

# AS/400

IBM  server

( OS/400 )

## *Architettura*

*Liberamente tratto da materiale fornito da  
Pierluigi Corno*

© 2002 IBM Corporation

## The AS/400's Architecture Goes Beyond Technology

Per parlare dell'As400 non è possibile ricorrere ad analogie con altri calcolatori

è invece necessario rifarsi ai seguenti concetti architettonici :

- technology independence
- object-based design
- hardware integration
- software integration
- single-level storage

## Technology Independence

Unlike other computer systems, the AS/400 isn't defined by hardware

A program doesn't "speak" directly to the hardware:

it "speaks" to a technology-independent machine interface (TIMI, or simply the machine interface).

Between this interface and the actual hardware are about four million lines of operating system software called the System Licensed Internal Code (SLIC).

This software layer insulates application programs from the underlying hardware characteristics.

The knowledge of hardware remains entirely within the SLIC.

Due to this technology-independent design, customer application programs are also unaware of technology changes and can exploit new technology immediately

when the processor technology changes, IBM can rewrite the SLIC components that are aware of those technology changes and thus preserve the integrity of the machine interface. The importance of this architectural principle was dramatically illustrated when AS/400 processor technology moved from 48-bit CISC to 64-bit RISC. Many customers needed merely to save their programs off their CISC machines and restore them on their new RISC machines to run them as fully 64-bit programs.

### **No other system can do that:**

conventional architectures require at least a recompilation of legacy programs and usually some rewriting before you can run them in a different system architecture.

The AS/400 lets customers run original 48-bit applications as 64-bit applications on a 64-bit operating system that contains a 64-bit relational database.

## Object-based Design

everything within the system — programs, data files, message queues — is an *object*.

Each object has two inseparable parts:

- *Descriptive part*, which defines the valid ways of using that data;
- a *data part*, which serves as the

If an object is defined as a program, its descriptive part states that the data part will be treated as executable, read-only, compiled code. The only operations allowed on this object are those that make sense for a program.

*For example, you can write into the middle of a data file, but you can't write into the middle of compiled code; the system won't let it happen.*

AS/400 objects' two-part design insures data integrity for all objects in the system and has important security implication.

*For example, one mechanism by which computer viruses enter systems is by masquerading as data. Once inside, the virus tries to become executable code and wreak havoc. Such a change of characteristics isn't possible on the AS/400 — if the system permits a package to enter the system as data, it must retain the characteristics of data forever. It cannot change its mind and become executable code.*

As a key part of the AS/400's fundamental design, objects are one of many reasons that the AS/400 enjoys a recognised reputation for rock-solid security and integrity.

## Hardware Integration

- The engineering/scientific computing environment is *compute intensive* (meaning users perform complex operations on a relatively small amount of data) – << integer >>
- the general business computing environment is *information intensive* (meaning users perform simple operations on a large amount of data). << array of char >>

The AS/400 is optimized for the general business environment, it contains hardware design characteristics that enable it to deliver outstanding performance in an information-intensive environment.

In a typical business transaction, an application program is loaded into main storage, and the main processor then begins to execute it.

When the main processor comes across a request for data to be read from disk, for example, it delegates that request to the input/output processor (IOP) that's dedicated to the disk device.

Then the main processor diverts its attention to another application program - the task that it's dedicated to doing — and returns to the original program only when the data it requested earlier is available in main storage.

On a large AS/400, you can have over 200 IOPs connected to high-speed buses, creating an extremely powerful server. Such a server design is excellent for information-intensive applications.

Certain software items are fundamental to all businesses.

In addition to the basic operating system drivers that handle the various I/O devices, businesses always need software for standard computing functions, such as security, communications, Web-serving, and backup and recovery

With a traditional system, customers often must purchase additional software components to add to their base operating system

These customers must also ensure that the release levels of additional modules are compatible with the release levels of all the other items they plan to integrate.

With the AS/400, however, *all necessary*\* business software components are fully integrated into the standard operating system.

*\* l'esagerazione è voluta !*

IBM tests all these components in the context of the other components, so the entire operating system works as one entity.

When IBM makes changes to OS/400, it gives customers a new release of the entire operating system.

*there are never any release conflicts between individual components of OS/400 because IBM ships a complete, fully tested operating system to customers with each release.*

Two benefits of this highly integrated operating system are immediately apparent to customers:

- fast deployment of new business solutions
- a remarkably low total cost of ownership.

*E il rovescio della medaglia ?*

## Single level Storage

1/2

**The AS/400's massive 64-bit address space can address 18 quintillion bytes of data! Architecturally, the AS/400 is designed to be capable of even more than this.**  
( *up to 128-bit addressing* ).

Mapped into this 64-bit space is the "real" storage: disk drives and main memory

Customers need not be aware of any of the storage technologies that underlie the huge address space because the AS/400 manages them automatically.

All programs and data simply reside in this massive space ( they are object.)

Users don't need to worry about where a program resides; they need only reference it by name.

when customers add more storage devices to the machine, they don't need to redistribute data across them; the system recognizes the new available storage and uses it. Most AS/400 installations don't even have a traditional database administrator because they don't need one. The system does much of this type of work on its own.

## Single level Storage

2/2

Processing business applications in a multi-application, multi-user environment involves frequent switching between different tasks.

Because of its single-level storage, the AS/400 accomplishes this function much more efficiently than conventional systems.

Switching to a new task in the AS/400 is as simple as performing a branch instruction to the location where the new task resides.

No need (as there is in Unix and Windows systems) to re-create a separate address space before the execution of a new task can begin.

Designed for the frequent task-switching that characterizes business environments, the AS/400's single-level storage not only simplifies storage management, but it also delivers exceptional performance.

## INDIPENDENZA DALLA TECNOLOGIA

“La possibilità di modificare e di estendere l'hardware e il software di livello più basso senza impatto sui programmi applicativi”

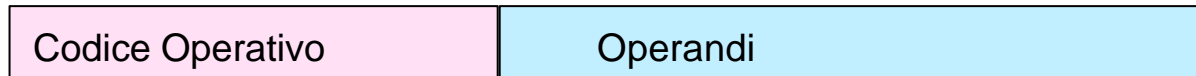
### INDIPENDENZA HARDWARE

- processori
- memoria
- dispositivi I/O

### INDIPENDENZA SOFTWARE

- funzioni di kernel
- funzioni di supervisore

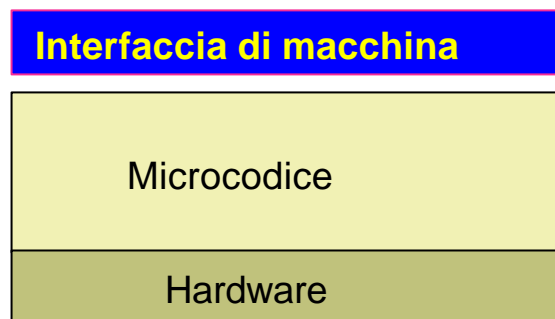
# ISTRUZIONI di MACCHINA



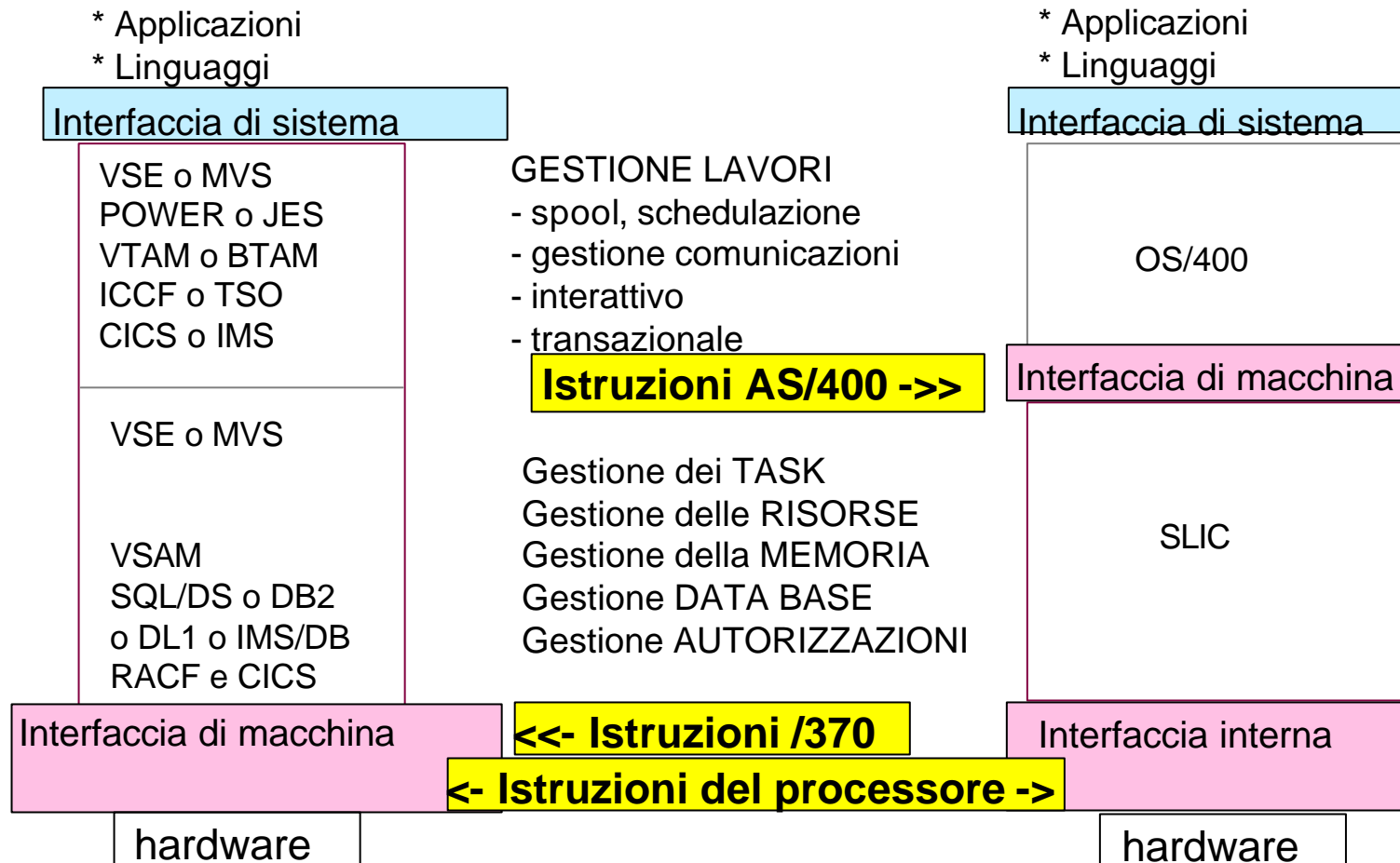
## ARCHITETTURA CONVENZIONALE



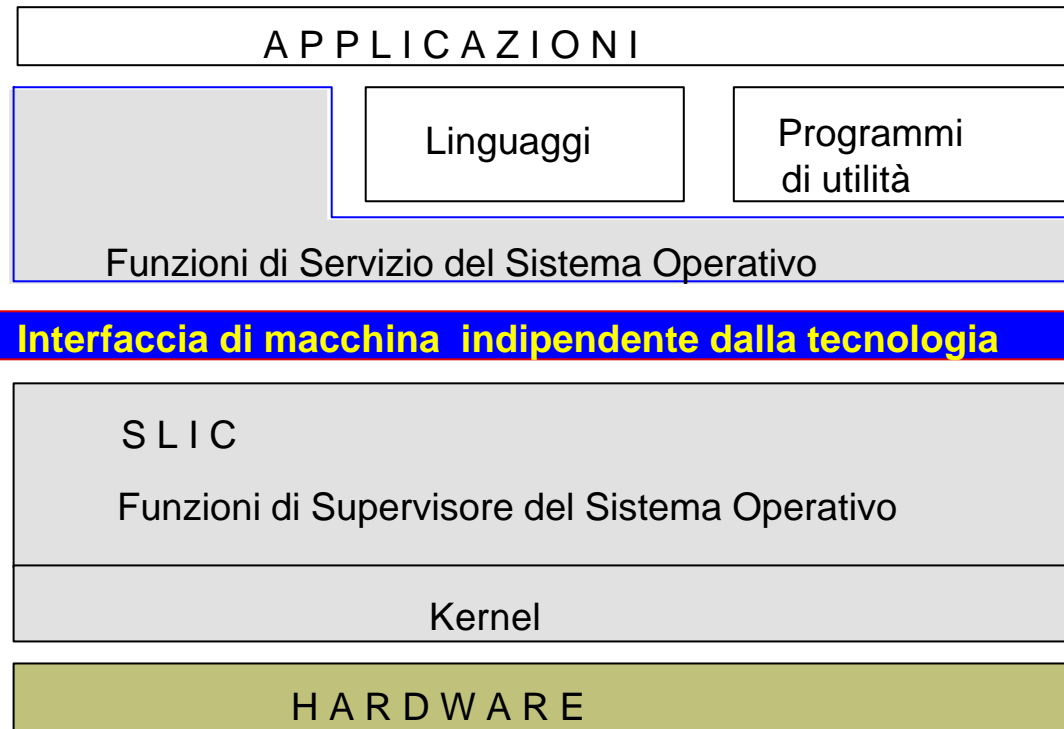
## ARCHITETTURA AS/400



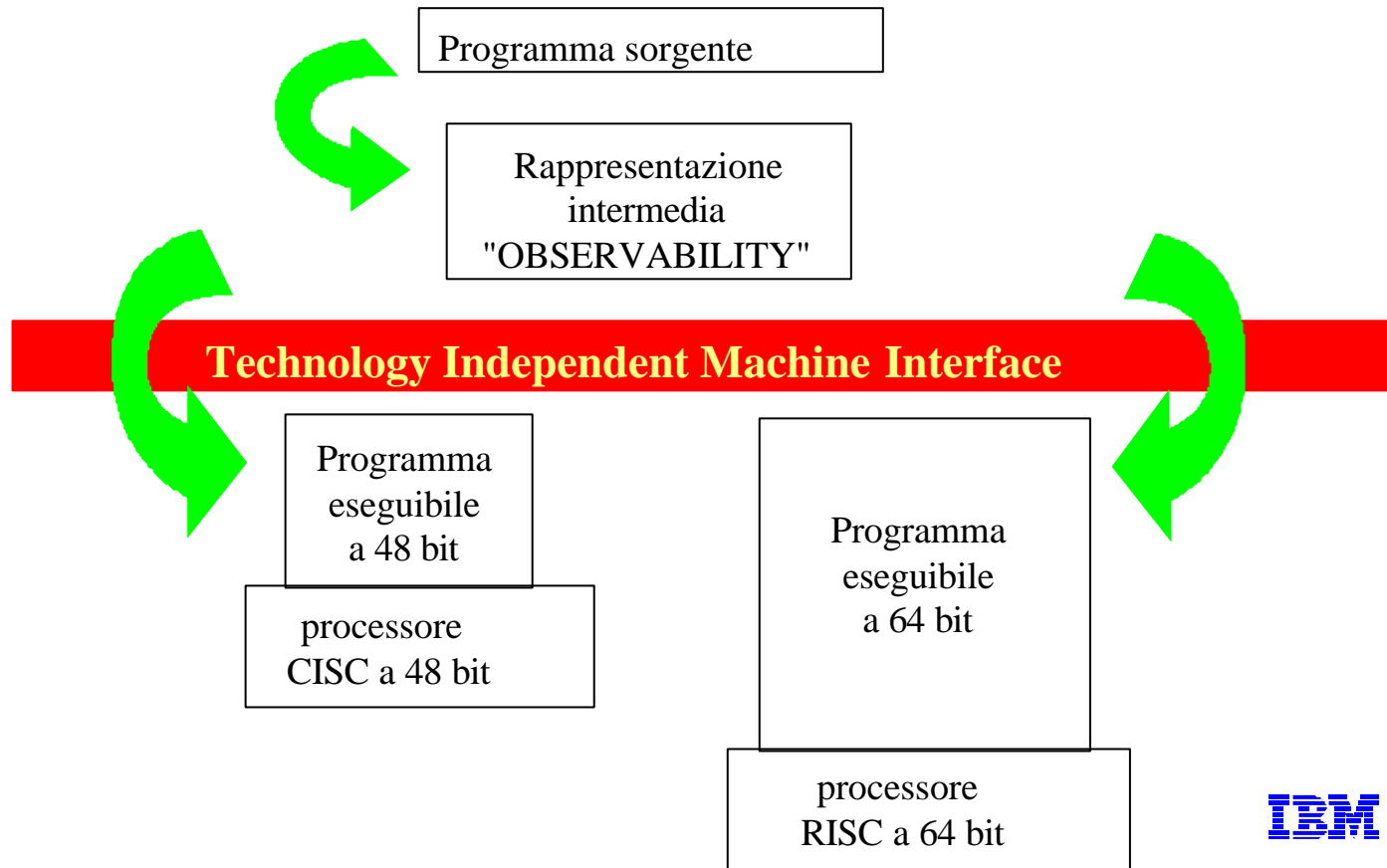
# Architetture dei sistemi IBM 370 e AS/400



