Rechnerstrukturen

Vorlesung im Sommersemester 2006

Prof. Dr. Wolfgang Karl

Universität Karlsruhe (TH)

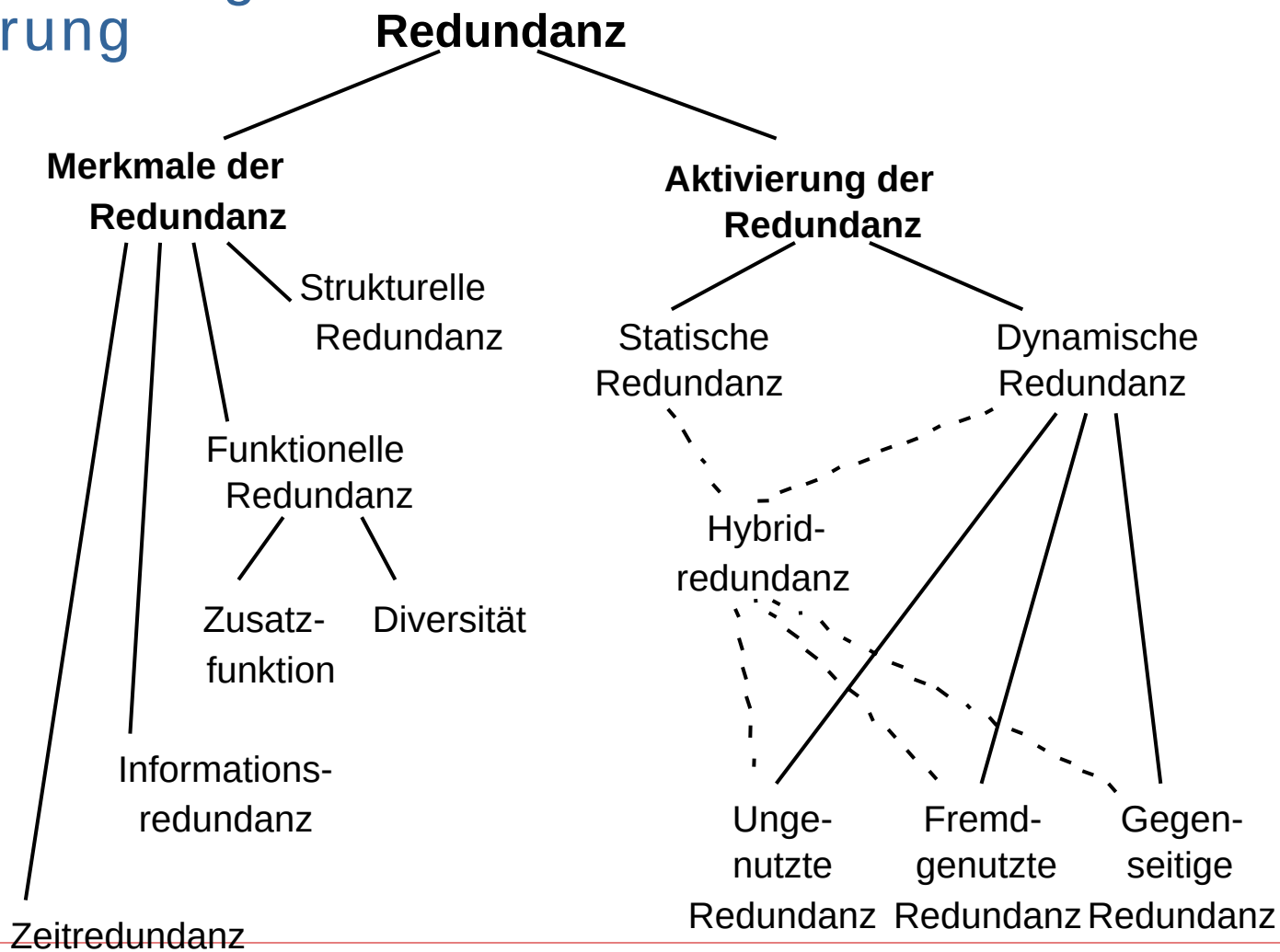
Fakultät für Informatik

Institut für Technische Informatik

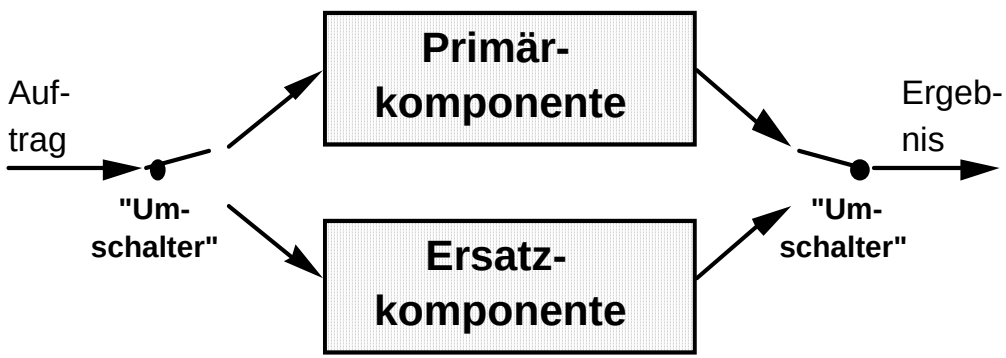
- **Kapitel 4: Fehlertoleranz, Zuverlässigkeit**

4.1: Grundlagen

- Redundanz
  - Unterscheidung nach ihren Merkmalen und ihrer Aktivierung

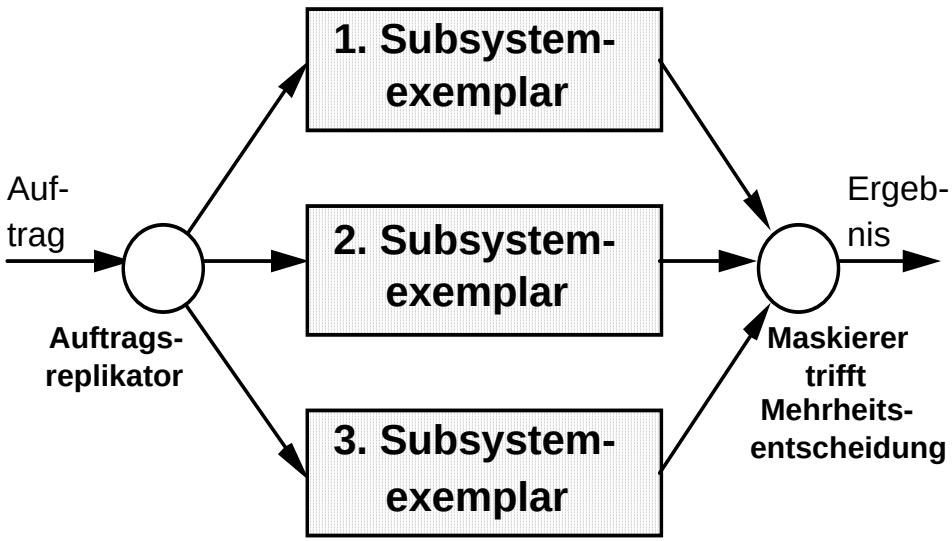


- Dynamische Redundanz (dynamic redundancy)
  - bezeichnet das Vorhandensein von redundanten Mitteln, die erst nach Auftreten eines Fehlers aktiviert werden, um eine ausgefallene Nutzfunktion zu erbringen.
  - Typisch für dynamische strukturelle Redundanz ist die Unterscheidung in Primär- und Ersatzkomponenten (bzw. Sekundär- oder Reservekomponenten).
  - Grundstruktur eines dynamisch strukturell redundanten Systems

