- Output: decision trees mapping samples to classes

Statistical Classification: C4.5

Outlook	Temperature	Humidity	Windy	Decision
sunny	85	85	false	don't play
sunny	80	90	true	don't play
overcast	83	78	false	play
rain	70	96	false	play
rain	68	80	false	play
rain	65	70	true	don't play
overcast	64	65	true	play
sunny	72	95	false	don't play
sunny	69	70	false	play
rain	75	80	false	play
sunny	75	70	true	play
overcast	72	90	true	play
overcast	81	75	false	play
rain	71	80	true	don't play



$$P(\text{play} | \text{windy}=\text{false}) = 6/8$$

$$P(\text{don't play} | \text{windy}=\text{false}) = 2/8$$

$$P(\text{play} | \text{windy}=\text{true}) = 1/2$$

$$P(\text{don't play} | \text{windy}=\text{true}) = 1/2$$

$$H(\text{windy}=\text{false}) = 0.81$$

$$H(\text{windy}=\text{true}) = 1.0$$

Entropy of Features

$$H(\text{windy}) = 0.89$$

$$H(\text{outlook}) = 0.69$$

$$H(\text{humidity}) = \dots$$

21

Linear Transforms

$$\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{pmatrix} = y = Tx \leftarrow \boxed{T \cdot} \leftarrow x = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{pmatrix}$$

Output **Input**

Example: $T = \text{DFT}_n = [e^{-2k\ell\pi i/n}]_{0 \leq k, \ell < n}$

22

Fast Fourier Transforms (FFTs)

- Can be expressed as structured matrix factorizations

$$\text{DFT}_{mn} = (\text{DFT}_m \otimes I_n) T_m^{mn} (I_m \otimes \text{DFT}_n) L_m^{mn}$$

- Formalism:

$$L_n^{mn} \quad in + j \mapsto jm + i, \quad 0 \leq i < n, \quad 0 \leq j < m$$

$$I_n \quad n \times n \text{ identity matrix}$$

$$A \otimes B \quad [a_{k,\ell} B]_{0 \leq k, \ell < n}, \quad A = [a_{k,\ell}]$$

$$T_m^{mn} \quad \text{a diagonal matrix}$$

23

Fast Fourier Transform: Size 4

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix} x = \begin{bmatrix} 1 & \cdot & 1 & \cdot \\ \cdot & 1 & \cdot & 1 \\ 1 & \cdot & -1 & \cdot \\ \cdot & 1 & \cdot & -1 \end{bmatrix} \begin{bmatrix} 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot \\ \cdot & \cdot & \cdot & i \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdot & \cdot \\ 1 & -1 & \cdot & \cdot \\ \cdot & \cdot & 1 & 1 \\ \cdot & \cdot & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot \\ \cdot & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & 1 \end{bmatrix} x$$

12 adds, 4 mults

4 adds

1 mult

4 adds

Matrix formalism:

$$\text{DFT}_4 = (\text{DFT}_2 \otimes I_2) T_2^4 (I_2 \otimes \text{DFT}_2) L_2^4$$