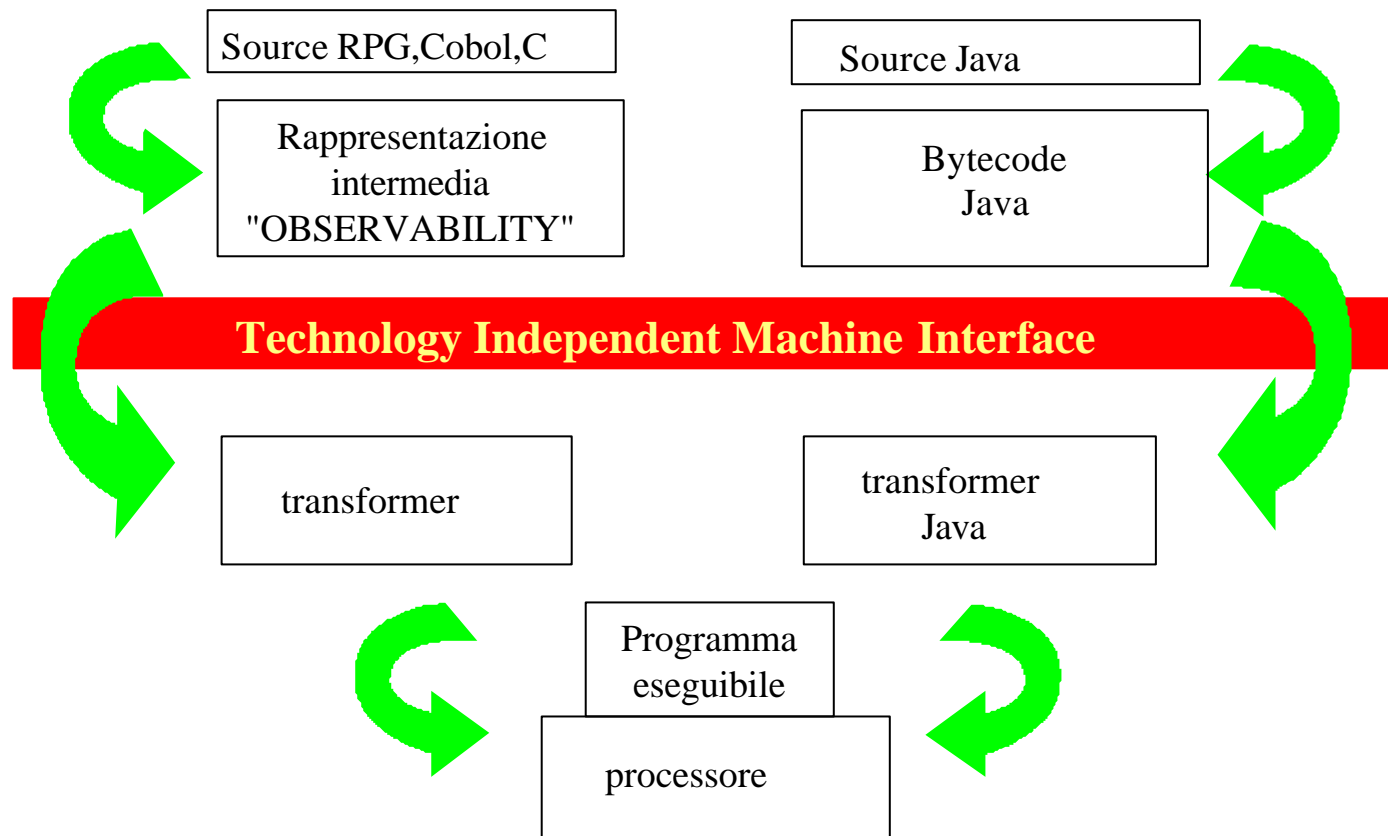
# ARCHITETTURA



# Architettura Applicativa Avanzata

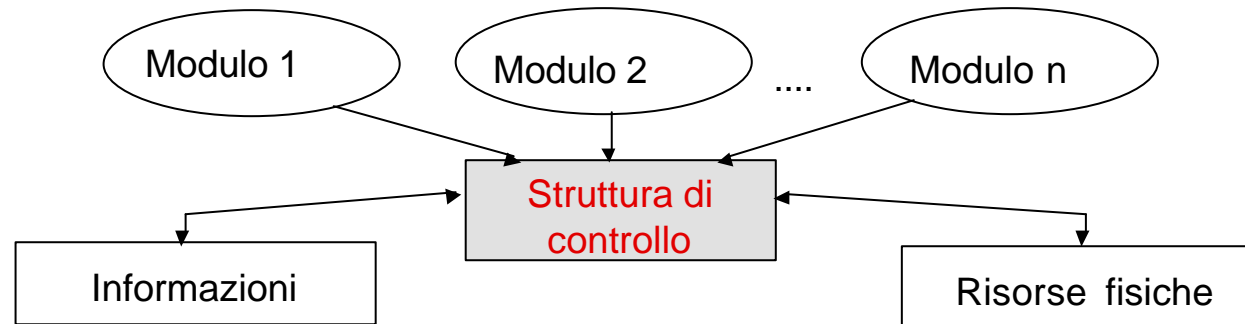


# Architettura Applicativa Avanzata e Java

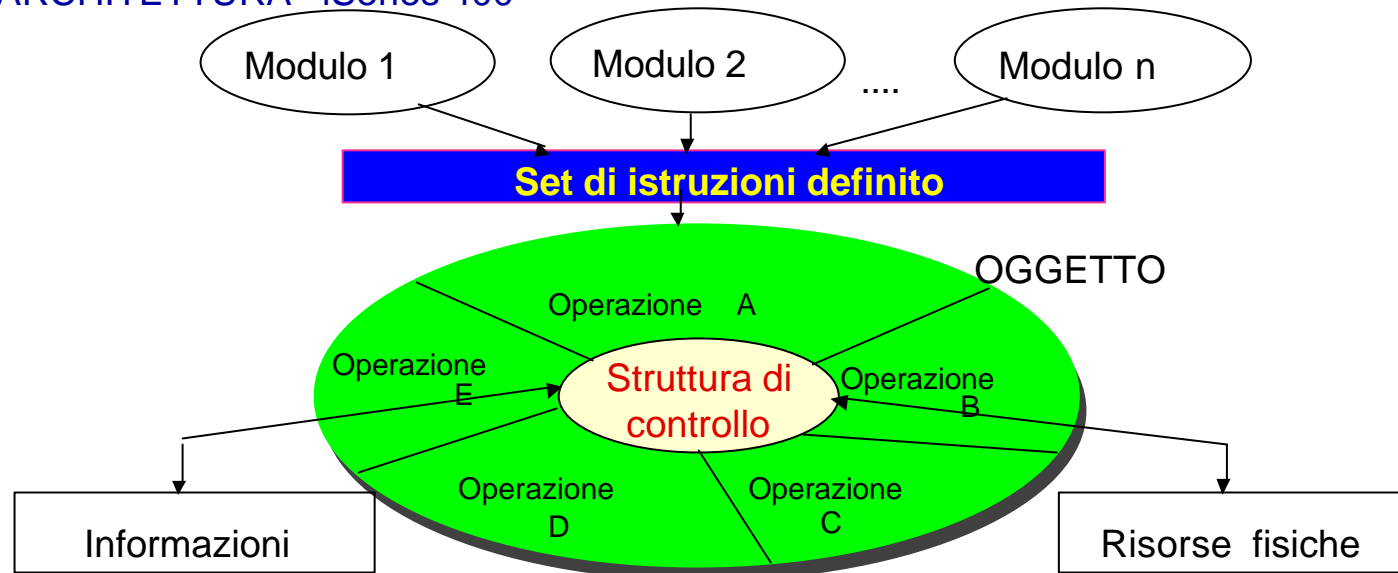


# GESTIONE PER OGGETTI

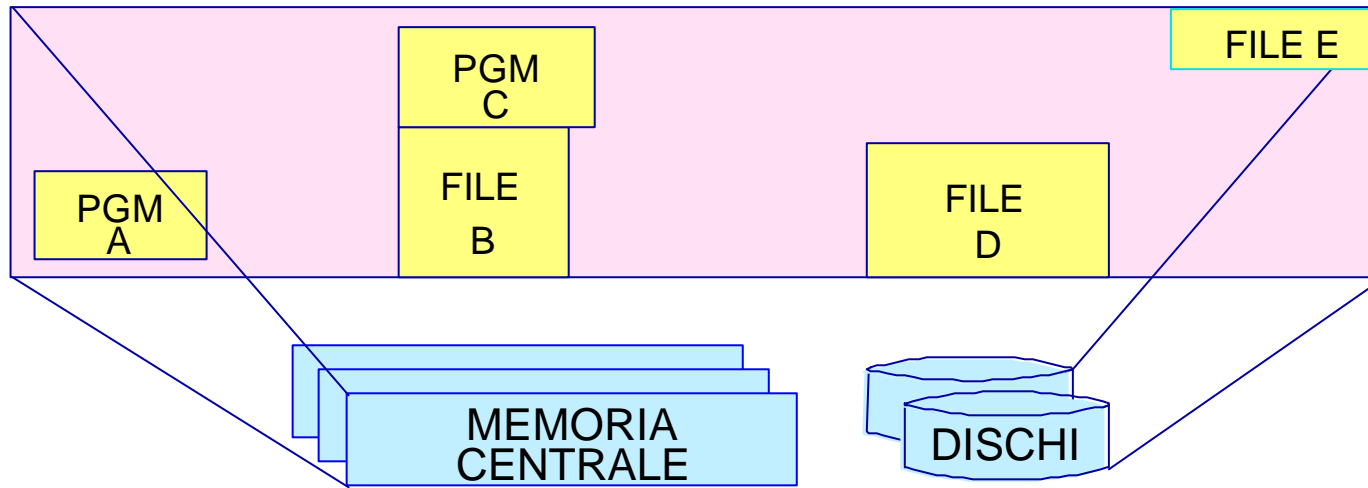
## ARCHITETTURA CONVENZIONALE



## ARCHITETTURA iSeries 400

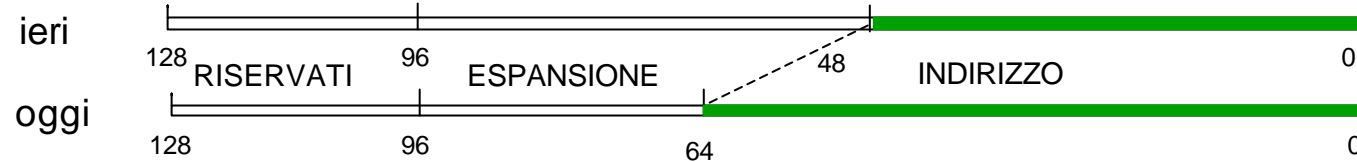


## INDIRIZZAMENTO UNIFICATO



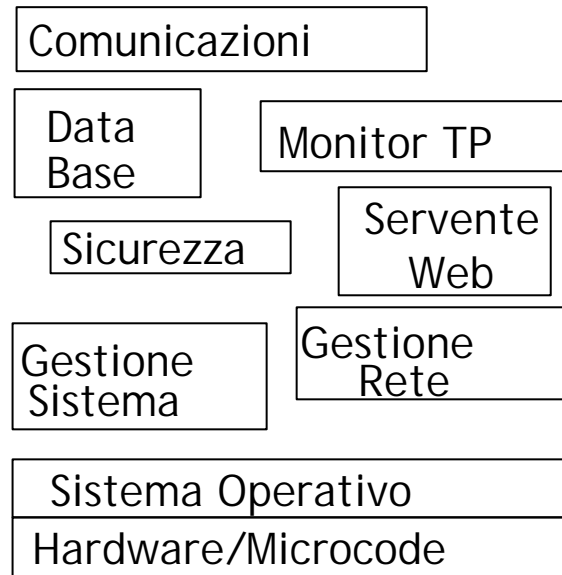
- ✓ INDIPENDENTE DALL'HARDWARE
- ✓ 281.000 MILIARDI DI CARATTERI ( $2^{48}$ ) - sistemi CISC
- ✓ 18 MILIARDI di MILIARDI di CARATTERI ( $2^{64}$ ) - sistemi RISC

### POINTER



## INTEGRAZIONE DEI COMPONENTI

UNIX, MVS



- Scelta dei componenti
- Portabilità applicazioni
- > gradi di libertà
- > costi

iSeries 400



- Rapidità di installazione
- Semplicità di gestione
- < gradi di libertà
- < costi

## Control Language (CL)

- **Orientato agli oggetti**  
comandi

XXX    YYY    ZZZ

che cosa fare  
(AZIONE)

su che cosa  
(AZIONE)