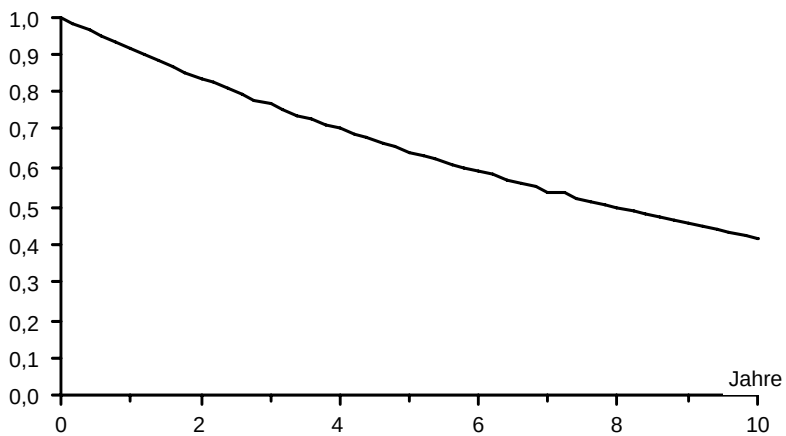


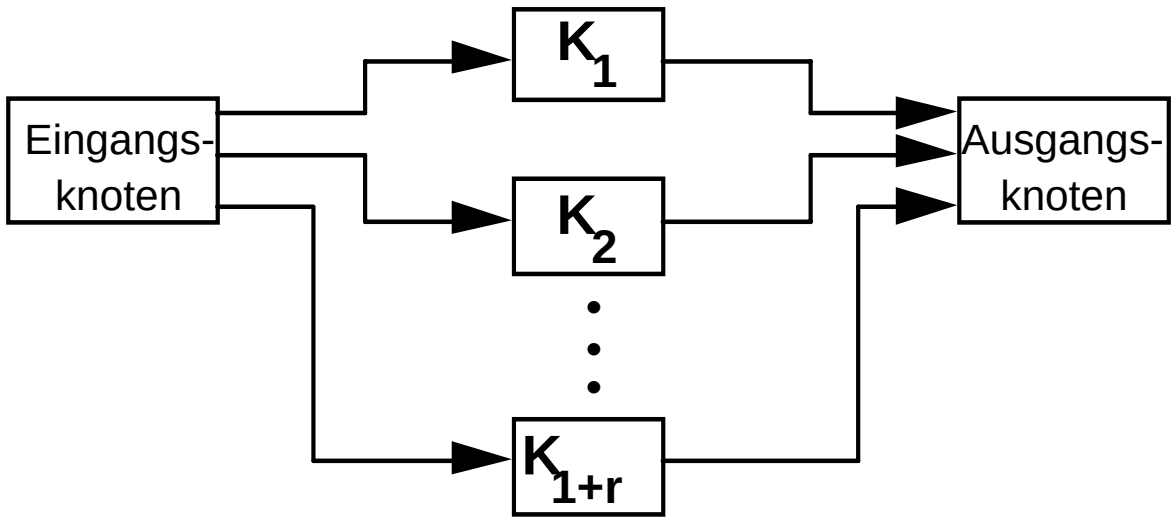
- Dynamische Redundanz (dynamic redundancy)
  - Bevor Ersatzkomponenten aktiviert werden, lassen diese sich auf eine der folgenden Arten verwenden:
  - Ungenutzte Redundanz
    - Ersatzkomponenten führen keine sonstigen Funktionen aus und bleiben bis zur fehlerbedingten Aktivierung passiv.
  - fremdgenutzte Redundanz:
    - Ersatzkomponenten erbringen nur Funktionen, die nicht zum betreffenden Subsystem gehören und im Fehlerfall bei niedrigerer Priorisierung ggf. verdrängt werden.
  - gegenseitige Redundanz:
    - Ersatzkomponenten erbringen die von einer anderen Komponente zu unterstützenden Funktionen, die Komponenten stehen sich gegenseitig als Reserve zur Verfügung.
      Dies ermöglicht einen abgestuften Leistungsabfall (graceful degradation).

