9
Ten Pounds in a Five-Pound Sack

Ten Pounds in a Five-Pound Sack

- Program Space as Cost
- Size Control
- Space Techniques
- Representation Is the Essence of Programming

Program space as cost

- Brooks' applications quantified costs as rental on capital-intensive infrastructure
- Space, time and bandwidth are the fundamental "physical" quantities in software design
  - I.e., execution time v. storage cost v. bus speed
- Some applications emphasize the constraints
  - E.g. Mobile applications
  - CPU and memory trade against battery life

Size control

- Partly technical and partly managerial job
- Size targets should be based upon the applications and users
- System should be subdivided and each component should have a size target
- Program size and speed tradeoffs need to be considered

Size control (cont.)

- The exact functionality of a module must be identified when its size specification done
- Apart from optimizing each component, total effect on the system should also be considered
- System integrity has to be maintained
- Total-system and user-oriented attitude is the most important function of the programming manager

Space techniques

- More function means implies more space
- Trading function for size: How much choice to be given to the (programming) user?
  - Program with many optional features, each of which takes little space
  - For any particular set of options the program takes less space
  - Designer should decide on the granularity of the options available to the users

Space techniques (cont.)

- Range of suitability cannot be made arbitrarily wide even with fine-grained modularity
- Breaking functions into small modules costs both performance and space
- For deciding the modularity of a function the above factors need to be considered

Space-time tradeoffs

- More space usually means faster execution
- Space budgets need to consider this as well
- Manager should ensure that the team is trained in the techniques for a new language and platform
- Dedicated subroutines for performing commonly used functions should be developed
- At least two programs for the above functions; one which is fast and the other which is small

Representation is the essence of programming

- Strategic breakthroughs will improve upon both the space and the performance of the programs
  - E.g. New algorithms
- Brooks suggests that often strategic breakthroughs will come from reworking of the data and/or tables
- E.g. Digitek's Fortran compiler uses a very dense specialized representation for the compiler code, so that external stoage is not needed.
- Time lost in decoding is gained in back in avoiding input-output

Hierarchies

CPU registers L1 cache L2 cache Dynamic RAM Disk controller buffers Magnetic medium	Storage
CPU bus Peripheral bus Local Area Network controller Regional network provider National network provider Global network provider	Communication