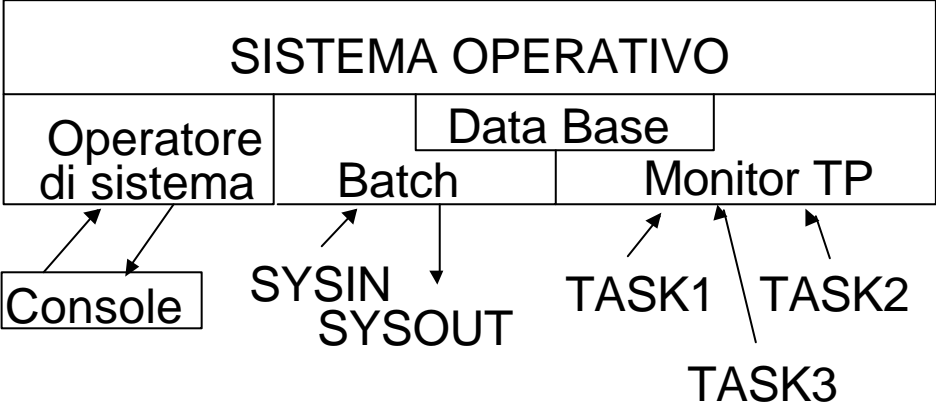
es: DSPOBJD  
CRTCLPGM  
CHGSYSVAL

- **Immisione interattiva (con parametri)**
- **Compilati in programma**
  - uso variabili
  - operazioni aritmetiche e logiche
  - richiamo programmi
  - richiamo da programmi

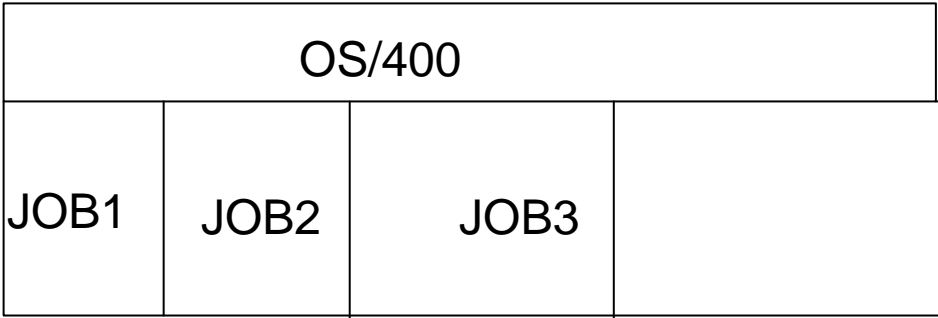
## PROGRAMMAZIONE DEL SISTEMA

# Monitor TP

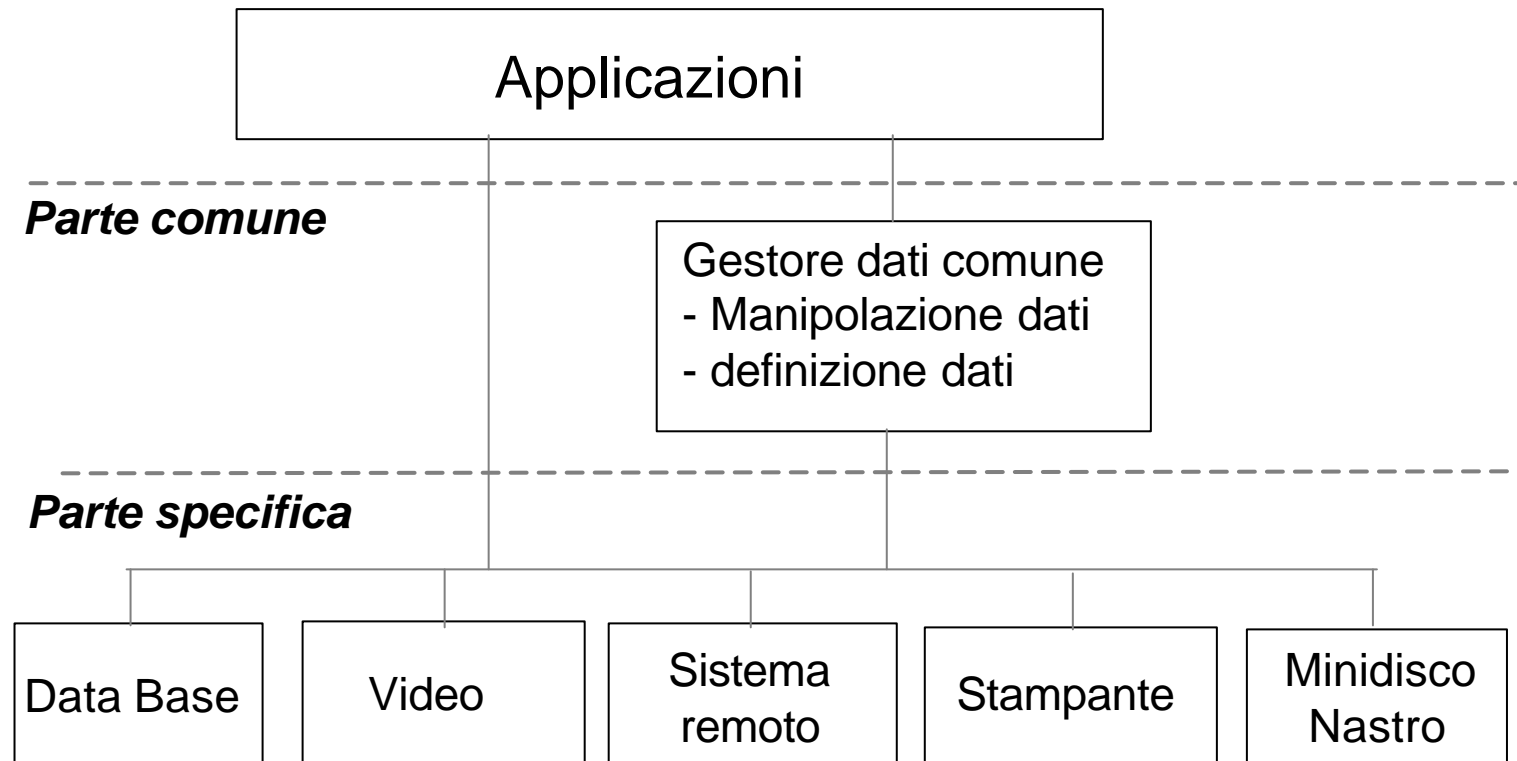
## Sistema tradizionale



## Sistema iSeries e AS/400



# Gestione integrata dei dati



Vantaggi



- ✓ Interfaccia integrata
- ✓ Semplicità - produttività

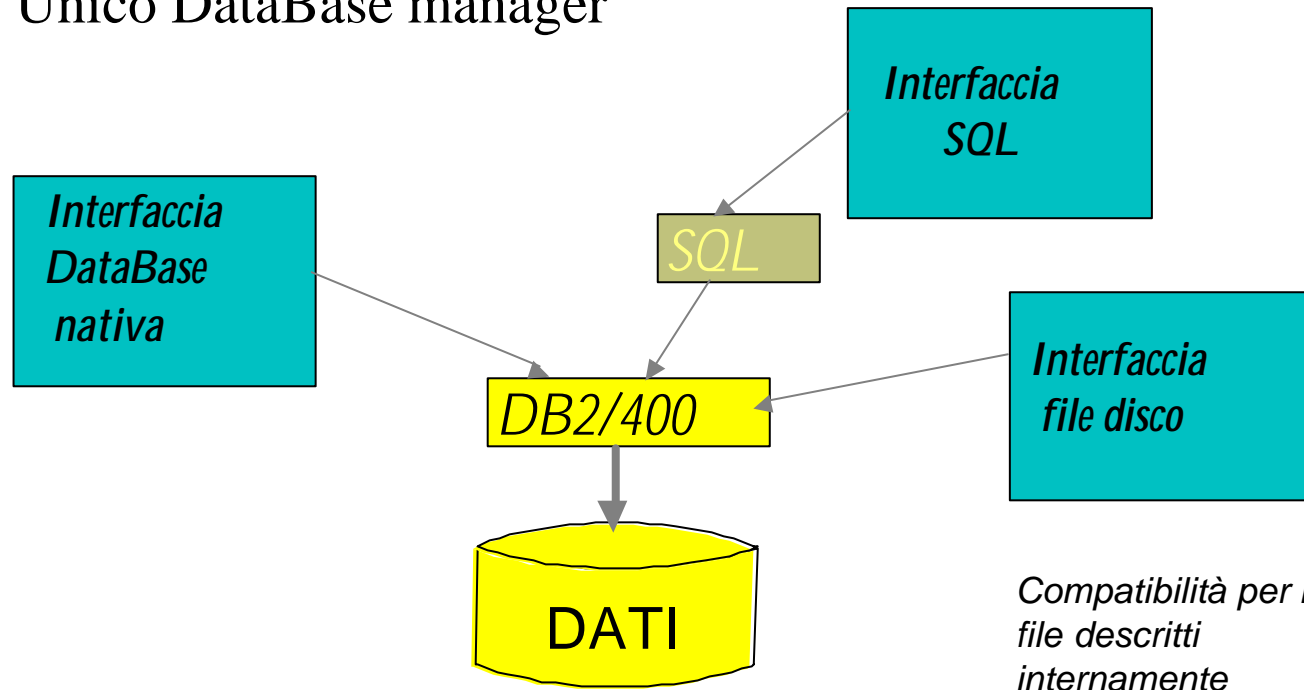
## Data Base relazionale integrato

<i>Bianchi</i>	<i>F</i>	<i>Stabilimento</i>	<i>00341</i>	<i>2.350.000</i>
<i>Rossi</i>	<i>M</i>	<i>Sede</i>	<i>16161</i>	<i>3.200.000</i>
<i>Neri</i>	<i>M</i>	<i>Filiale Roma</i>	<i>02403</i>	<i>1.200.000</i>
<i>Verdi</i>	<i>F</i>	<i>Filiale Torino</i>	<i>03861</i>	<i>4.800.000</i>
<i>Bianchini</i>	<i>M</i>	<i>Stabilimento</i>	<i>10841</i>	<i>3.800.000</i>
<i>Rossini</i>	<i>F</i>	<i>Sede</i>	<i>03304</i>	<i>2.530.000</i>

**File fisici = contenitori di dati (tables)**

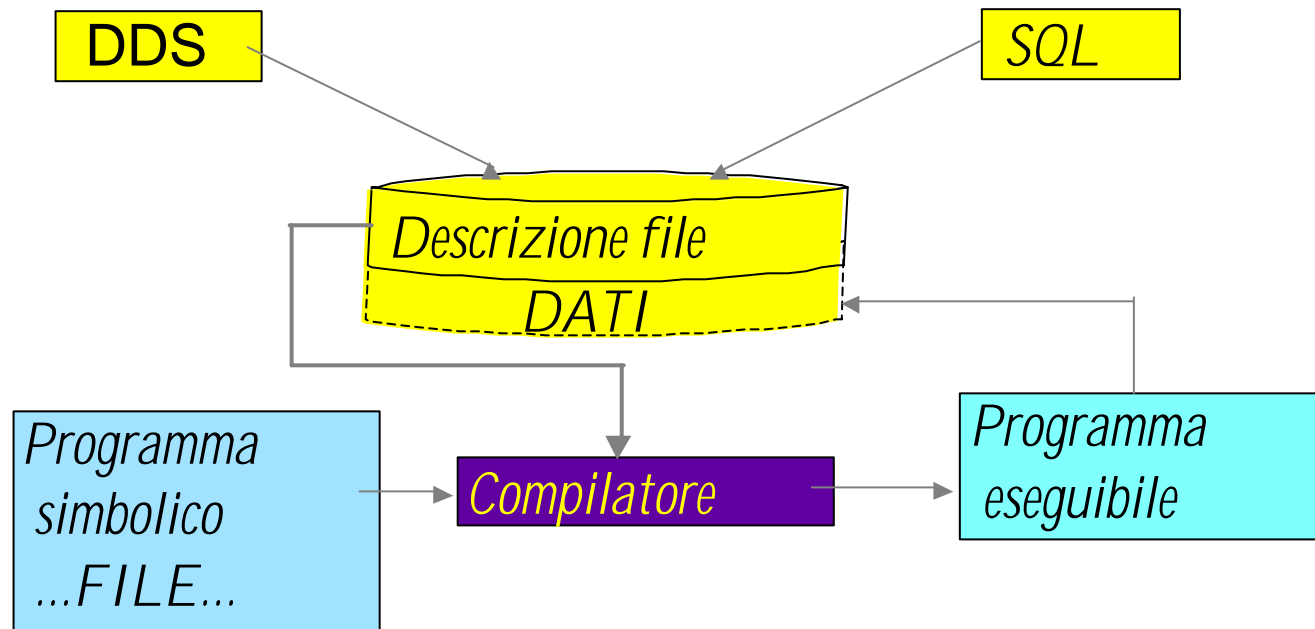
**File logici = relazioni alternative (logical view e altro)**

## Unico DataBase manager


















- *Dati non duplicati*
- *Diverse interfacce*
- *Funzioni realizzate via microcode*

## *Descrizione dati esterna ai programmi*



- *Standardizzazione nomi campi*
- *Indipendenza dati/programmi*
- *Modificabilità*

# DB2 Universal Database: LOBs e DataLinks

SOLD	ONHAND	RATING	ARTIST	TITLE	COVER	VIDEO	MUSIC	SCRIPT	DataLink
234		59PG-13	Arnold	The Exterminator					<a href="http://x.x.x">http://x.x.x</a>
13		45R	Kevin	Dancing with Bulls					<a href="http://x.x.x">http://x.x.x</a>
1295		209G	Glenn	101 Doll Imitations					<a href="http://x.x.x">http://x.x.x</a>
379		112G	Buzz	Toy Glory					

*LOBs nel DataBase*

*Pointer a oggetti esterni*

- Supporto per applicazioni multimediali in rete
- Tipi di dati per memorizzare oggetti molto grandi
- Integrità e controllo da parte di OS/400

