Disclaimer

These slides were presented at the President's Information Technology Advisory Committee's (PITAC) November 4, 1998 meeting by the chairs of its six panels. The panels were asked to suggest revisions to the PITAC's Interim Report. The information in these slides will be taken into consideration as the PITAC drafts its final report.

PITAC Software Sub-Panel

Dave Farber,
Hector Garcia-Molina (co-chair),
Jim Gray (co-chair),
Susan Graham,
Bill Joy,
Irving Waldawski-Berger
Bob Winner

Transformations & Expeditions: What are the barriers?

- Usability: Human-Computer interfaces
- How to build & evolve software?
- Trustworthy (reliability, security, integrity)

- Core software algorithms
- Base tools and components
- Interoperability

Summary of Software Findings

- Demand for software far exceeds the Nation’s ability to produce it.
- The Nation depends on fragile software.
- Technologies to build reliable and secure software are inadequate.
- The Nation is under-investing in fundamental software research.

Software Engineering

Fundamental research in

Software development methods and Component technologies

- component-based software design and production techniques
- study scientific and technological foundations needed for a software component industry,
- measurably reliable and secure components.
- aggregation into predictably reliable and secure systems,
- theories, languages and tools that support automated analysis, simulation, and testing of components and their aggregation into systems,
- protocols and data structures to promote interoperability of applications running in parallel across wide-area networks
- national library of software components in subject area domains.

Other Recommendations

- Make **software research a substantive component** of every major IT research initiative.
- Support fundamental research in **human-computer interfaces and interaction**.
- Major Recommendation:

  Make fundamental software research an absolute priority.

Our Process

- Presented report in many venues.
- Solicited criticism via Email
- Corresponded with NSF Software Workshop
- Committee interacted via Email and Teleconference
- Collected NSF data

Summary Reaction to Interim Report

- Strong agreement with conclusions, but...
- Broaden definition of software
- Strengthen

  Human Computer Interaction,
  Information Management,
  Middleware
  Applications… not strongly represented.
- Represent fundamental science.
- Recommended “Big Tent” software section.

How We Propose to Change the Report

- Let findings stand.
- Merged **Component Libraries** into “software engineering methods & component technologies”
- Elevated **Human Computer Interaction** to second bullet and broadened scope.
- Added section on **Information Management** (classic CS, middleware,…)
- Let major recommendation stand: “***make software research a fundamental component of every IT research initiative***”

Software Engineering

Methods, Component Technologies, & Component Libraries

- Enthusiasm that real progress is being made
- Great promise in Component Technologies.
- Much short-term commercial activity,
  but many opportunities for long-range research.

Why Elevate HCI and IM?

- Software Construction is an important problem -- as reflected by findings.
- What to build is also important.
- Biggest barrier to use of computers is usability.
- Theory and new algorithms are key to innovation.

Human Computer Interaction

- Simpler and more intuitive user interfaces
- Interactions are primitive today:
  keyboard, pointing, text, simple graphics.
  Style of interaction is un-coordinated.
- Fundamental research into perception, motor patterns, attention, decision making,…
- Technologies for vision, speech recognition, and synthesis, and information display.

Information Management technologies

- New approaches to
  digital libraries (content),
  knowledge discovery, and
  data visualization
- Fundamental computer science disciplines
  New data structures and algorithms,
  System software,
  Languages & Tools

Our Charter

1) What is currently funded?