24

Other Transform Algorithm

$$\begin{aligned}
 \text{DFT}_n &\rightarrow P_{k/2,2m}^\top (\text{DFT}_{2m} \oplus (I_{k/2-1} \otimes C_{2m} \text{rDFT}_{2m}(i/k))) (\text{RDFT}'_k I_m), \quad k \text{ even}, \\
 \begin{pmatrix} \text{RDFT}_n \\ \text{RDFT}'_n \\ \text{DHT}_n \\ \text{DHT}'_n \end{pmatrix} &\rightarrow (P_{k/2,2m}^\top I_2) \left(\begin{pmatrix} \text{RDFT}_{2m} \\ \text{RDFT}'_{2m} \\ \text{DHT}_{2m} \\ \text{DHT}'_{2m} \end{pmatrix} \oplus \left(I_{k/2-1} \otimes D_{2m} \begin{pmatrix} \text{RDFT}_{2m}(i/k) \\ \text{RDFT}'_{2m}(i/k) \\ \text{rDHT}_{2m}(i/k) \\ \text{rDHT}'_{2m}(i/k) \end{pmatrix} \right) \right) \begin{pmatrix} \text{RDFT}'_k \\ \text{RDFT}_k \\ \text{DHT}'_k \\ \text{DHT}_k \end{pmatrix} I_m, \quad k \text{ even}, \\
 \begin{pmatrix} \text{rDFT}_{2n}(u) \\ \text{rDHT}_{2n}(u) \end{pmatrix} &\rightarrow L_m^{2n} \left(I_k \otimes \begin{pmatrix} \text{rDFT}_{2m}((i+u)/k) \\ \text{rDHT}_{2m}((i+u)/k) \end{pmatrix} \right) \begin{pmatrix} \text{rDFT}_{2k}(u) \\ \text{rDHT}_{2k}(u) \end{pmatrix} I_m, \\
 \text{RDFT-3}_n &\rightarrow (Q_{k/2,2m}^\top I_2) (I_k \otimes \text{rDFT}_{2m}(i+1/2/k)) (\text{RDFT-3}_k I_m), \quad k \text{ even}, \\
 \text{DCT-2}_n &\rightarrow P_{k/2,2m}^\top (\text{DCT-2}_{2m} K_{2m}^{2n} \oplus (I_{k/2-1} \otimes N_{2m} \text{RDFT-3}_{2m}^\top)) B_n(L_{k/2}^{n/2} I_2) (I_m \text{ RDFT}'_k Q_{m/2,k}), \\
 \text{DCT-3}_n &\rightarrow \text{DCT-2}_{2m}^\top, \\
 \text{DCT-4}_n &\rightarrow Q_{k/2,2m}^\top (I_{k/2} \otimes N_{2m} \text{RDFT-3}_{2m}^\top) B_n(L_{k/2}^{n/2} I_2) (I_m \text{ RDFT-3}_k Q_{m/2,k}). \\
 \text{DFT}_n &\rightarrow (\text{DFT}_k I_m) T_m^n (I_k \text{ DFT}_m) L_k^n, \quad n = km \\
 \text{DFT}_n &\rightarrow P_n(\text{DFT}_k \text{ DFT}_m) Q_n, \quad n = km, \quad \gcd(k, m) = 1 \\
 \text{DFT}_p &\rightarrow R_p^\top (I_1 \oplus \text{DFT}_{p-1}) D_p (I_1 \oplus \text{DFT}_{p-1}) R_p, \quad p \text{ prime} \\
 \text{DCT-3}_n &\rightarrow (I_m \oplus J_m) L_m^n (\text{DCT-3}_m(1/4) \oplus \text{DCT-3}_m(3/4)) \\
 &\quad \cdot (F_2 I_m) \begin{bmatrix} I_m & 0 & -J_{m-1} \\ 0 & 1 & 2I_m \end{bmatrix}, \quad n = 2m \\
 \text{DCT-4}_n &\rightarrow S_n \text{DCT-2}_n \text{diag}_{0 \leq k < n} (1/(2 \cos((2k+1)\pi/4n))) \\
 \text{IMDCT}_{2m} &\rightarrow (J_m \oplus I_m \oplus I_m \oplus J_m) \left(\begin{pmatrix} 1 \\ -1 \end{pmatrix} I_m \right) \oplus \left(\begin{pmatrix} -1 \\ 1 \end{pmatrix} I_m \right) J_{2m} \text{DCT-4}_{2m} \\
 \text{WHT}_{2k} &\rightarrow \prod_{i=1}^t (I_{2^{k_1+\dots+k_{i-1}}} \text{WHT}_{2^{k_i}} I_{2^{k_{i+1}+\dots+k_t}}), \quad k = k_1 + \dots + k_t \\
 \text{DFT}_2 &\rightarrow F_2 \\
 \text{DCT-2}_2 &\rightarrow \text{diag}(1, 1/\sqrt{2}) F_2 \\
 \text{DCT-4}_2 &\rightarrow J_2 R_{13\pi/8}
 \end{aligned}$$