## DB2/400 nuovi supporti per business intelligence:

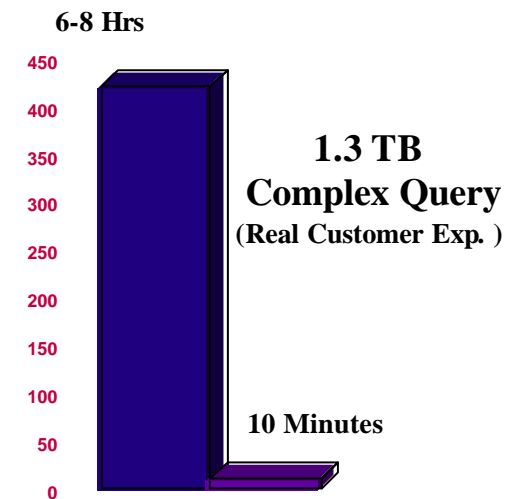
### *Encoded Vector Indexing*

#### **Prestazioni**

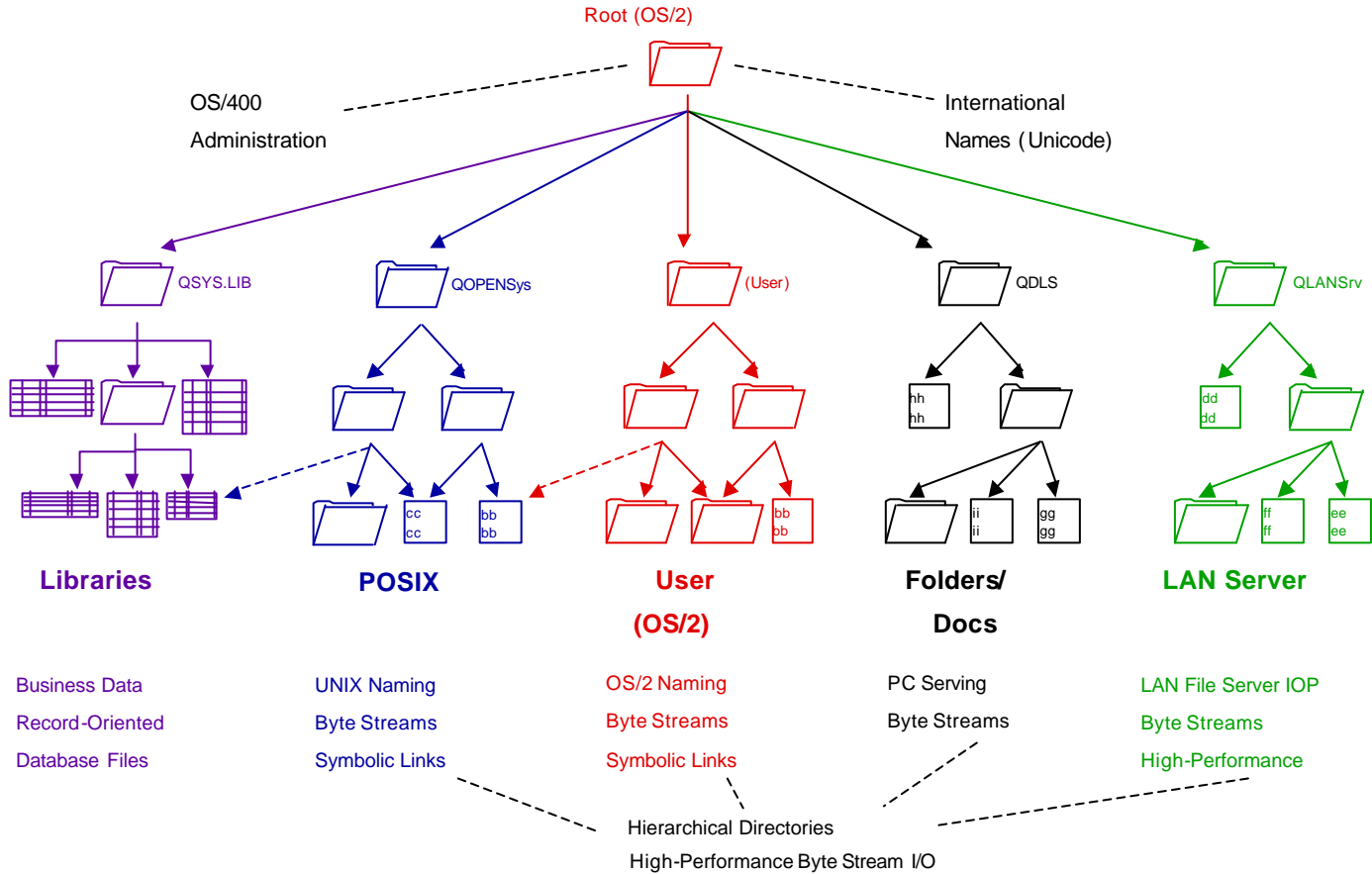
- Nuova tecnologia basata su BitMap (Encoded Vector Indexes)
- Statistiche accurate per l'ottimizzazione dei query
- Primo DB a svilupparli
- Tecnologia brevettata IBM

#### **Benefici**

- 1/16 delle dimensioni dei precedenti indici
- Costruzione fino a 10 volte più rapida
- Più indici residenti in memoria
- Ottimizzazione dei query più rapida



# Integrated File System Structure



## IBM DB2 UDB for iSeries (V5R2)

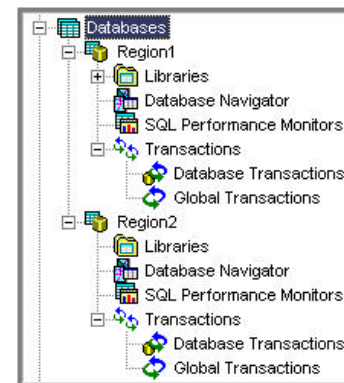
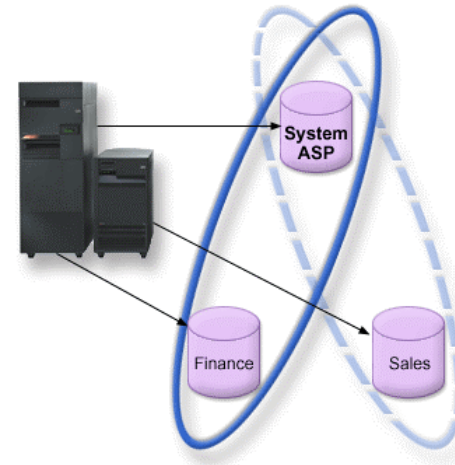
### Independent ASP

**Consolidamento dati su un unico server con nomi diversi**

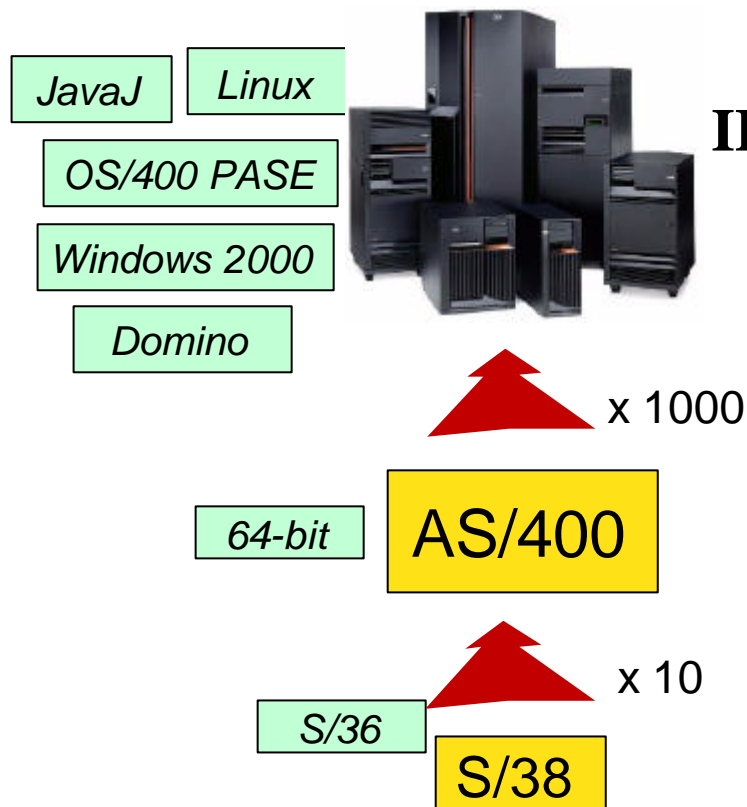
- DataBase indipendenti, librerie di sistema, nomi di libreria

### Supporto di standard aperti

- miglioramenti SQL
- Java Transaction API (JTA)
- X/Open Distributed Transaction Processing (XA-DTP)
- Compatibilità con la famiglia DB2 UDB



# IBM ❁ iSeries



## IBM @server iSeries

29 aprile 2002

37.400 CPW  
256 GB  
72 TB

21 giugno 1988

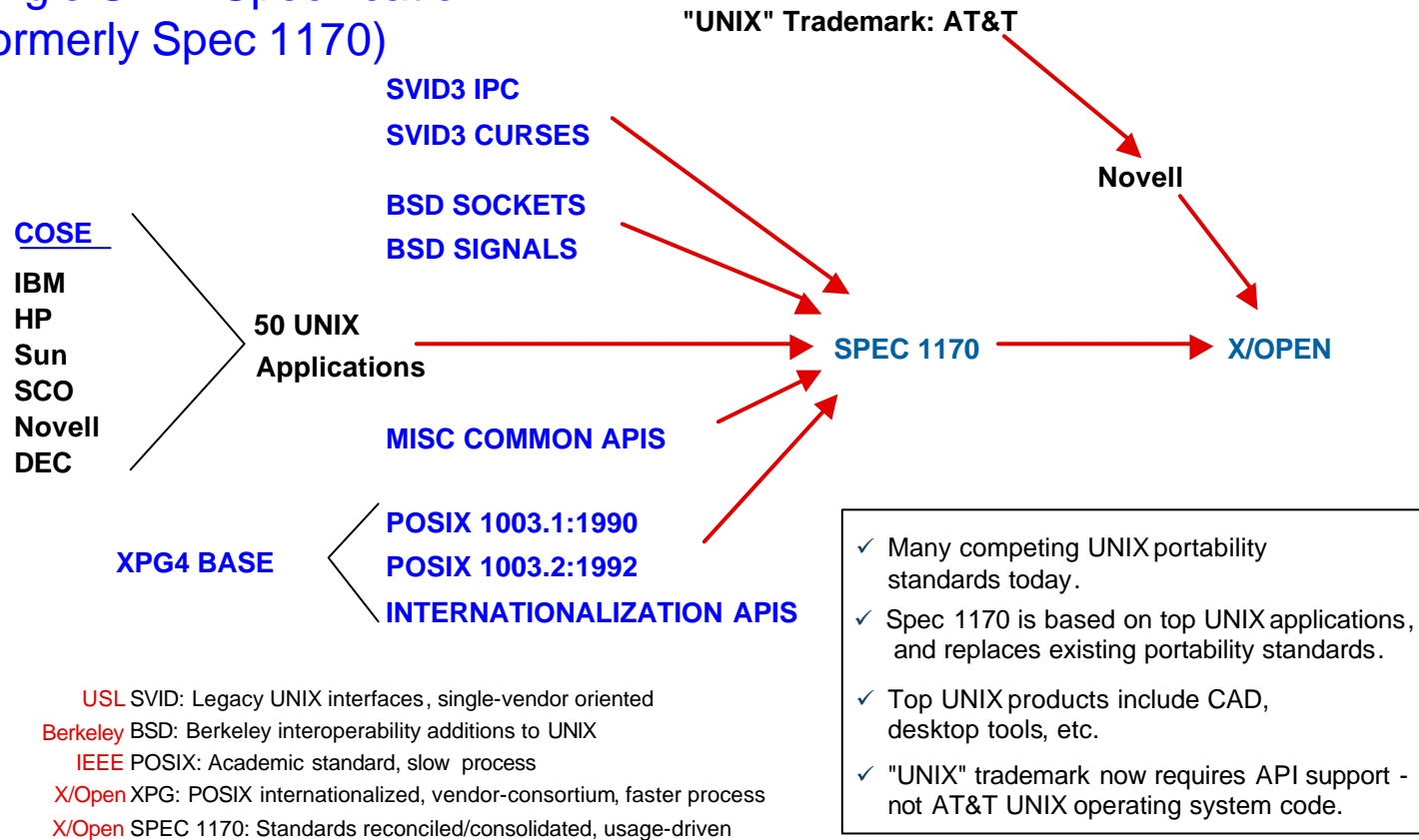
15 CPW  
96 MB  
27,3 GB

24 ottobre 1978

1,7 CPW  
1,5 MB  
387 MB

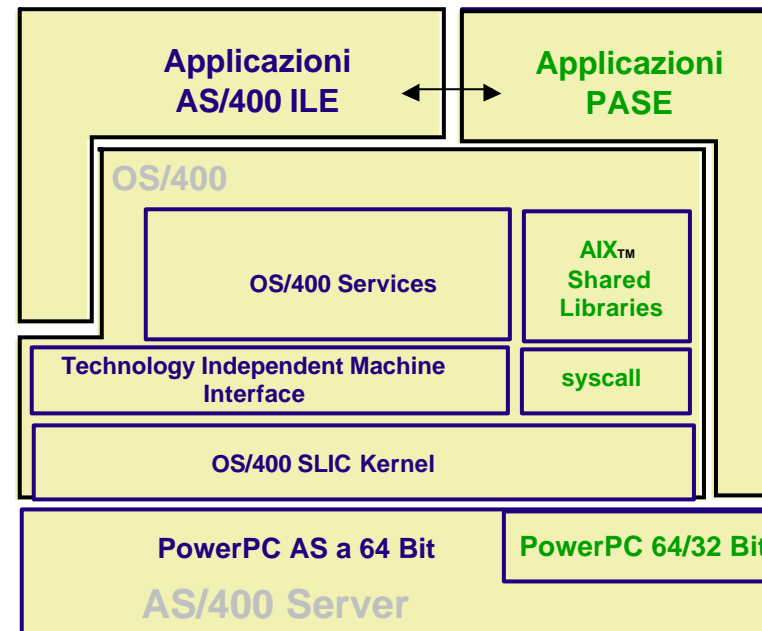
# OS/400 Sistema Aperto

## Single UNIX Specification (formerly Spec 1170)

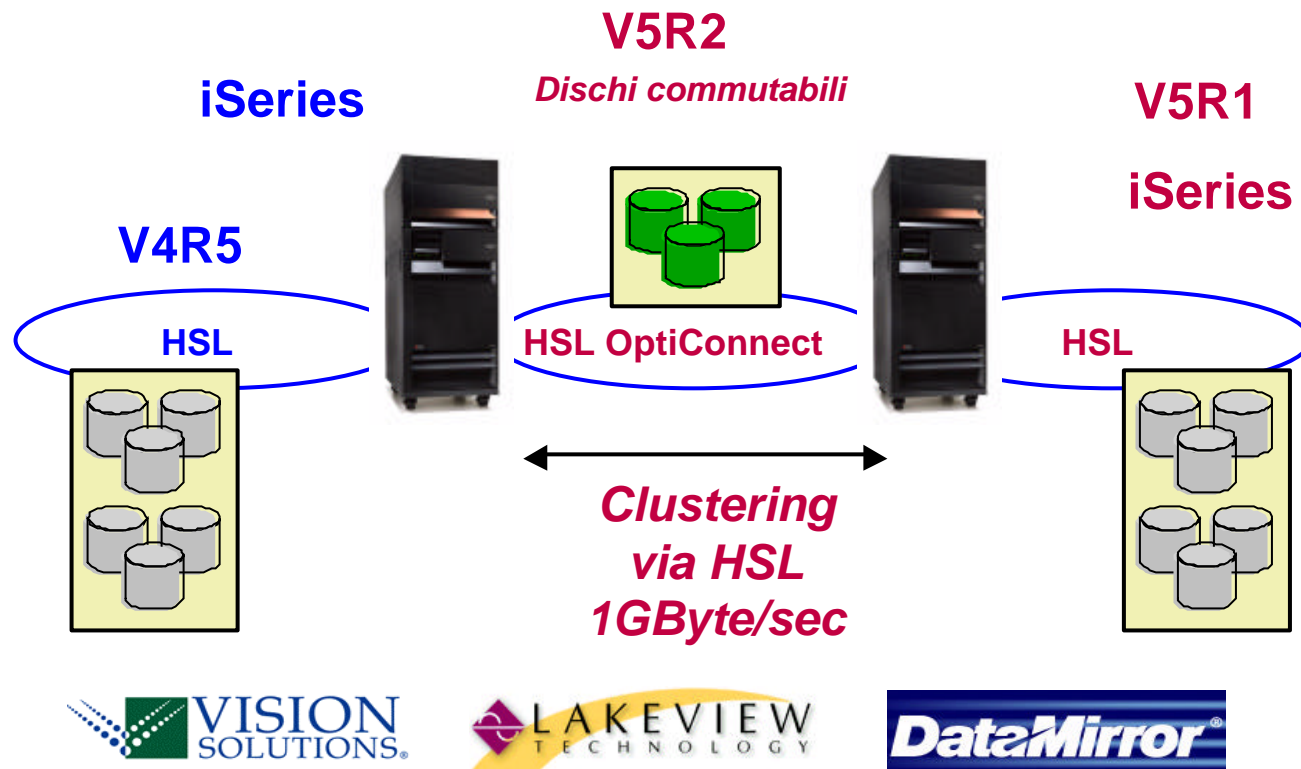


# PASE (Portable Application Solution Environment)

- Runtime integrato in OS/400<sup>7</sup> per il porting di applicazioni UNIX
  - Usa un subset delle librerie AIX<sup>TM</sup>
  - Non è un sistema operativo o un ambiente di emulazione
- Capacità del PowerPC di commutare fra ambienti
- Applicazioni
  - Integrazione con il file system AS/400<sup>R</sup> e il DB2 Universal Database for AS/400
  - Chiamate a programmi Java<sup>TM</sup> e ILE
  - Utilizzo di tutte le caratteristiche dell'ambiente operativo AS/400