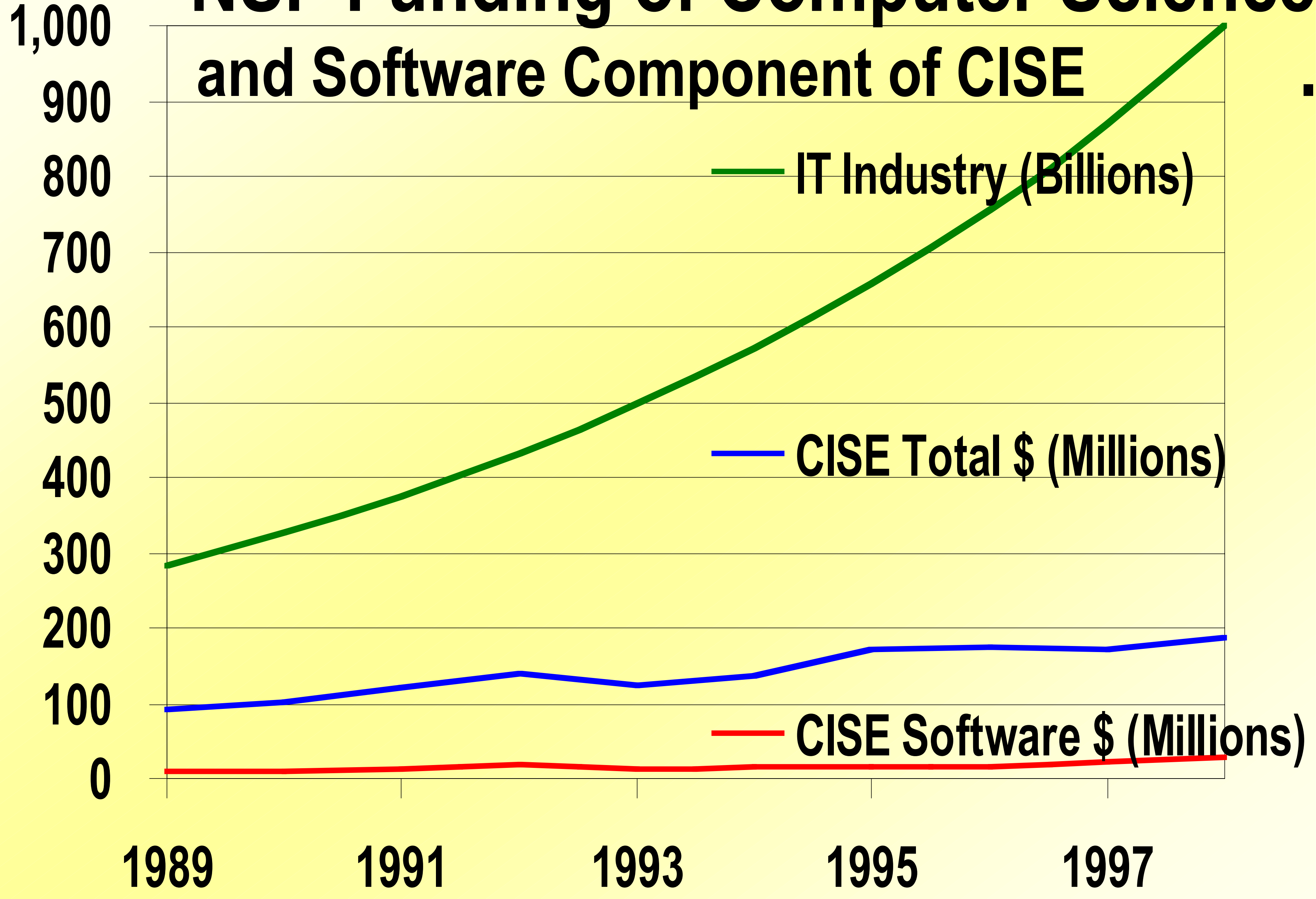
2) What incremental funding will provide?
Prioritize!

3) Give total funding recommendations for research goals.