25



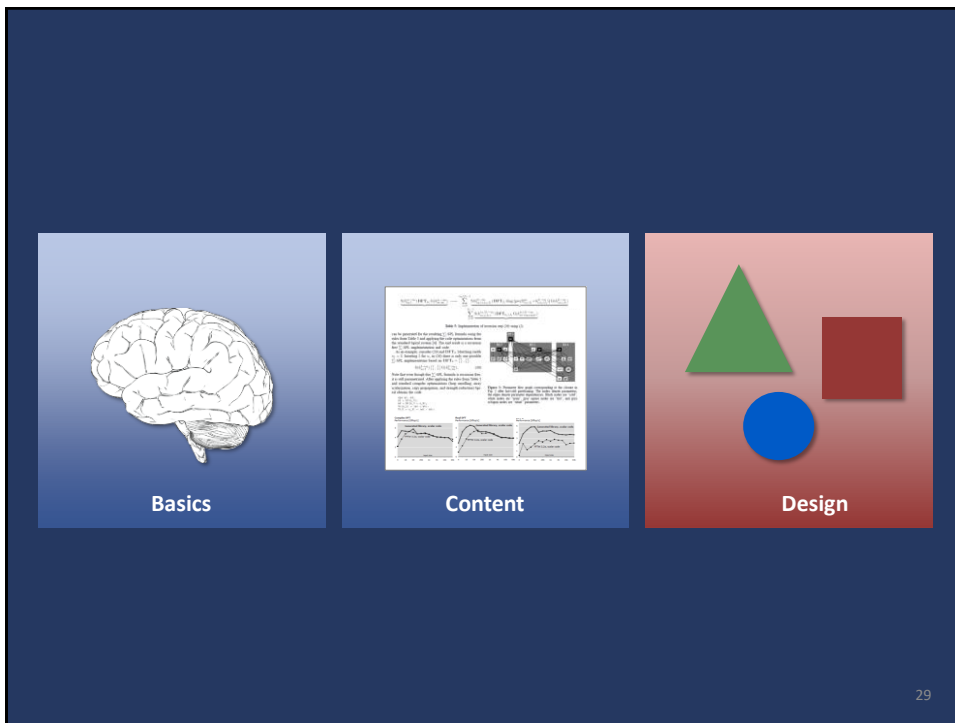
- Precise experimental setup
- More later on data

Common Mistakes

- Thinking: If one can understand it well, people will think it's trivial
- Too many slides
- Slides too packed

External Material

- Everything included with copy-paste:
Images, graphics, text (even if only one sentence)
- *Acknowledge source on the same slide!*
bottom right, gray is one option



Design is about efficient communication,
not about making things pretty

Two Design principles

- *Alignment*
- *Layering*

31

Alignment

- Everything is aligned to something else
- If in doubt align *left*

32

Nervousness

Top 10 fears

1. Fear of snakes
2. Fear of public speaking
3. Fear of heights
4. Fear of closed spaces
5. Fear of spiders (and insects)
6. Fear of needles
7. Fear of mice
8. Fear of flying
9. Fear of dogs
9. Fear of thunder
9. Fear of crowds

Source: U.S.A. Gallup Poll,
February 18-21, 2001
(1,016 respondents)