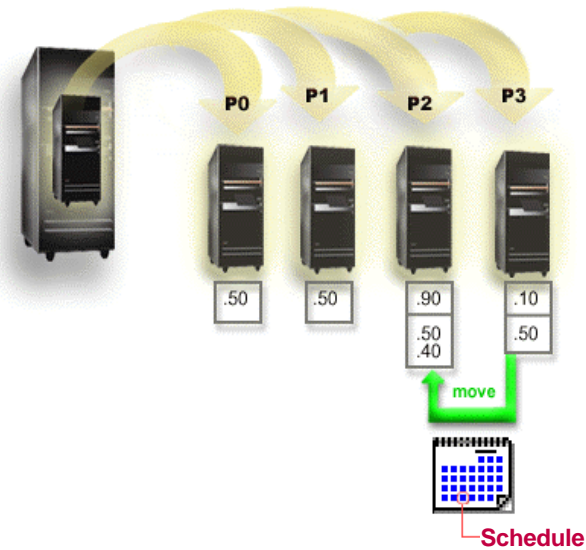
## Opticonnect e clustering via High Speed Link



# Logical Partitioning di classe mainframe

Tecnologia evoluta, flessibile

- Spostamento dinamico delle risorse
- Processori condivisi
- Fino a 32 partizioni
- Virtual Ethernet
- LPAR anche sui modelli 270 SStar
- Linux in una partizione
- Primo server del mercato piccolo e intermedio con partitioning
- Granularità
  - 0,10 + 0,01 processore - 1 MB memoria
- Sofisticati tool grafici di gestione via Navigatore V5R2
- Gestione multi-partizione



## Linux su iSeries

iSeries fornisce a Linux l'espandibilità e la robustezza

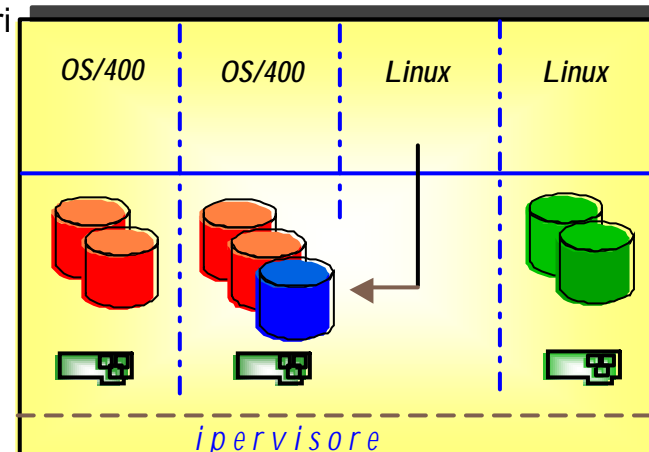
Kernel di Linux in una logical partition

Supporto dei processori condivisi

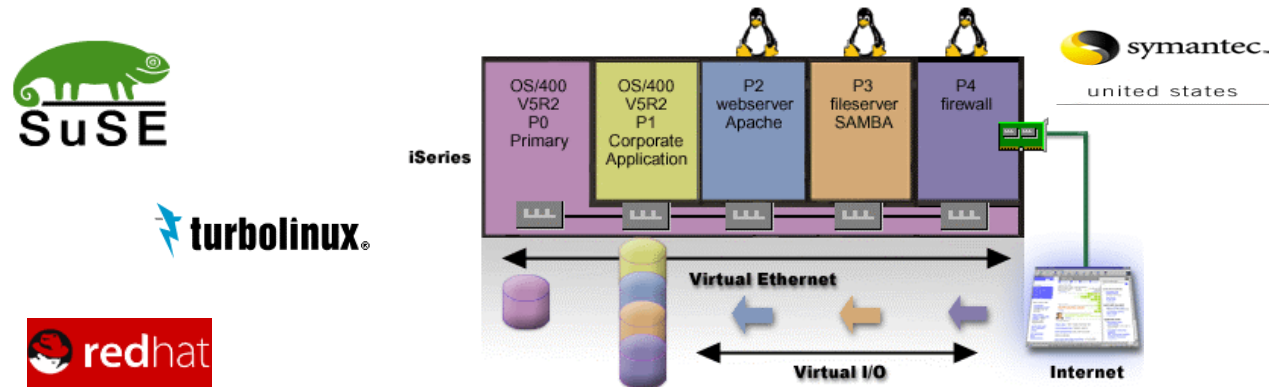
- Partizionamento anche di monoprocessori su SStar

Allocazione flessibile delle risorse

- I/O dedicato e virtuale



## Server Consolidation con Linux (V5R2)



- Gestione dinamica dei processori anche per le partizioni Linux
- Linux per iSeries Linux supporta il kernel a 64 bit
- Enterprise Firewall della Symantec pianificato per iSeries
- IBM DB2 Universal Database e WebSphere Application Server\*

\*Statement of Direction: This presentation contains IBM plans and directions. Such plans are subject to change without notice.

# Integrazione di Windows2000 con iSeries

## Blades

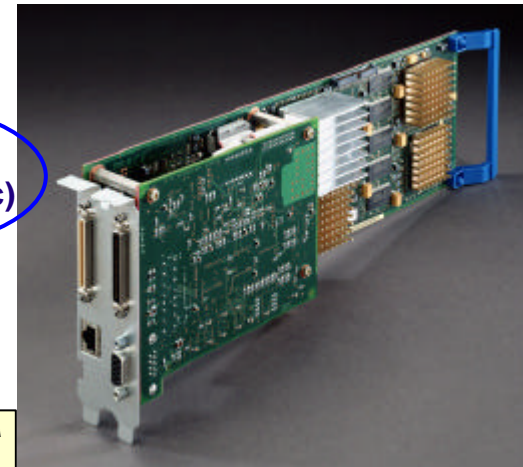
Modello iSeries	N. di IXS
270	3
820	12
830	28
840	32
890	32

- Pentium III 1 GHz
- Xeon 1,6 GHz
- Hyperthreading technology
- Ethernet 10/100 integrato
- da 1 fino a 4 GB RAM
- fino a 2 TB disco

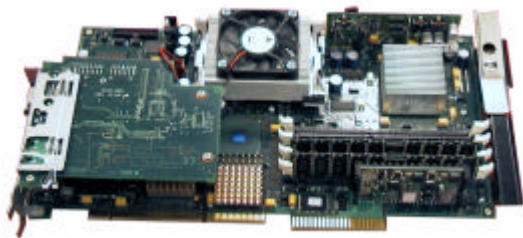


## Integrated xSeries Adapter

**HSL**  
(1 Gigabyte/sec)



## Integrated xSeries Server

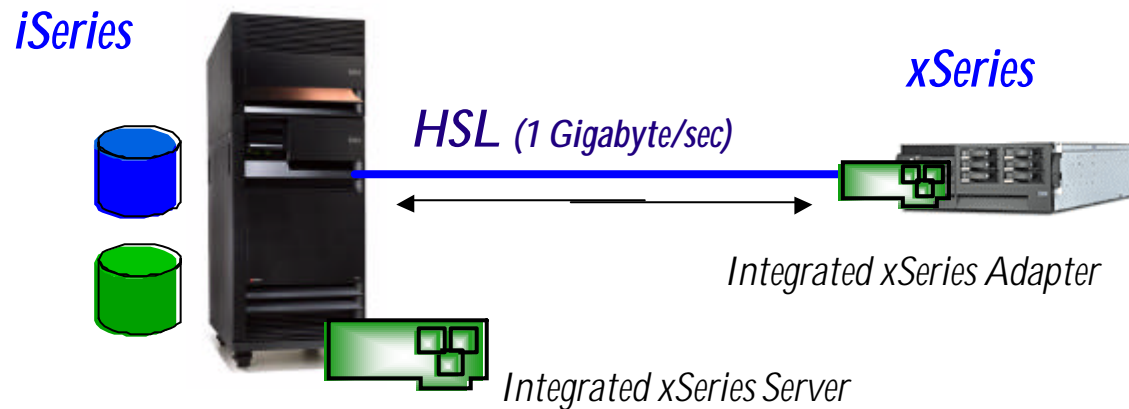


Modello iSeries	N. di IXA
270	2
820	8
830	16
840	32
890	32

## Collega xSeries Server

- fino a 4 processori
- x235, x255, x360, x440
- x250, x350, Netfinity 7100 e 7600

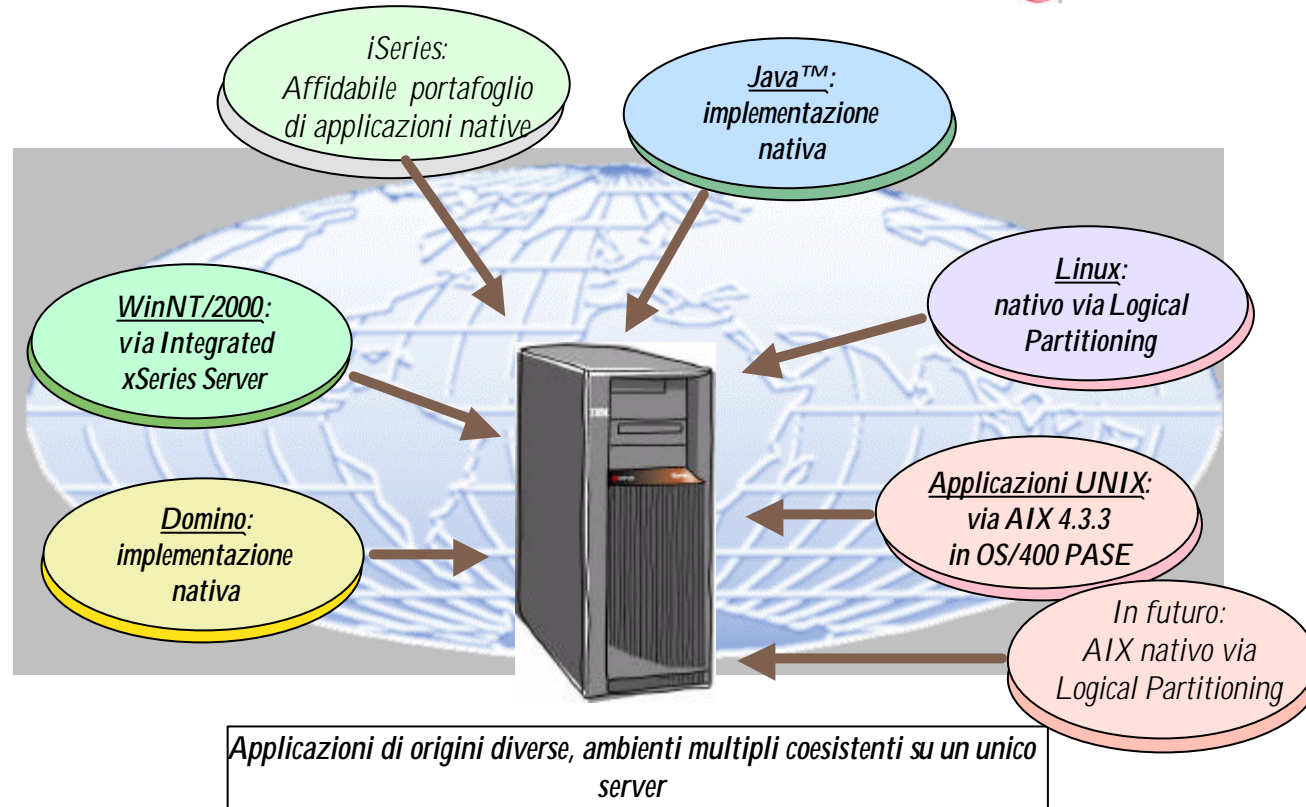
## Integrazione di Windows2000 con iSeries



- Condivisione dischi, nastri, DVD e CD-ROM di iSeries
- Gestione dei server e delle applicazioni Windows da iSeries
- Storage Area Network (SAN) integrata per server Windows

