Introduction to **PC farm**

A flexible, scalable solution
to achieve
high computing power

Yen-Chu Chen
Institute of Physics
Academia Sinica

PC farm

- Introduction
- Hardware configuration
- Software configuration
- Operation
- Example:
  - E871 production farm
  - CDF production farm
- Conclusion

Introduction

- The need of high computing power around the world.
- FNAL, a place where people need and work on high computing power.
- Concepts
  - Reliability
  - Scalability
  - Flexibility

Nature of Computing

- Highly sequential
- Semi-parallel
- Highly parallel

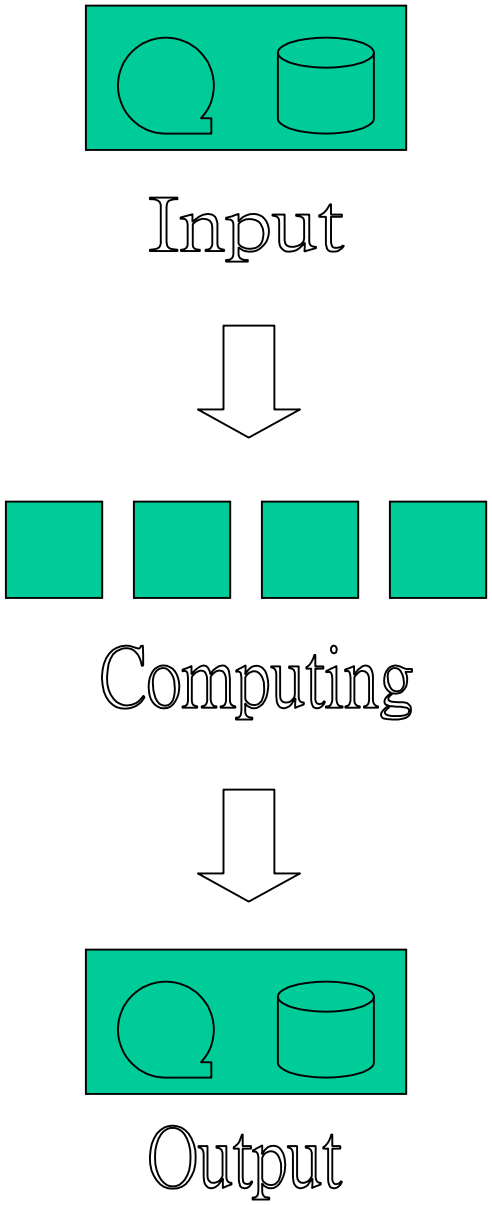
- Highly sequential
  - Task can be done only sequentially.
  - Very limited IO is required.
  - Need high performance CPU.
  - Example:
    - Solving an equation or a set of equations.

- Semi-parallel
  - Task can be separated into smaller tasks.
  - Information between tasks are required.
  - Limited IO.
  - Need high performance CPU and good communication bandwidth.
  - Example:
    - Lattice gauge theory, …..

- Highly parallel
  - Task can be separated and each smaller task can be computed completely independently.
  - Usually need a lot of IO.
  - Needs good bandwidth of IO.
  - High performance CPU is preferable.
  - Example:
    - High energy experiment data analysis, ...

Hardware configuration

- Consideration
  - Balancing along the IO stream
- Input
  - Tapes or files on local (or network) disk
  - distributing
- Computing
  - Number of worker nodes
- Output
  - disk files, tapes
  - concatenating



Software configuration

- Fundamental:
  - LSF : Load Sharing Facility
    - Commodity software
    - An utility that share the load on computers clustered together.
  - FBS : Farm Batch System
    - Developed by FNAL computing division.
    - Based on LSF.
    - Allow user to submit job with desired computing power.

- Input section
  - Read in data from tape drive or any storage system and distribute smaller files into local disks.
- Computing section
  - Check if the ‘Input section’ is done.
  - Worker nodes compute in parallel.
  - Result is written to local disk.
- Output section
  - Check if the ‘Computing section’ is done.
  - Transfer all files in local disks to disk of IO nodes.

Storage IO

- IO nodes
  - read data from storage system directly and
  - distribute small files to worker nodes.
  - Split the result file into several output streams.
  - Concatenate files in output streams together and put them onto tape when there is one tape worth.