Practice the presentation

Be perfectly prepared

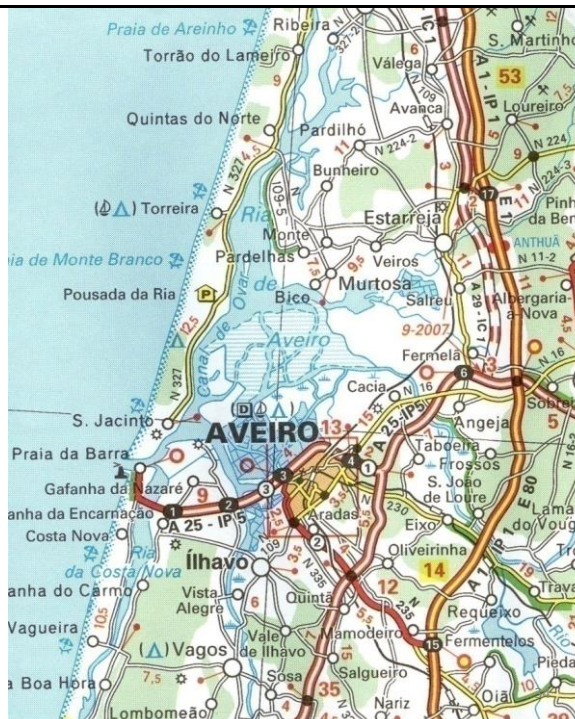
Take every small opportunity to present

If it's really bad

- Try some tricks from books
- See a specialist

Layering

Hierarchical organization of elements through proper use of contrast, emphasis, and de-emphasis



Nervousness

Top 10 fears

1. Fear of snakes
2. Fear of public speaking
3. Fear of heights
4. Fear of closed spaces
5. Fear of spiders (and insects)
6. Fear of needles
7. Fear of mice
8. Fear of flying
9. Fear of dogs
9. Fear of thunder
9. Fear of crowds

Source: U.S.A. Gallup Poll,
February 18-21, 2001
(1,016 respondents)

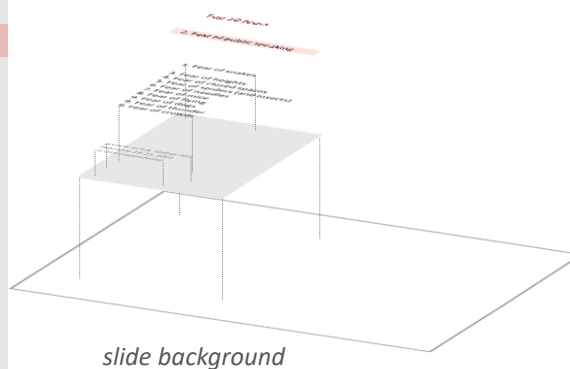
- Practice the presentation
- Be perfectly prepared
- Take every small opportunity to present
- If it's really bad
 - Try some tricks from books
 - See a specialist

Layering

Top 10 fears

1. Fear of snakes
2. Fear of public speaking
3. Fear of heights
4. Fear of closed spaces
5. Fear of spiders (and insects)
6. Fear of needles
7. Fear of mice
8. Fear of flying
9. Fear of dogs
9. Fear of thunder
9. Fear of crowds

Source: U.S.A. Gallup Poll,
February 18-21, 2001
(1,016 respondents)



Nervousness

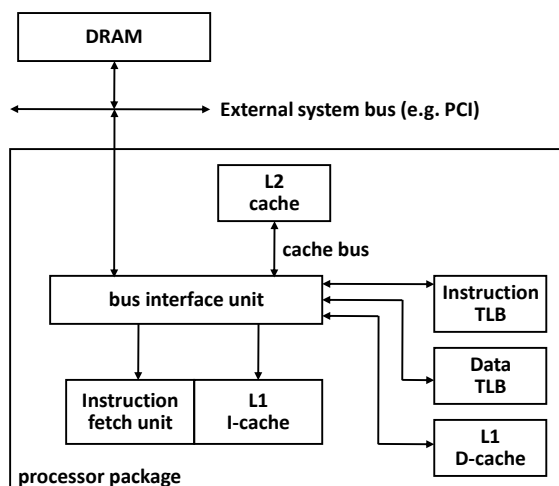
Top 10 fears

1. Fear of snakes
2. Fear of public speaking
3. Fear of heights
4. Fear of closed spaces
5. Fear of spiders (and insects)
6. Fear of needles
7. Fear of mice
8. Fear of flying
9. Fear of dogs
9. Fear of thunder
9. Fear of crowds

Source: U.S.A. Gallup Poll,
February 18-21, 2001
(1,016 respondents)