- Statische Redundanz (*static redundancy*)
  - bezeichnet das Vorhandensein von redundanten Mitteln, die während des gesamten Einsatzzeitraums die gleiche Nutzfunktion erbringen.
  - Beispiel der statischen strukturellen Redundanz: *n*-von-*m*-System
  - 2-von-3-System:



- Verbesserung der Zuverlässigkeit durch Redundanz
  - Nichtredundantes Einfachsystem: S1=K1
  - Bei konstanter Ausfallrate beschreibt man die Zeitabhängigkeit der Funktionswahrscheinlichkeit φ (S_1, t) durch eine Exponentialverteilung
    - mit $z(t)=\lambda$, $\varphi(S_1, t) = e^{-\lambda \cdot t}$.
  - Beispiel:
    - Funktionswahrscheinlichkeit φ (S_1, t) mit $\lambda=10^{-5/h}$





Systemfunktion

$$S_{1+r} = K_1 \vee \ldots \vee K_{1+r}$$

Funktionswahrscheinlichkeit

$$\varphi(S_{1+r}, t) = 1 - \prod_{i=1}^{1+r} (1 - \varphi(K_i, t))$$

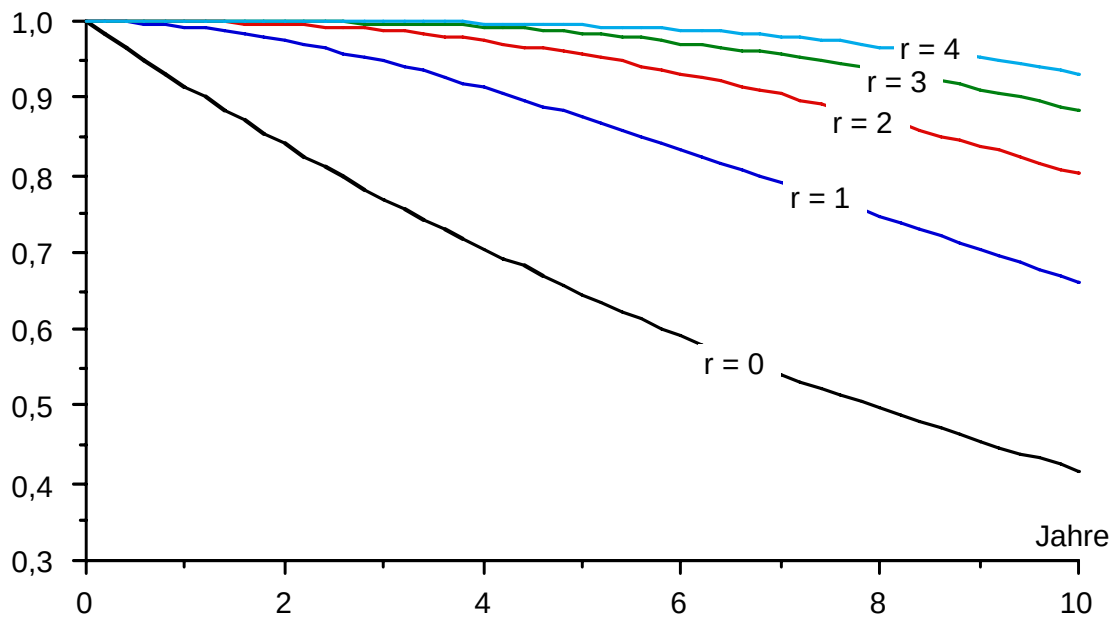
gleiche konstante Ausfallrate λ

$$\varphi(S_{1+r}, t) = 1 - (1 - e^{-\lambda \cdot t})^{1+r}$$

Zuverlässigkeitsverbesserung

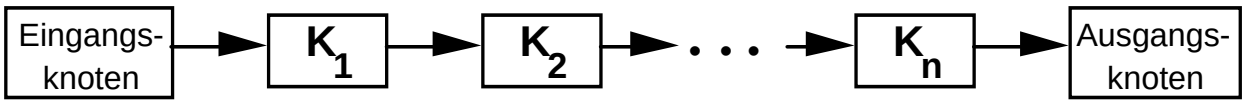
$$\Phi_{S1 \to S1+r} = (1 - e^{-\lambda \cdot t})^{-r}.$$

Funktionswahrscheinlichkeit für Parallelsystem



Annahme einer Komponentenausfallrate von $\lambda = 10^{-5}/\mathrm{h}$

Seriensystem (Nichtredundantes Mehrfachsystem)