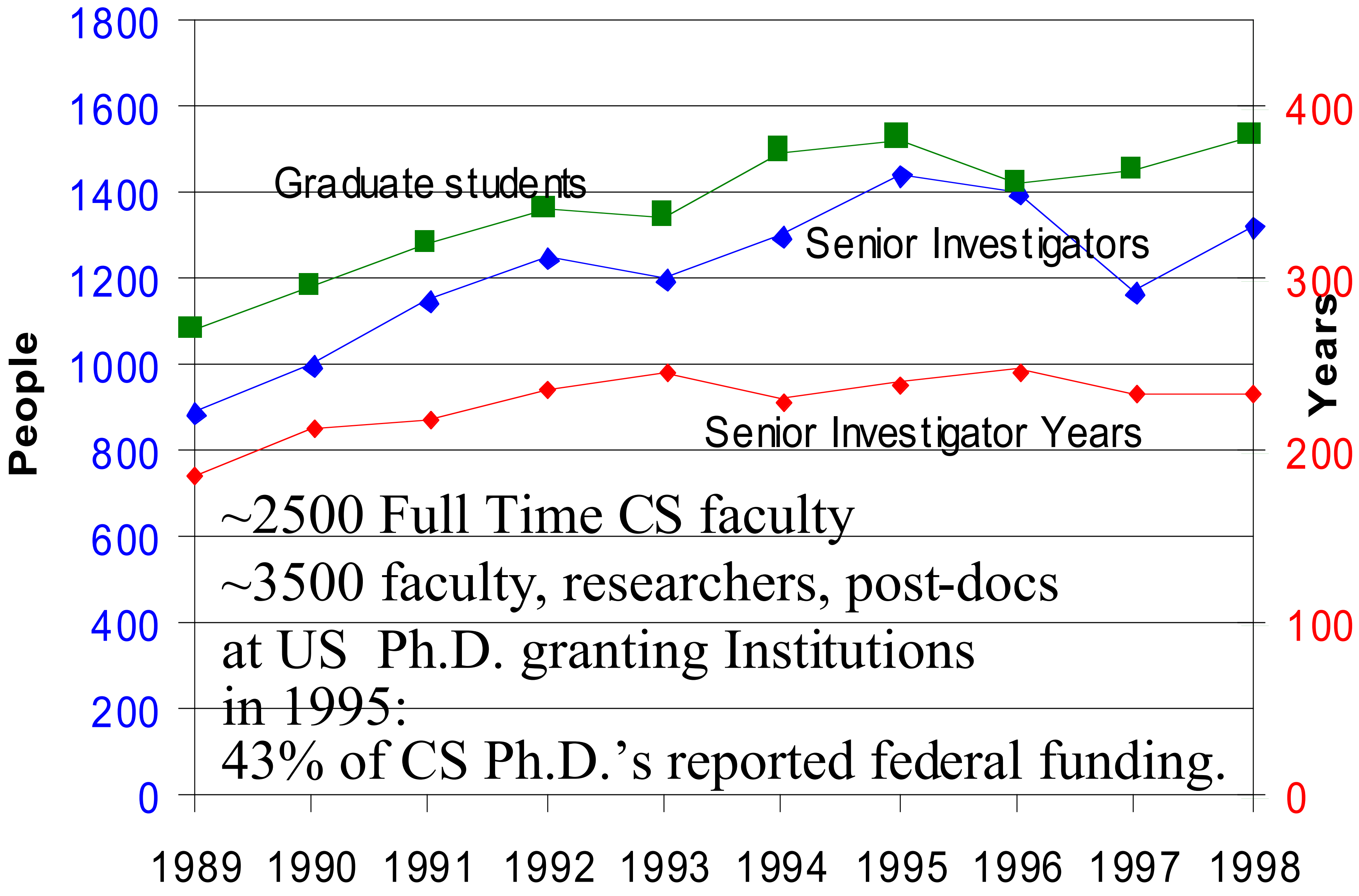
4) Is the current management structure effective?
If not, how should it be changed?