PC as a workstation

- Speed
  - 113 MIPS for Pentium Pro 200
  - 197 MIPS for Pentium II 350
  - scalable with clock rate
- OS
  - Linux, Windows NT
- Compiler
  - g77, absoft F77, ...
  - gcc(g++), KCC, ...
- Software
  - CERNLIB, FNAL tools
- Comparison of results on PC and on SGI, IBM

Examples

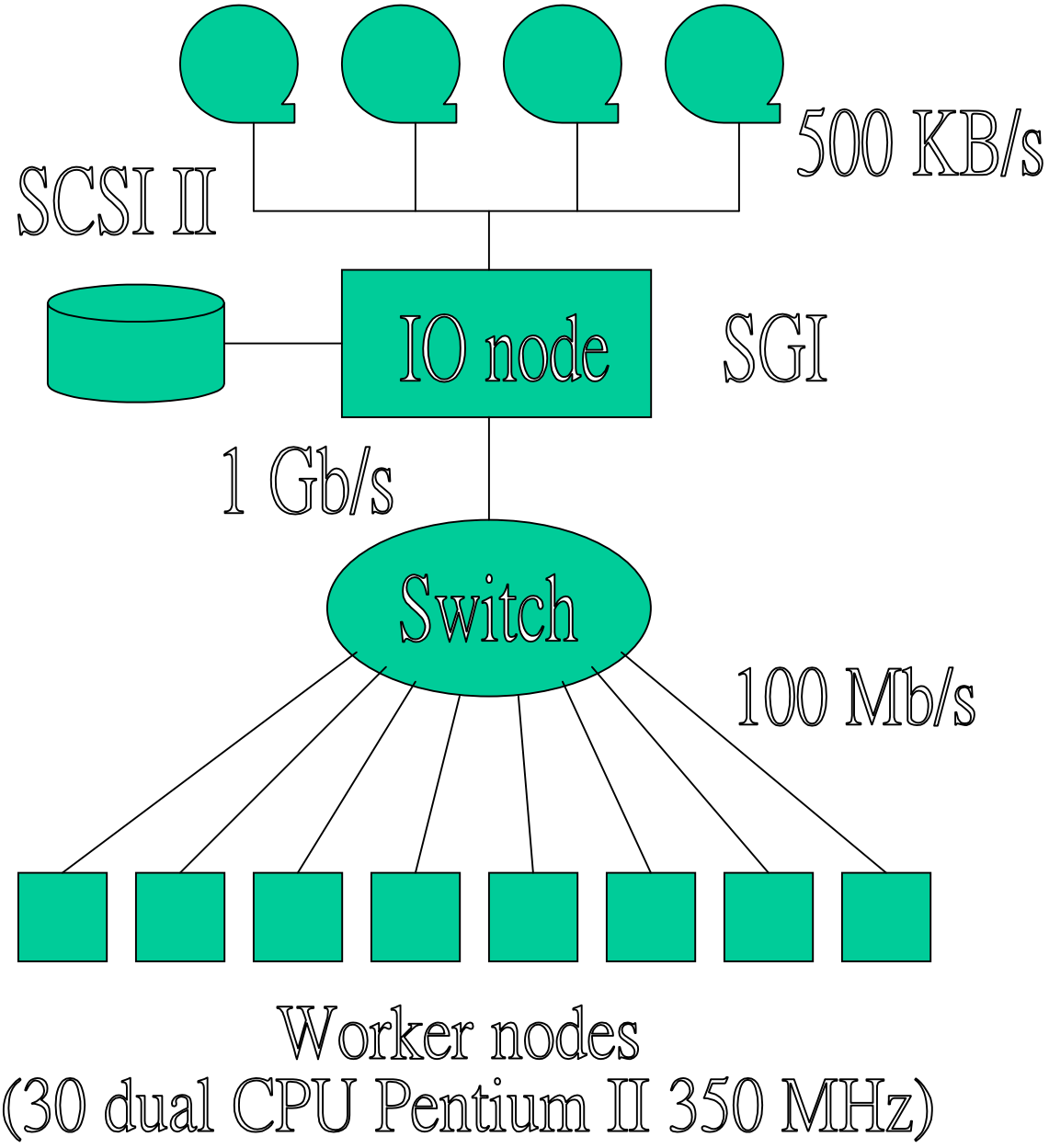
- E871 production farm

- CDF production farm
  - prototype farm

E871 production farm

- E871 needs high computing power to reach 10^{-4} sensitivity
  - 10,000 tapes in vault from ‘97 run. Each tape holds 5 GB.
  - With 4000 MIPS, E871 needs one year just to go through all data sample once!
  - 20,000 more tapes with higher event density are expected in ‘99 run. Four times more events are expected.

Hardware configuration of E871 PC farm



IO Balancing

- Hardware IO rates
  - Tape drive speed: 500 KB/s.
  - Ethernet: 1Gb/s, 100 Mb/s.
- Rates of analysis
  - E871 event size: 550 Byte/event.
  - To keep tape drive spinning at full rate E871 needs 1.1 ms/event.
  - CPU time usage: 5.5 ms/event.
  - 10 process per tape. Each process takes 3 hours to finish.
  - 2 processes per CPU.
  - 60 CPU's in 30 dual PC nodes. 120 processes in total.
  - 12 tapes per 3 hours!