Seriensystem

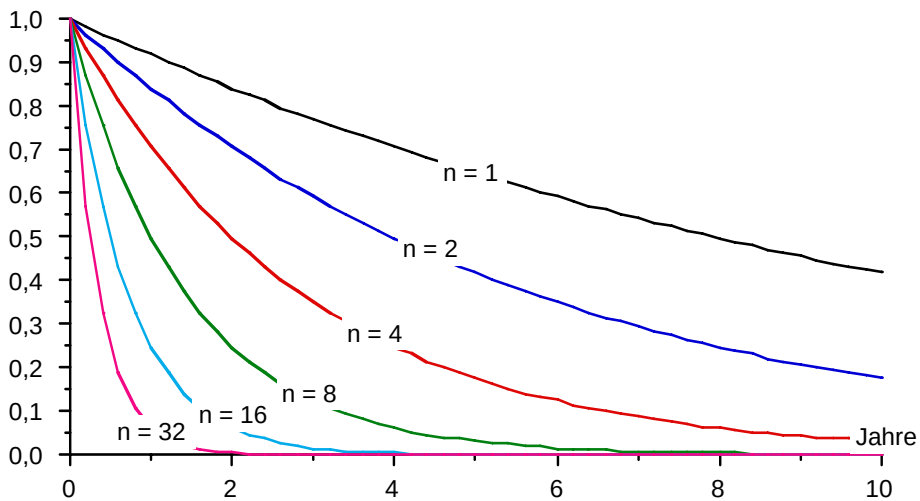
$$S_n = K_1 \wedge \ldots \wedge K_n$$

Zuverlässigkeit

$$\varphi(S_n, t) = \prod_{i=1}^{n} \varphi(K_i, t)$$

Funktionswahrschein-lichkeit $\varphi\,(S_n, t)$

für $\lambda = 10^{-5}/h$



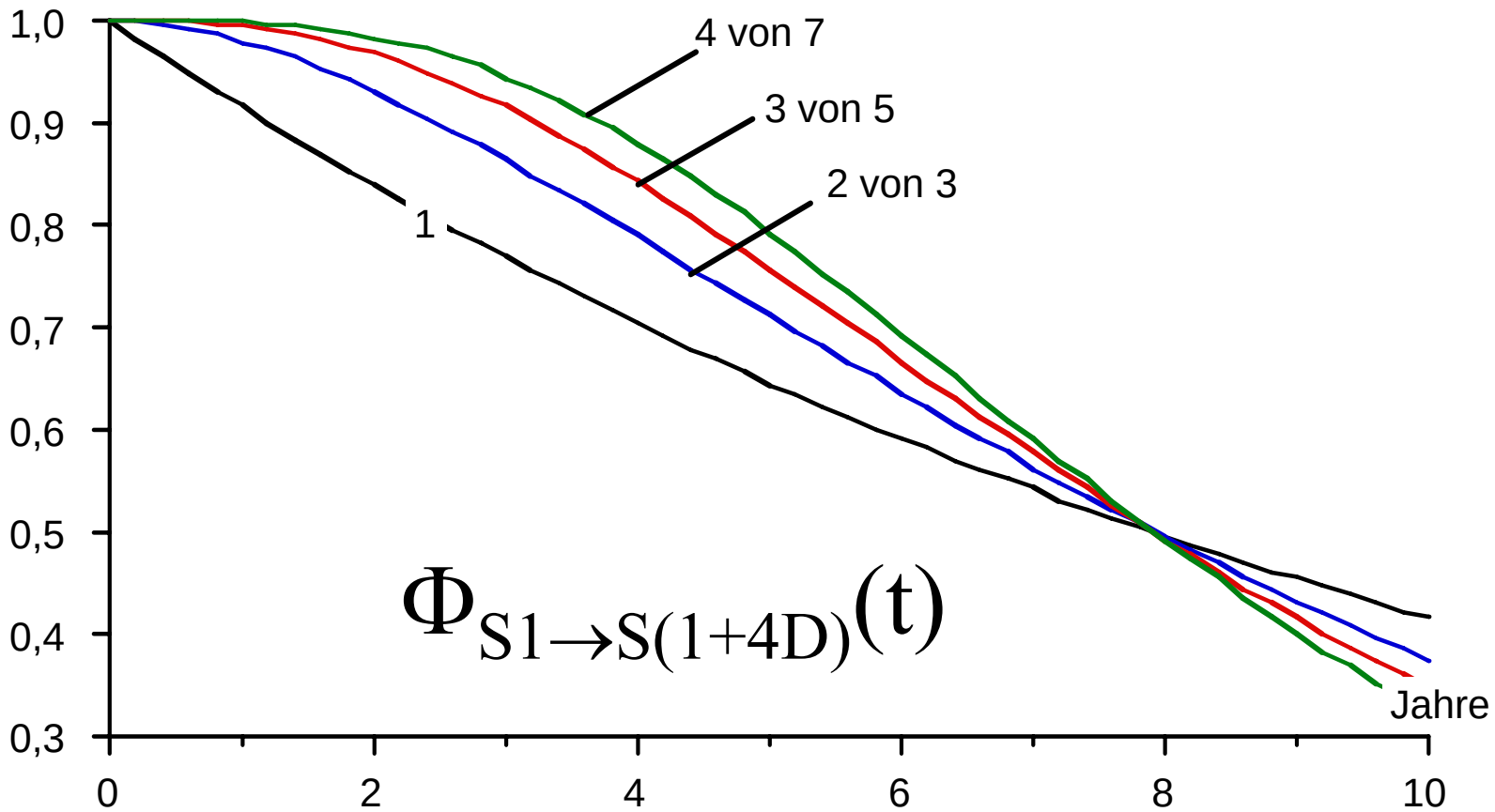
Ist die Fehlererfassung zu gering oder verbieten sich wiederholte Berechnungen wegen den geforderten maximalen Antwortzeiten, so kann statische Redundanz eingesetzt werden.