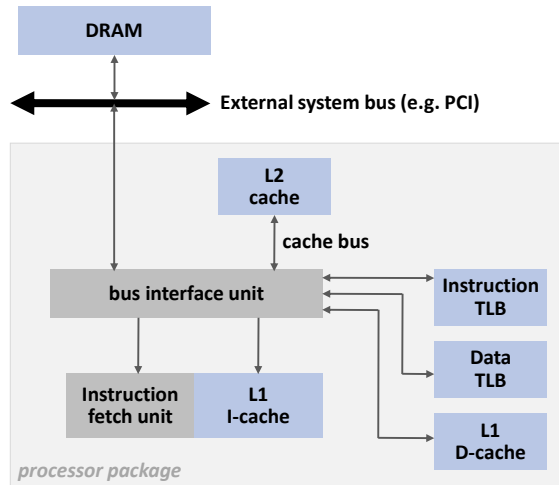
- Practice the presentation
- Be perfectly prepared
- Take every small opportunity to present
- If it's really bad
 - Try some tricks from books
 - See a specialist

P6 Memory System



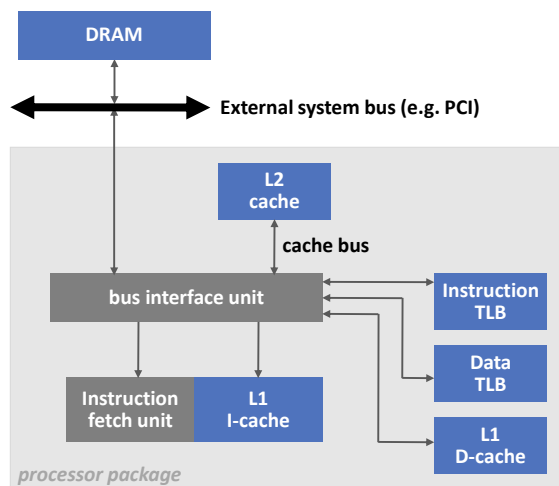
Example
good

P6 Memory System



Example
good

P6 Memory System



Code

```
#include <ia32intrin.h>

// n a multiple of 4, x is 16-byte aligned
void addindex_vec(float *x, int n) {
    __m128 index, x_vec;

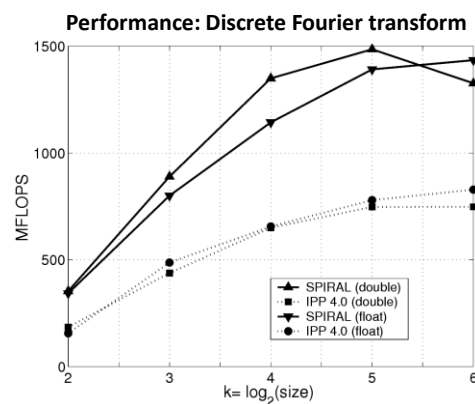
    for (int i = 0; i < n/4; i++) {
        x_vec = _mm_load_ps(x+i*4);           // load 4 floats
        index = _mm_set_ps(i+3, i+2, i+1, i); // create vector with indexes as values
        x_vec = _mm_add_ps(x_vec, index);      // add the two
        _mm_store_ps(x+i*4, x_vec);          // store back
    }
}
```