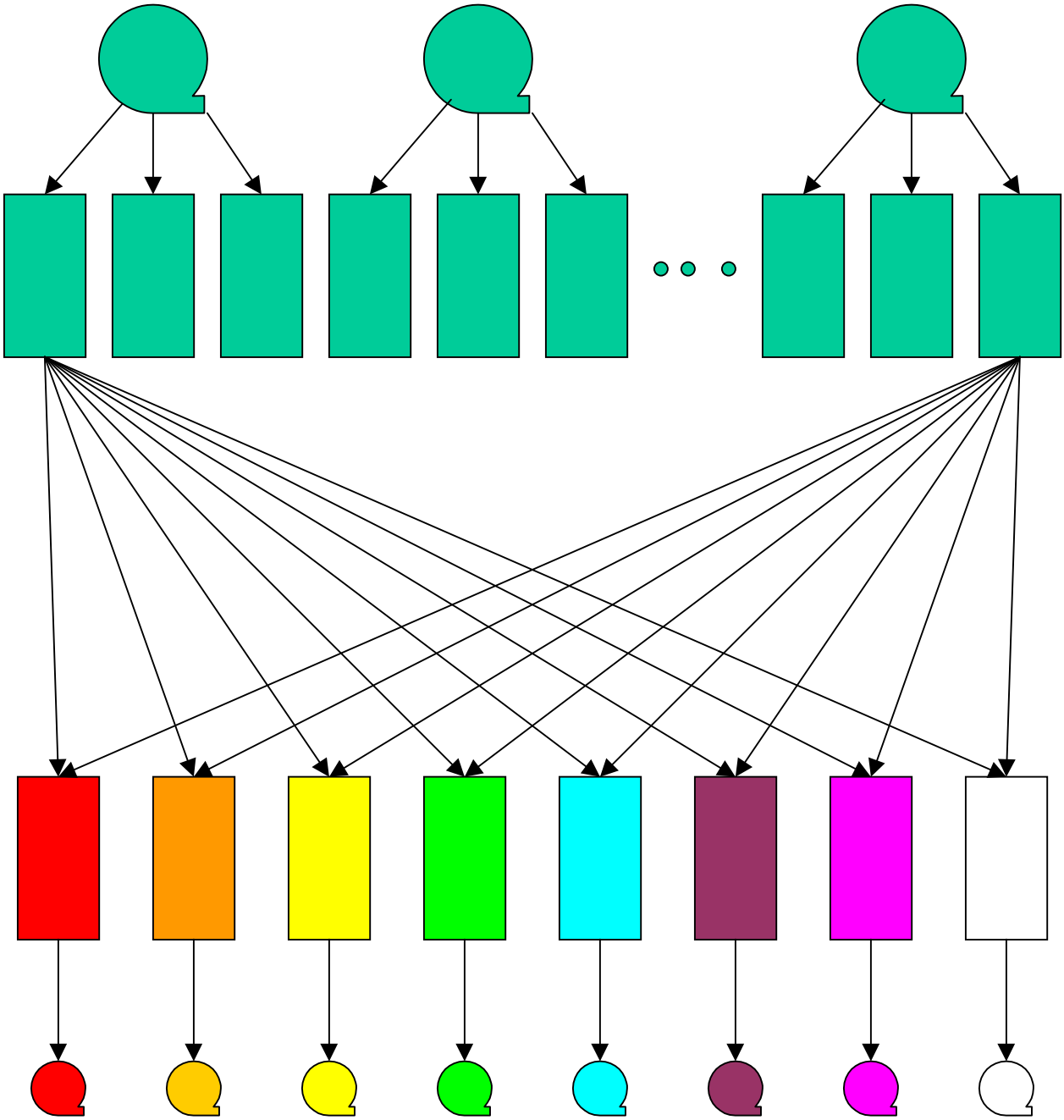
Operation

- Input
  - Split data of each tape into 10 disk files located on disk of IO node.
- Computing
  - 10 processes run on 5 PC nodes to analyze these 10 files.
  - Result of each process remain on local disk of PC node.
- Output
  - Result files are transferred to disk on IO node.

Storage IO

- Storage
  - When there is enough data in a stream for one tape, concatenate files of that stream and put result onto tape.

Data Streams



Status

- More than 5% of non-polarized data sample is analyzed by PC farm.
- A slow ramping process, low efficiency when start up.
  - It is a whole new system, a lot of problems, such as the reliability of transferring files.
  - Inefficiency was due to
    - operator mount tape wrongly
    - mislabeling
    - missing constant files
    - …
- Estimate 3 months to finish the production.