NSF Funding of Computer Science

and Software Component of CISE



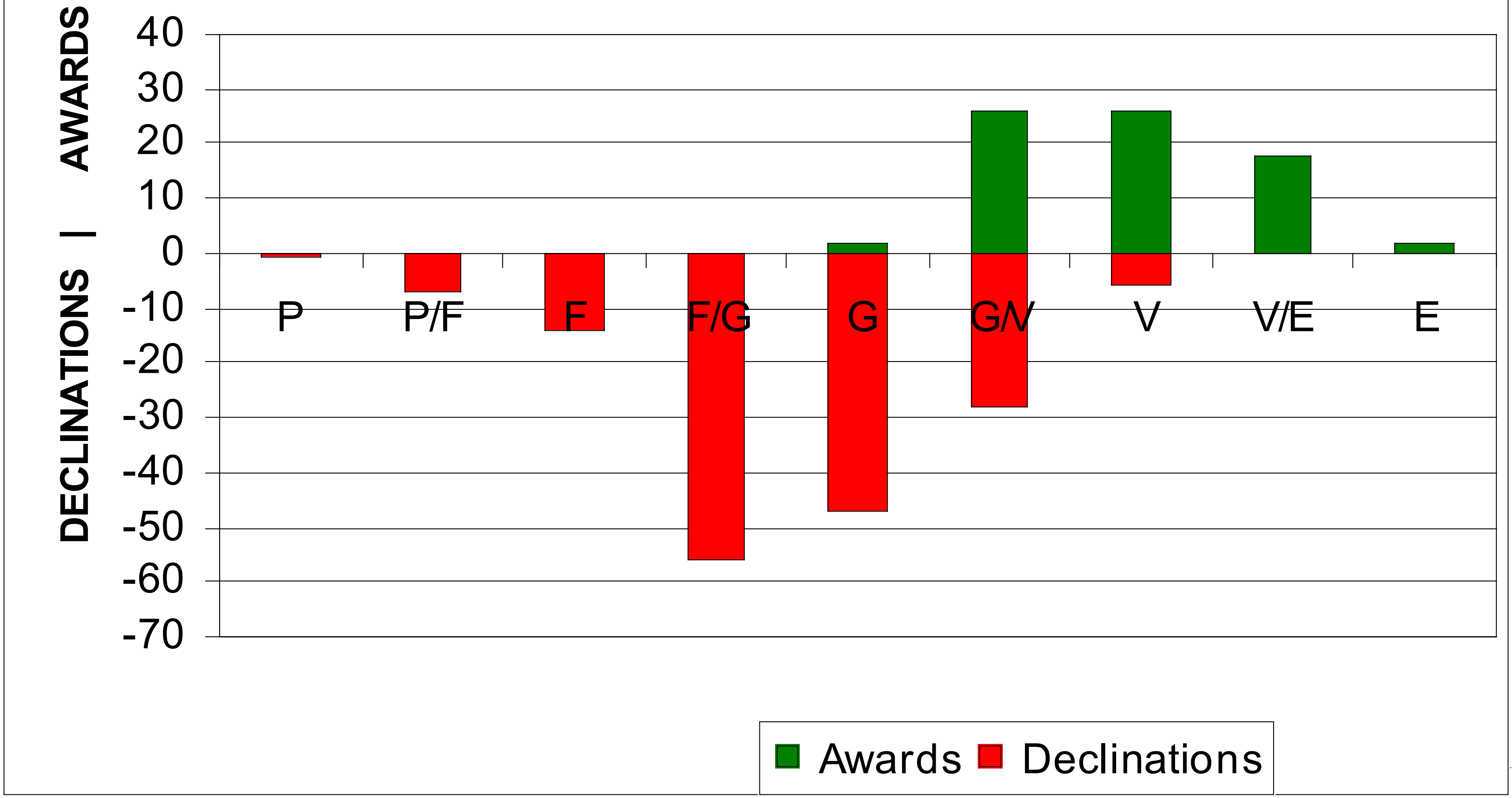
NSF CISE IT Research Support



~2500 Full Time CS faculty

~3500 faculty, researchers, post-docs
at US Ph.D. granting Institutions
in 1995:
43% of CS Ph.D.'s reported federal funding.

FY 1998
Operating Systems and Compilers & Software Engineering and Languages



Size and Duration NSF Grants

- Typical duration 2 years
- Typical size 75 k$/year
- 30% success rate

  (after pre-selection:

    - NSF grants too small
    - not worth the effort)
- 2 years is NOT long-term.

Observations

- KDI: 1300 LOI, 650 proposals, 70 grants
- Federal research investment has gone down
  - as % of it economy and
  - as % of IT industry
- IT is dis-inflationary
- We are eating our seed corn now
  - academic move to industry
  - funding short-term small-scale research
- Need to Invest in future

Recommendations

- Bring research investment in line with IT economy
- CUT GRANT OVERHEAD
  - Bigger grants (> 100k$/year)?
  - Longer (> 2 years)?
  - More grants (>30%)?
- Enable larger projects.

Summary

- Proposed Revision of Software Section
  - Consolidate “components”
  - Elevate HCI
  - Add Information Management
  - Let findings and software-research stand.
- Reduce overhead in grant process
- Enable long-term software research