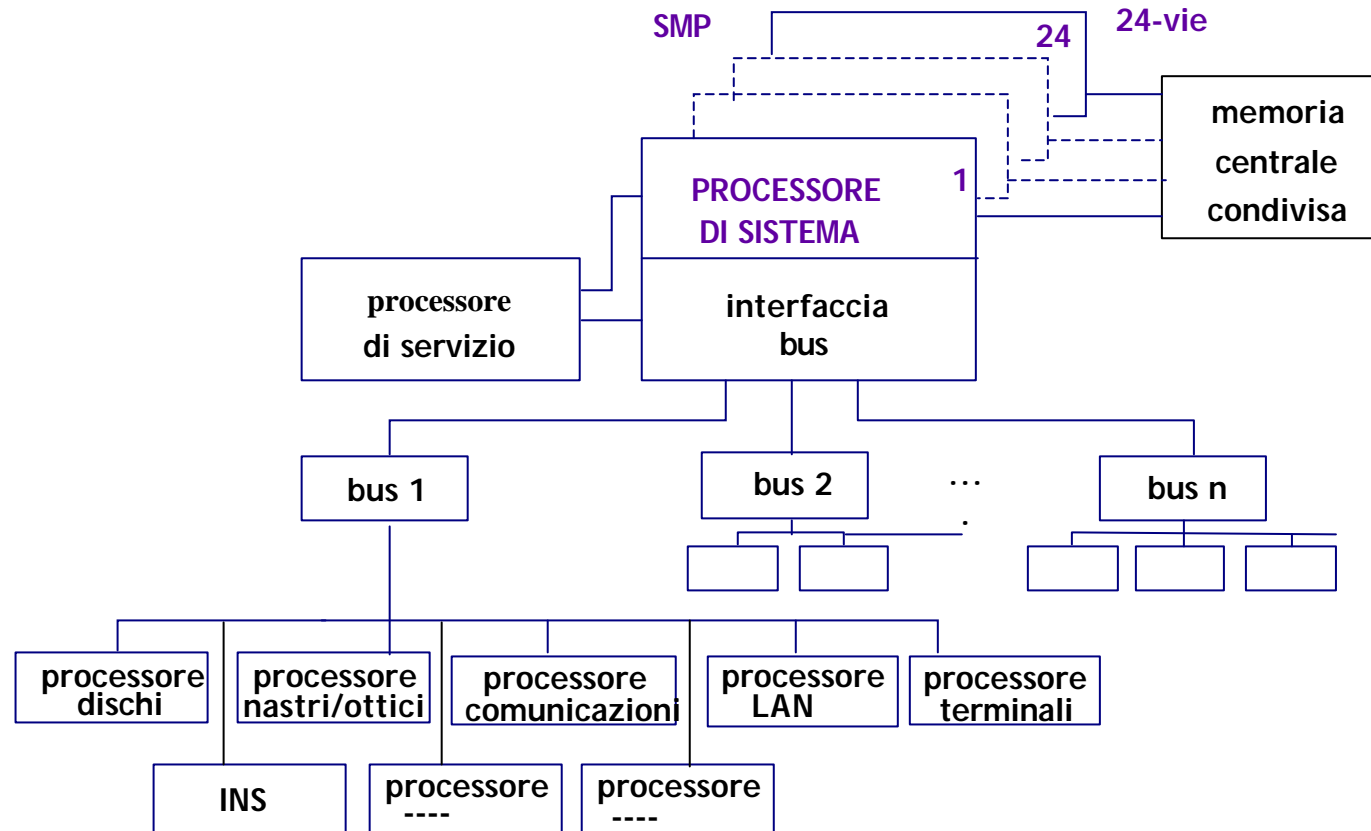
# iSeries: integrazione applicativa

IBM  server iSeries

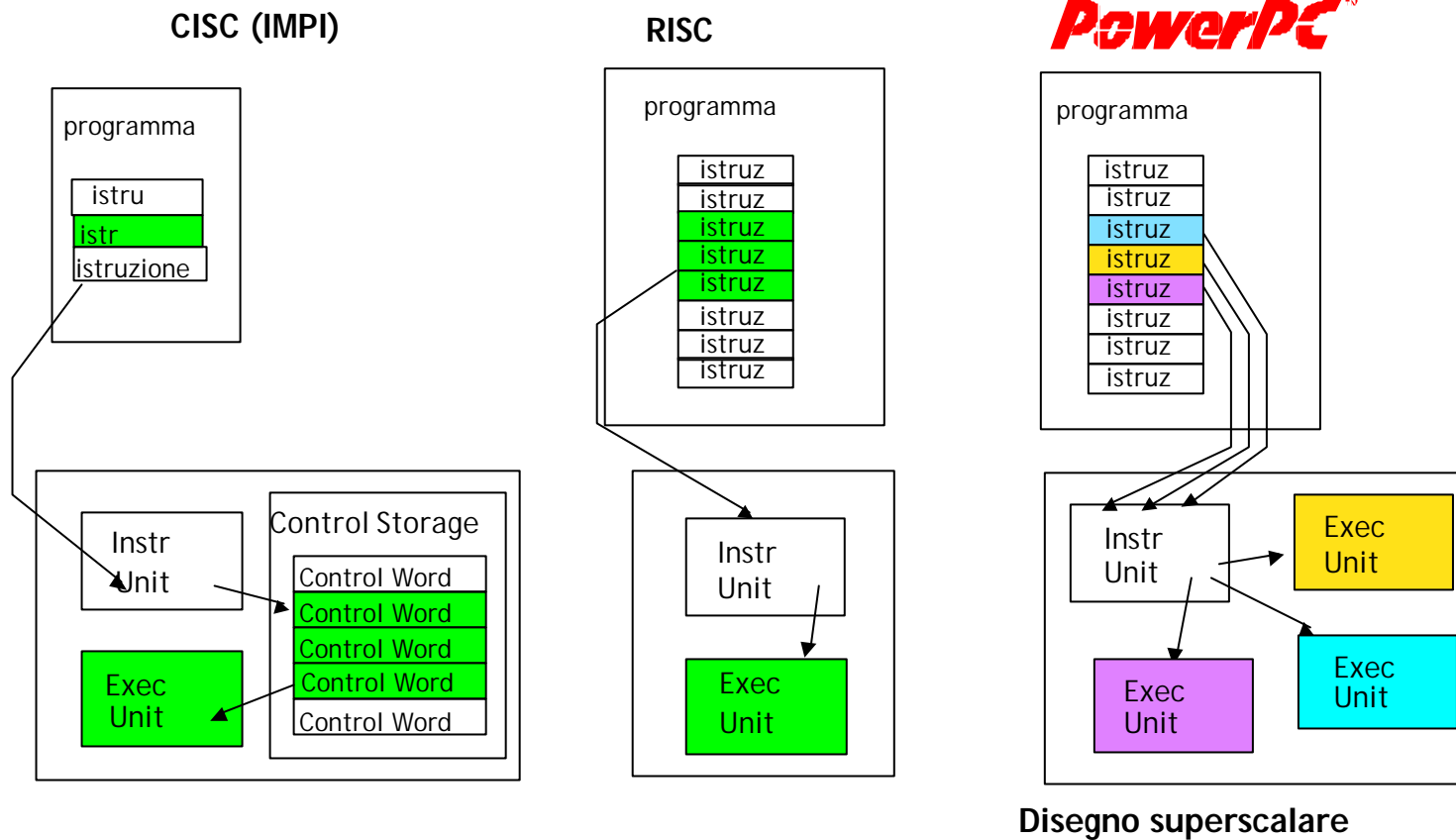


© 2001 IBM Corporation

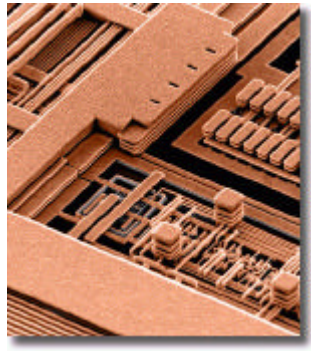
# Architettura hardware AS/400 e iSeries



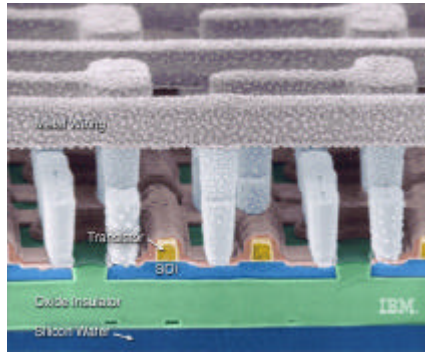
# Disegno dei Processori



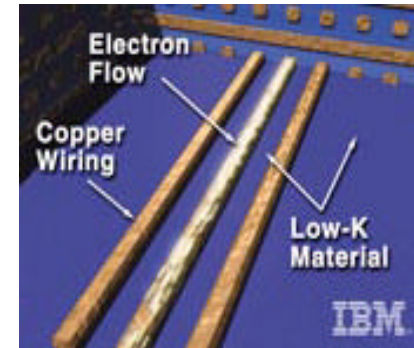
*Tecnologia d'avanguardia*



*Circuiti in Rame*

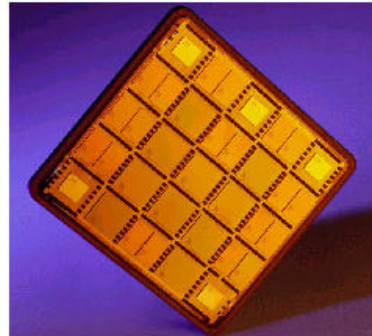


*SOI (Silicon-on-Insulator)*

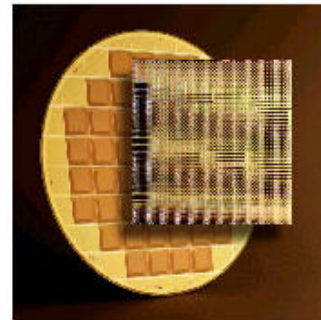


*Low-K*

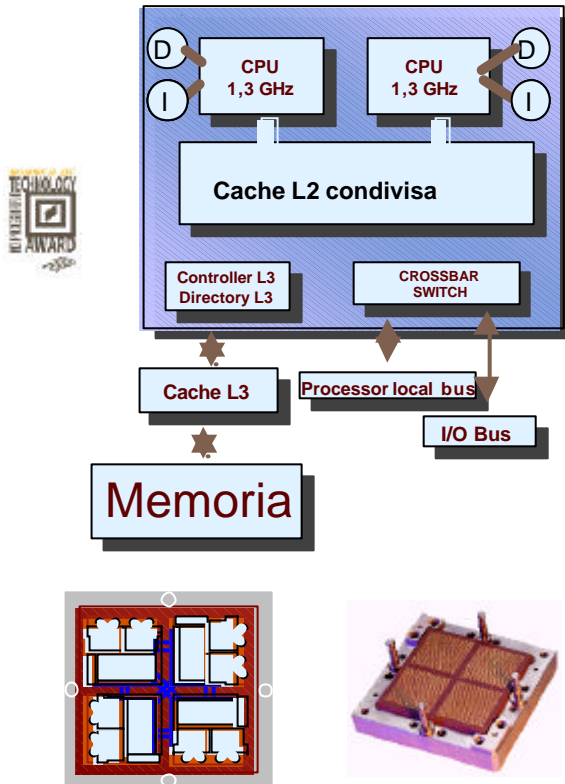
*MCM*



**Power4**

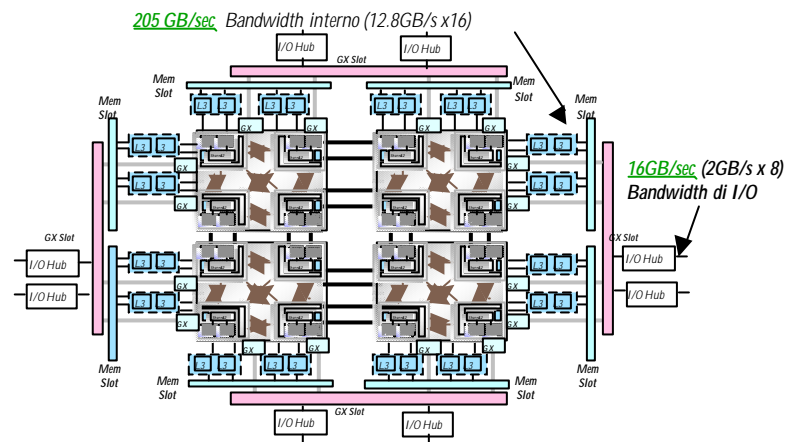


# Architettura sistemi con Power4




- 2 Processori su 1 chip
- 174 Milioni di Transistor
- 1.3 GHz
- Tecnologia  
Copper + SOI  
0.18 micron

- fino a 32 Processori
- 256 GB di memoria



## I modelli IBM iSeries dal maggio 2002

IBM  server iSeries



- | Modello 250  
| monoprocessori  
| RAM 1 GB  
| Dischi 70 GB  
| Northstar  
| 75 CPW
- | Modello 270  
| 1 - 2 vie  
| RAM 16 GB  
| Dischi 840 GB  
| Pulsar/SStar  
| 2350 CPW
- | Modello 820  
| 1 - 4 vie  
| RAM 32 GB  
| Dischi 8 TB  
| Pulsar/SStar  
| 3700 CPW
- | Modello 830  
| 2 - 8 vie  
| RAM 64 GB  
| Dischi 22 TB  
| IStar  
| 7350 CPW
- | Modello 840  
| 8 - 24 vie  
| RAM 128 GB  
| Dischi 38 TB  
| IStar/SStar  
| 20200 CPW
- | Modello 890  
| 16 - 32 vie  
| RAM 256 GB  
| Dischi 72 TB  
| Power4  
| 37400 CPW