CDF production farm

- Based on the experience of E871 PC farm and RUN I production.
- There will be hundreds of PC's.
  - Enough capability to process data taken on-line in real time.
- The procedure should be sort of automatic.
- Good error handling.
- Good book keeping.
- Enough flexibility to redirect the computing power.

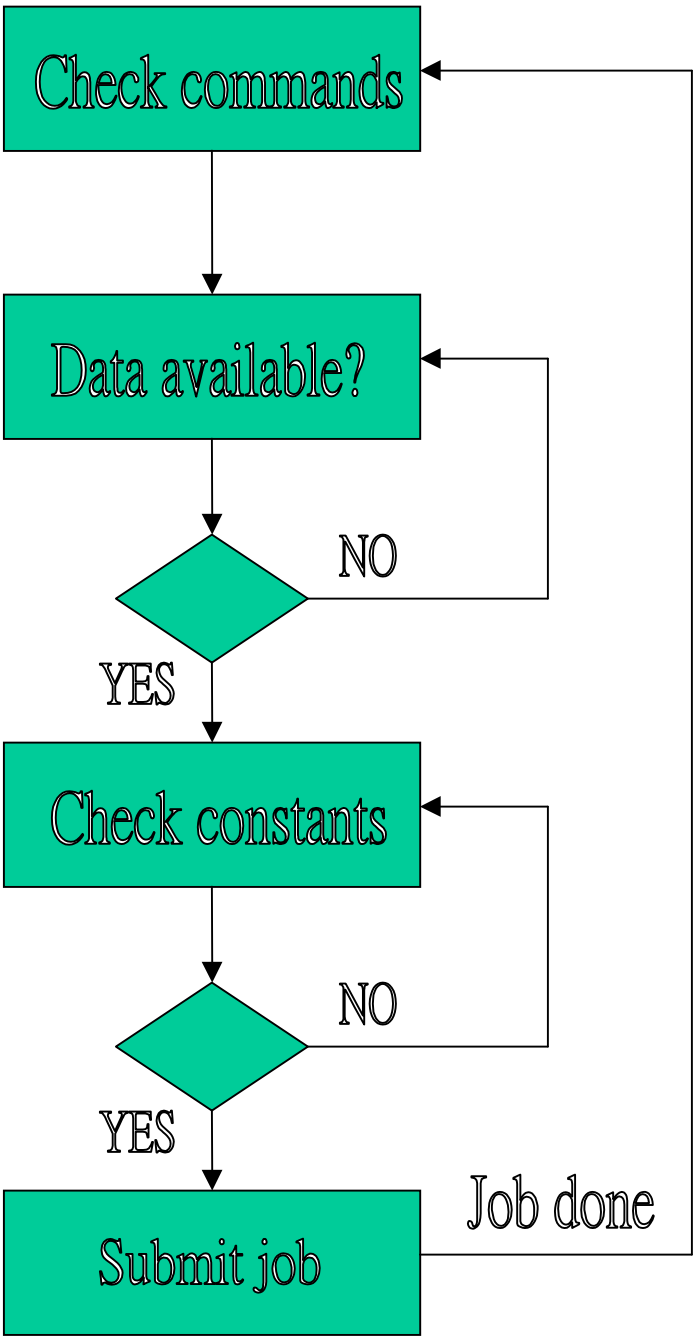
Hardware Configuration

- Expanded version of the E871 PC farm with the tape drives replaced with mass storage system which is capable of PB.
- All IO nodes and worker nodes are PC's running LINUX.
- There will be hundreds of PC's.
- GB Ethernet link.
- SCSI III or fiber channel for device connection.

Software Configuration

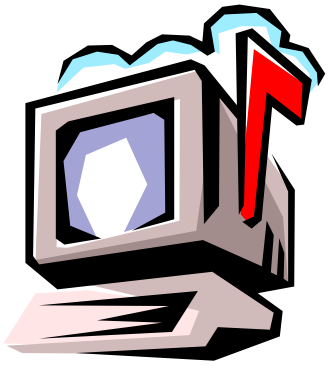
- Based on LSF, FBS.
- Has similar job submission procedure as in E871 PC farm.
- On top of job submission there is a loop to run the procedure automatically.

The Loop


Check commands
Data available?
NO
YES
Check constants
NO
YES
Job done
Submit job

Status

- Prototype farm has
  - 10 dual CPU Pentium II 350 MHz PC's as worker nodes and
  - 3 dual CPU Pentium II 350 MHz as control node and IO nodes.
- CDF RUN II executable runs on prototype farm.
- Exercising FBS.
- Planning the details of the software automation loop.
- Expect to have a small scale PC farm running in Summer of '99.

Conclusion

- PC farm is highly scalable.
- PC farm is flexible in using resources.
- PC farm is a reasonable solution to achieve high computing power.
- Enjoy physics when there is enough computing power to get result in time.