Dabei führen mehrere Komponenten die gleiche Berechnung aus, um anschließend die errechneten Ergebnisse zu vergleichen und ein mehrheitliches auszuwählen.

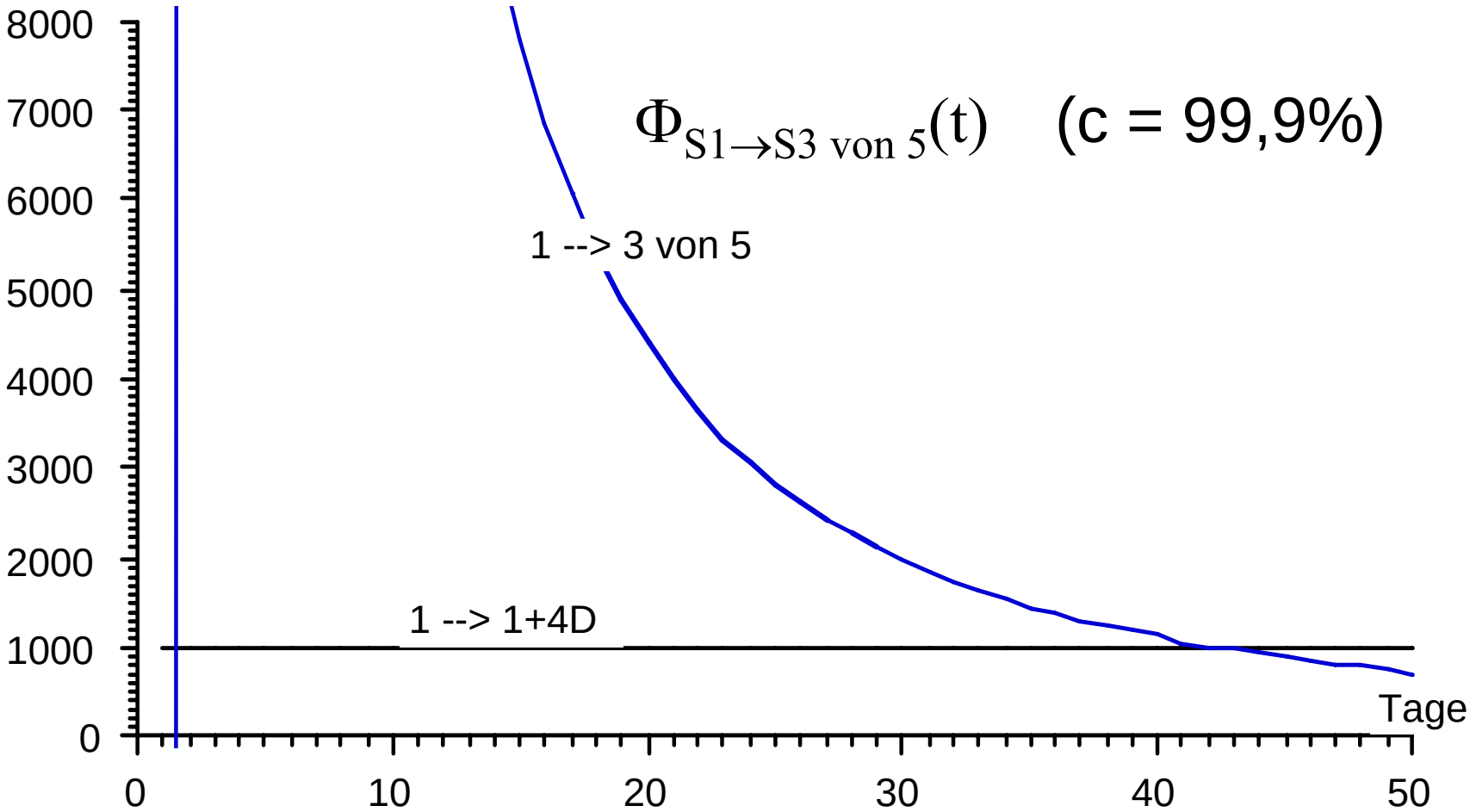
Bis zu f fehlerhafte Komponenten können überstimmt werden, wenn mindestens n=f+1 fehlerfreie, insgesamt also m=2·f+1 Komponenten vorhanden sind.

$$S_{m\ von\ m} = \bigvee_{1 \le i_1 < \ldots < i_n \le m} K_{i_1} \wedge \ldots \wedge K_{i_n}$$

Statisch redundantes System



Statisch redundantes System



Maskierer M trifft in einem statisch redundanten System die Mehrheitsentscheidung. Dies verändert Systemfunktion und Zuverlässigkeit:

$$S_{n\,von\,mM} = M \wedge \left(\bigvee_{1 \le i_1 < \ldots < i_n \le n+r} K_{i_1} \wedge \ldots \wedge K_{i_n} \right)$$

Funktionswahrscheinlichkeit ist durch die des "Zuverlässigkeitsengpasses" M beschränkt:

$$\varphi\,(S_{n\ von\ mM}) \le \varphi\,(M).$$

- **1. Alternative: Verbesserung der Komponenten**
  - **+einfacher Ansatz zur Verbesserung**
  - **+bis zu einer gegebenen Grenze kostengünstig**
  - **–ab dieser Grenze steigen die Kosten überproportional**
  - **–Lösung oft nicht leistungsfähig und zuverlässig zugleich**

- **2. Alternative: Zusätzliche Komponenten**
  - **–bei Verdoppelung oder Vervierfachung hoher Aufwand**
  - **+Prüfzeichen sind ein effizientes Mittel gegen spezielle Fehler**
  - **– Ansatz ist unflexibel, da meist voller Zusatzaufwand notwendig**

- **3. Alternative: Zusätzliche Subsysteme (zusätzliche Rechner)**
  - **+alle Rechner gleich, keine Spezialrechner**
  - **+flexible Lastverteilung (und Umverteilung bei Fehler) möglich**
  - **–überproportionale Kosten**
  - **–Aufwand zur Herstellung der aktuellen Zustandsinformation**
  - **–Aufwand zur Vermeidung von Inkonsistenzen**