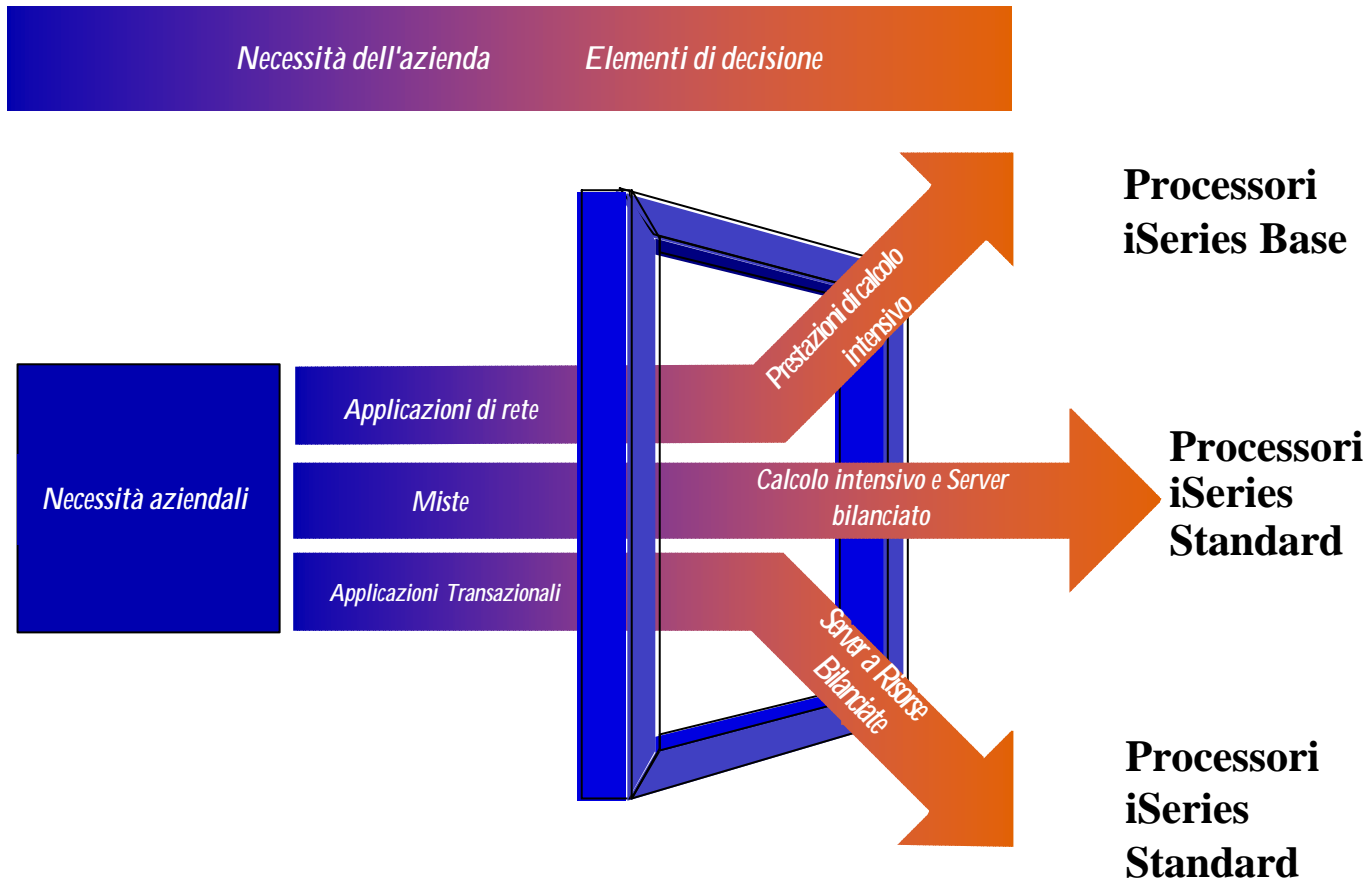
valori massimi per modello

## Applicazioni di tipo diverso

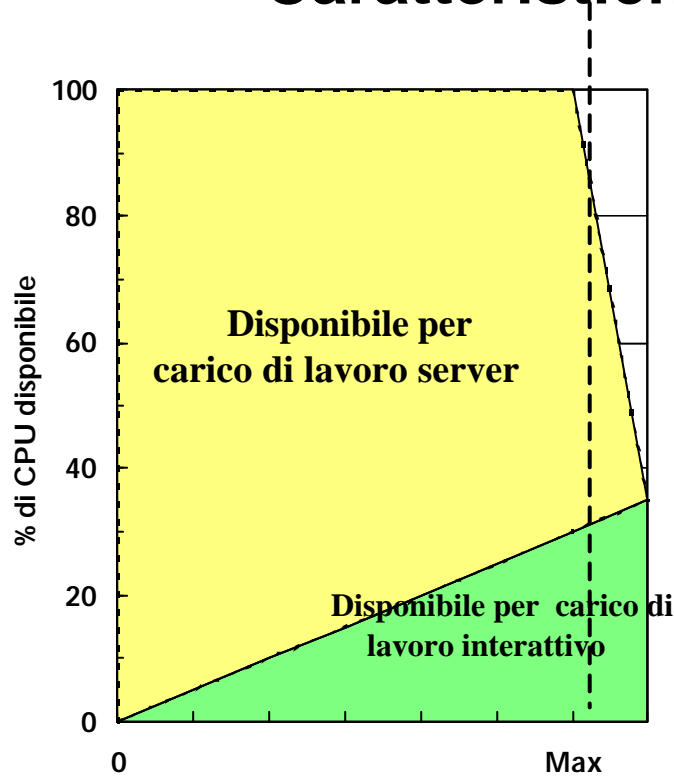
Workload	Caratteristiche	Esempi
<ul style="list-style-type: none"> <li>n Applicazioni specifiche e servente Web</li> </ul>	<ul style="list-style-type: none"> <li>n CPU intensive</li> <li>n Grande memoria</li> </ul>	<ul style="list-style-type: none"> <li>n LinuxJ</li> <li>n WebSphere</li> <li>n Domino</li> <li>n Client/Server</li> <li>n e-business</li> <li>n JavaJ</li> </ul>
<ul style="list-style-type: none"> <li>n Crescita, transazioni miste</li> </ul>	<ul style="list-style-type: none"> <li>n Risorse bilanciate (I/O, memoria, CPU)</li> <li>n 5250 data stream</li> <li>n CPU intensive</li> <li>n Grande memoria</li> </ul>	<ul style="list-style-type: none"> <li>n Appl tradizionali "green screen"</li> <li>n Windows</li> <li>n LinuxJ</li> <li>n WebSphere</li> <li>n Domino</li> <li>n JavaJ</li> </ul>
<ul style="list-style-type: none"> <li>n Transazioni 5250</li> </ul>	<ul style="list-style-type: none"> <li>n Risorse bilanciate (I/O, memoria, CPU)</li> <li>n 5250 data stream</li> </ul>	<ul style="list-style-type: none"> <li>n Appl tradizionali "green screen"</li> </ul>

*Not intended to include the complete list*

## Carichi di lavoro diversi su iSeries

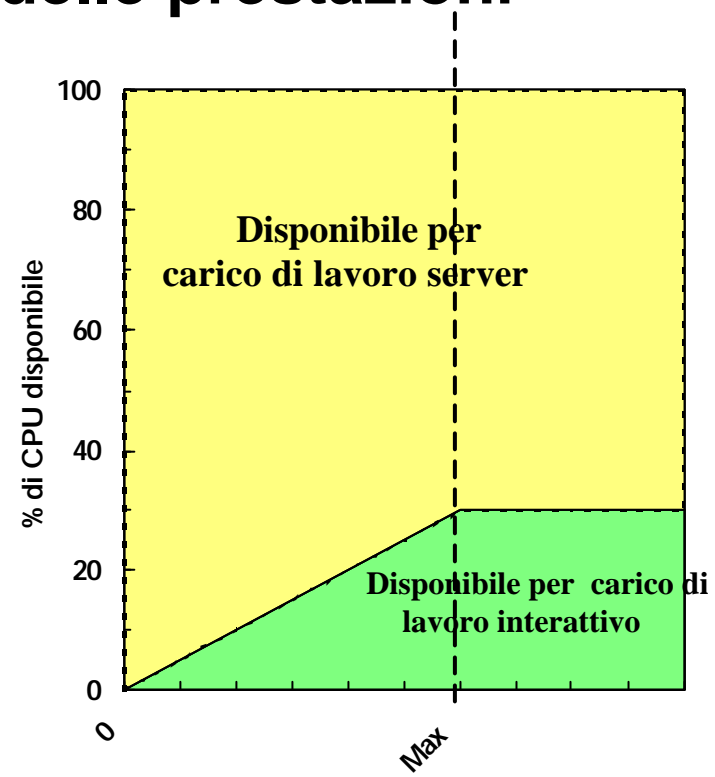


# Caratteristiche delle prestazioni



Northstar

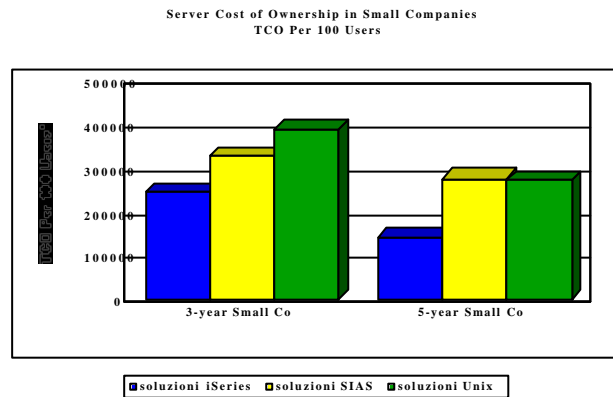
■ Interactive Workloads ■ Server Workloads



Pulsar/Istar/SStar

■ Interactive Workloads ■ Server Workloads

## Total Cost of Ownership (TCO)

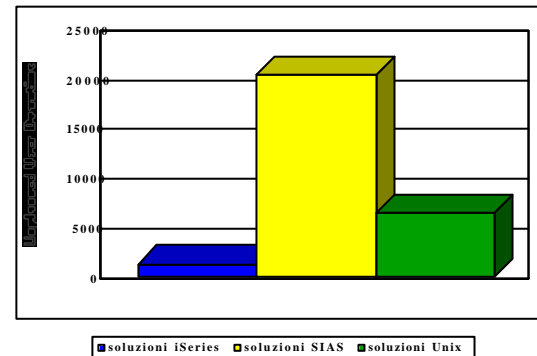


### Continuità di servizio

- soluzioni iSeries 99.98%
- soluzioni Unix 99.90%
- soluzioni SIAS 99.67%

### Staff IT per iSeries\*

- solo 2/3 rispetto a soluzioni Unix
- solo 40% rispetto a soluzioni SIAS



\* Source "Server Cost of Ownership in ERM Customer Sites: A Total Cost of Ownership (TCO) Study" IDC September 2001  
SIAS = Standard Intel Architecture Server

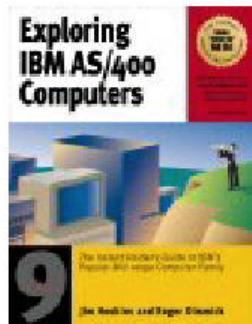
**Da AS/400..... a..... eServer iSeries**

- ✓ *Da Caratteri a Navigatore*
- ✓ *Da SNA only a Full TCP/IP*
- ✓ *Da Terminali Stupidi a PC/Browser*
- ✓ *Da Proprietario a Linux*
- ✓ *Da Gestionale a Universale*

- ✓ *Da il più Affidabile a il più Affidabile*
- ✓ *Da Impenetrabile a Impenetrabile*
- ✓ *Da Tecnologia di Punta a Tecnologia ancora più di Punta*

- n *E' il Server leader del mondo transazionale per le Piccole e Medie Imprese*
- n *Ha il costo d'uso più basso*
- n *Ha soluzioni testate, sicure, di alta affidabilità*
  - è *Non ha mai preso un virus, nè si conoscono hackers riusciti*
- n *Espande il gestionale a internet per l'e-business*
  - è *Un unico servente iSeries fa girare tutte le applicazioni, da quelle di base all'e-business*
  - è *30,000 applicazioni a 64 bit*
- n *E' un server universale, progettato per il futuro*
  - è *Tecnologia di livello superiore*
  - è *OS/400 + JAVA, Windows 2000, AIX, Linux, Domino*
- n *E' funzionale per gli Application Service Provider*
- n *E' costruito per il Business*

Traduzione da:



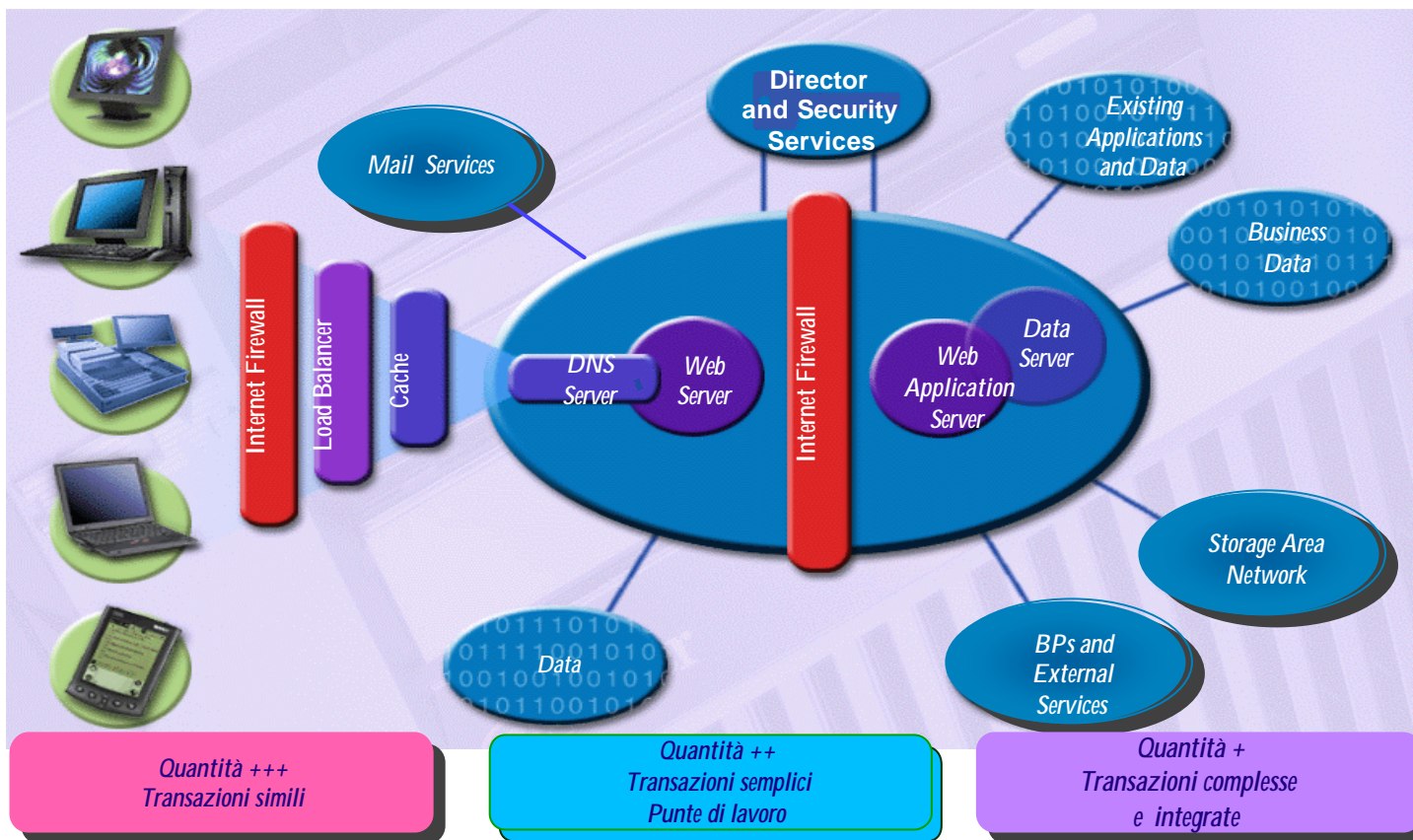
<http://www.jacksonlibri.it/servizi/Libri/schede/2386/>



# IBM server



## Infrastruttura per l'e-business



## *Tecnologia d'avanguardia per prestazioni eccezionali*



### **zSeries**

- I server più affidabili per le transazioni di dati mission-critical

### **pSeries**

- I server UNIX più veloci e caratterizzati dalle più avanzate tecnologie



### **iSeries**

- I server integrati a elevate prestazioni per aziende di piccole e medie dimensioni