- Fixed-width font (Consolas or Courier)
- Layering for readability

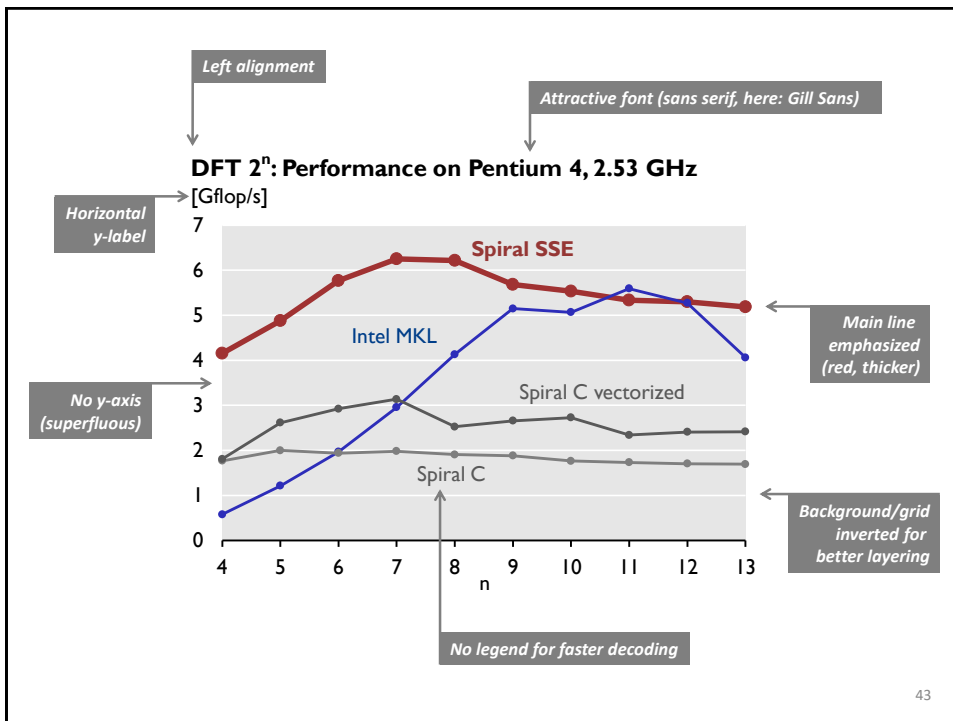
41

Presenting a Viewgraph: Example

- Start like this:
 - We compare the performance of Spiral and IPP
 - The x-axis shows k , the y-axis shows MFLOPS
 - This means higher is better (or vice-versa)
 - For example, this datapoint means that ...
- Now you can explain more
- Then conclude
- *But this plot is bad...*



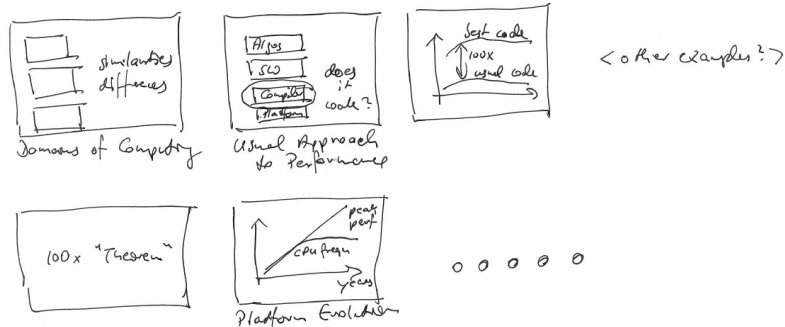
42



Final words

Creating a Presentation

- Think about good visuals (diagrams, graphs, fotos, screenshots) to support the story; then sketch the presentation on paper



45

How to Get Better

- Study the principles and apply them
- Give your best in every presentation
- Learn to verbalize the reason for design decisions and for problems with a slide
 - Explain and help others
 - Evaluate presentations you see
- Reduce text more and more
- Think hard about visualizations and good examples
- Experiment
- Expand your knowledge
 - Books (next slide)
 - Watch great presentations online (e.g., TED talks)

*Last Tip:
Never end with a*

Thank you!

slide

47

Some Books This Lecture Draws From

- Cliff Atkinson, ***Beyond Bullet Points***, Microsoft Press, 2005
- Nancy Duarte, ***Slide:ology***, O'Reilly, 2008
- Stephen Few, ***Show Me the Numbers***, Analytics Press, 2004
- Edward Tufte, ***Beautiful Evidence***, Graphics Press, 2006
- Edward Tufte, ***The Visual Display of Quantitative Information***, 2nd edition, Graphics Press, 2006
- Garr Reynolds, ***Presentation Zen***, New Riders, 2008
- Dan Roam, ***The Back of the Napkin***, Portfolio, 2008
- Robin Williams, ***The Non-Designer's Design & Type Books***, Peachpit Press, 2008