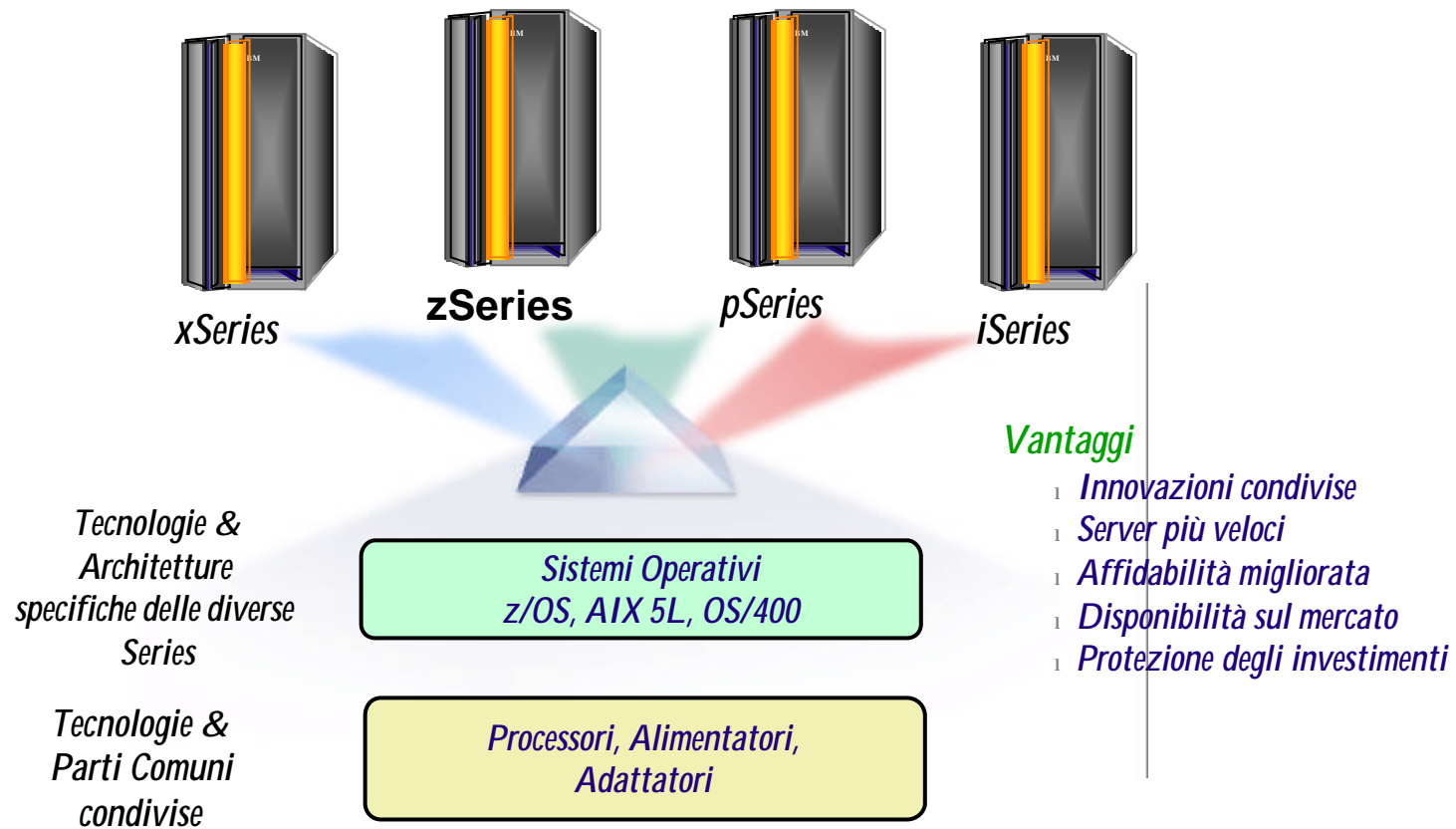
### **xSeries**

- I server entry, basati su processore Intel

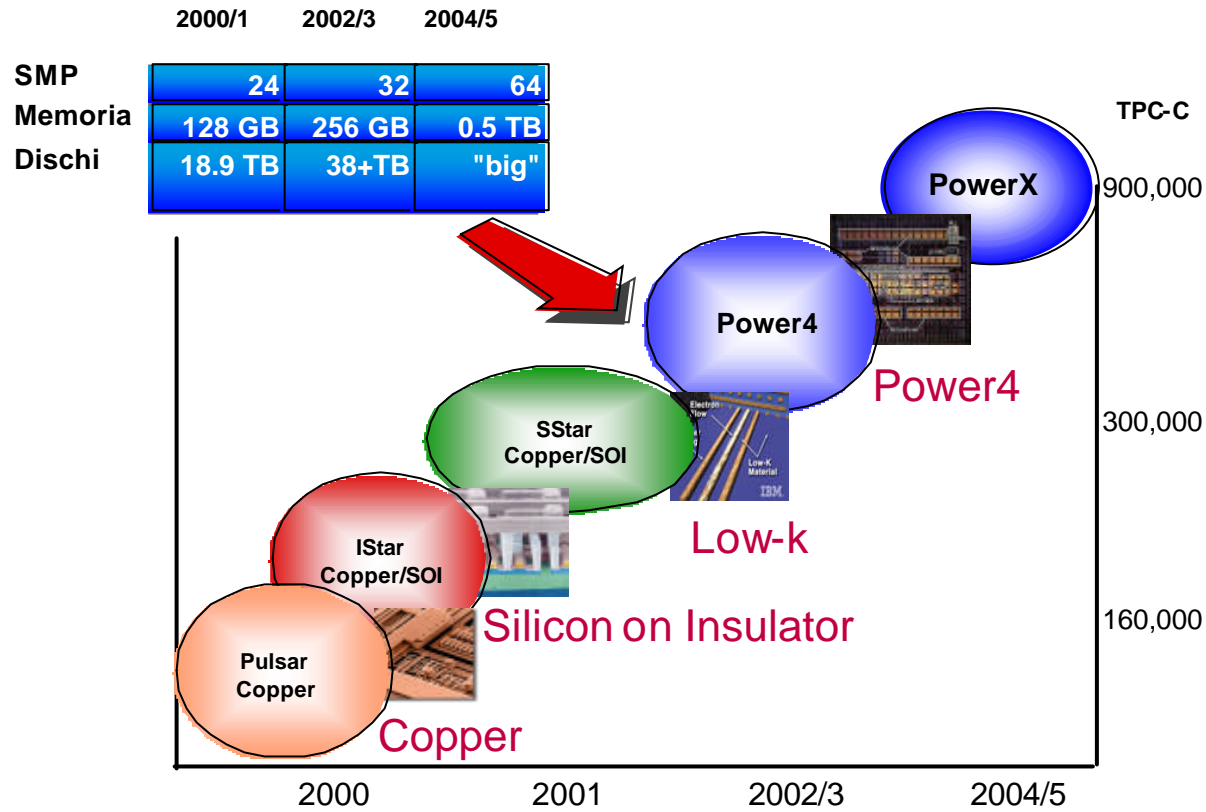


- *z = Zero Downtime*
- *p = performance*
- *i = integrated*
- *x = x-Architecture*

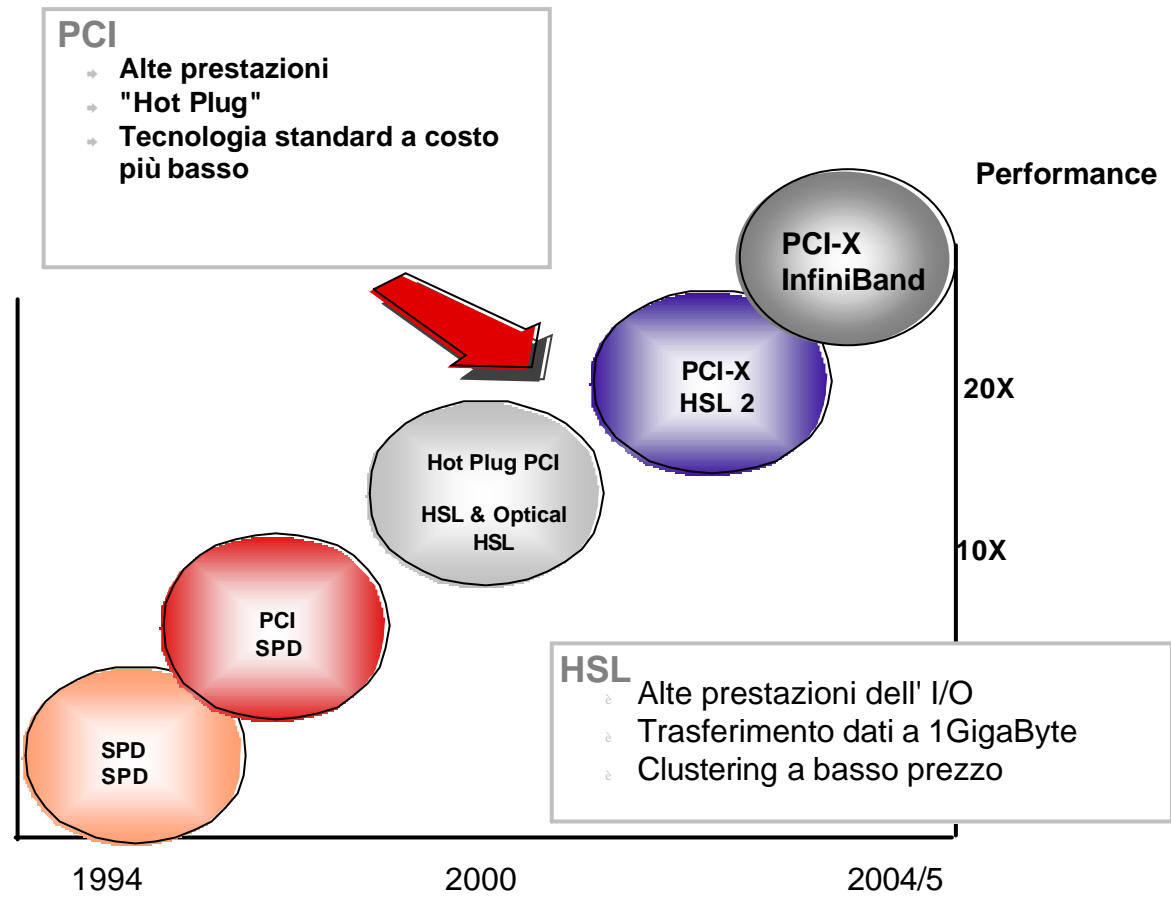
## Tecnologia innovativa - i Server del futuro



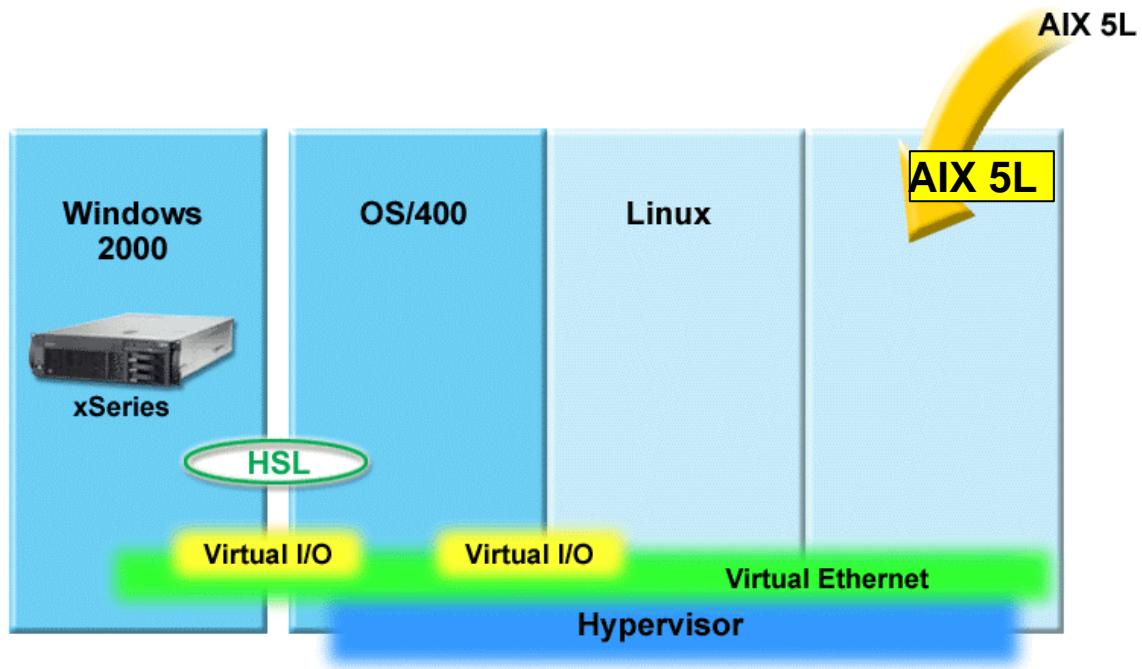
# Roadmap dei processori



# Roadmap dell'I/O



## Integrazione IBM

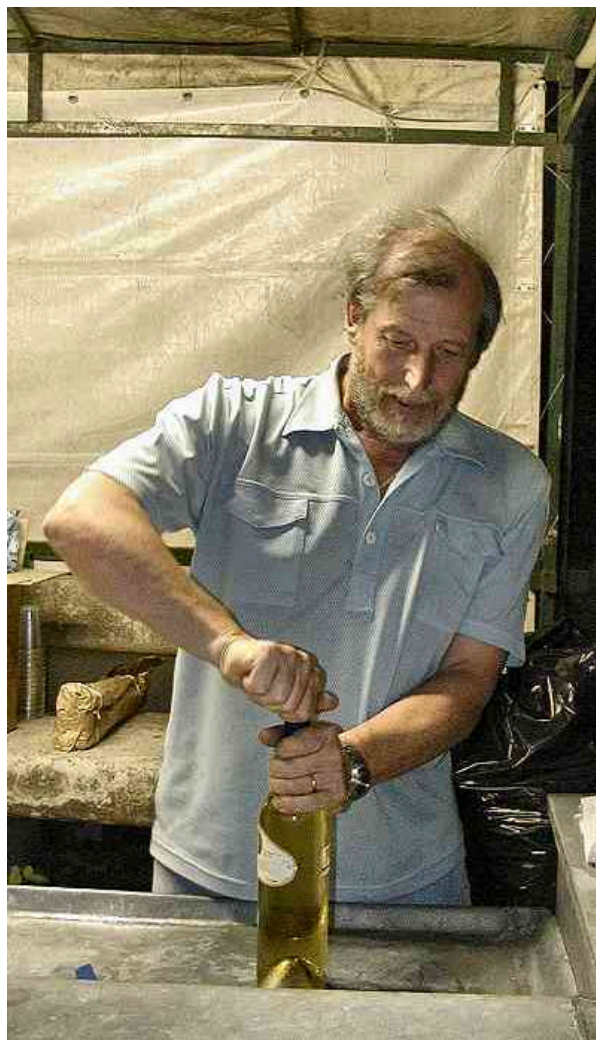


IBM ✿ iSeries

*Gr@zie !*

*Da parte di* [gianni.rossini@postel.it](mailto:gianni.rossini@postel.it)  
[rossiniemme@libero.it](mailto:rossiniemme@libero.it)

*e per il materiale fornito*  
*a Pierluigi Corno - IBM Italia*  
[pierluigi\\_corno@it.ibm.com](mailto:pierluigi_corno@it.ibm.com)



*Se il programma non gira il rosso ha comunque la soluzione giusta*