- **Kapitel 3: Multiprozessoren – Parallelismus auf Prozess/Thread-Ebene**

3.5: Multiprozessoren mit verteiltem Speicher

- IBM Blue Gene/L Überblick
  - Massively Parallel Supercomputer
    - Ziel:
      - günstiges Cost/Performance-Verhaltnis für ein breites Spektrum von Anwendungen
      - Günstiges Performance/Power-Verhältnis
    - Grundlegender Ansatz:
      - System-on-Chip Design für den Prozessor
        - » Hohe Integrationsdichte
        - » Low Power
        - » Low Design Cost
      - Hohe Skalierbarkeit der Anwendungen
        - » Hohe Anforderung an die Skalierbarkeit des VErbindungsnetzwerkes

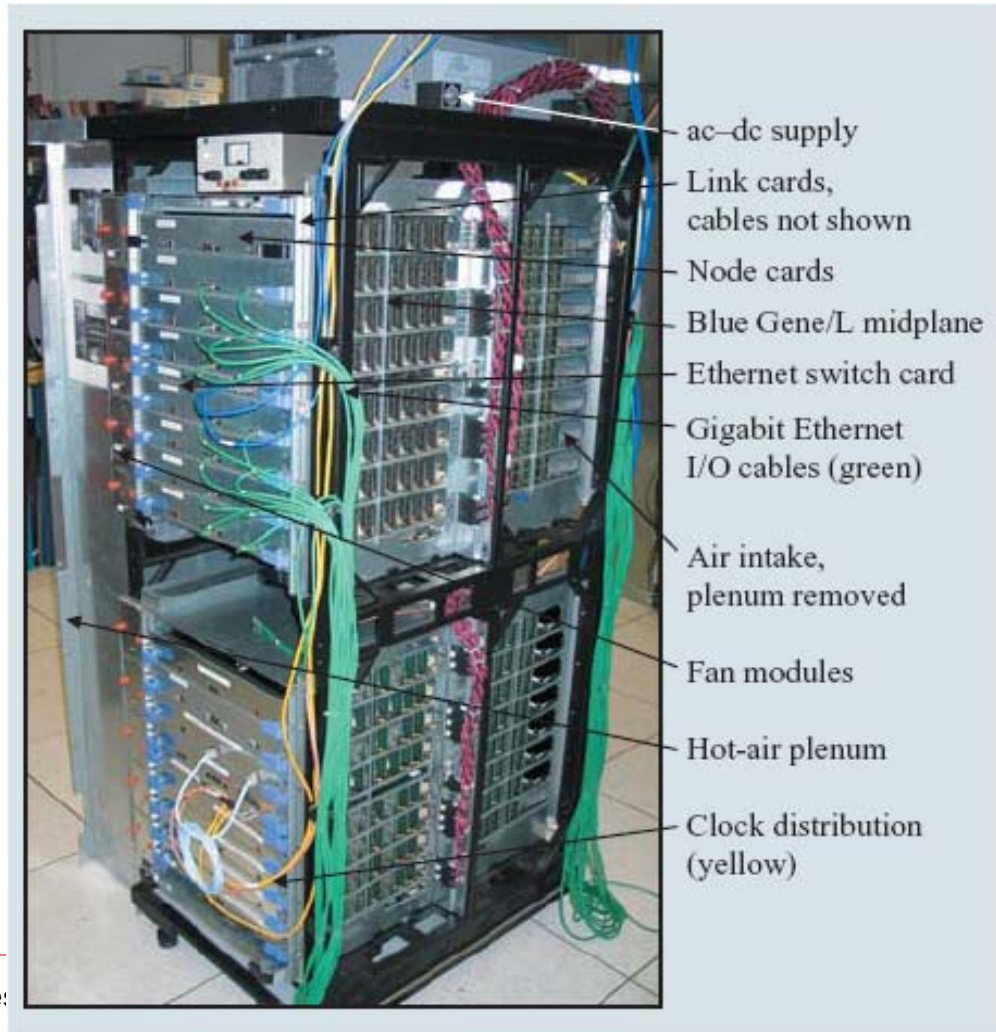
- IBM Blue Gene/L Überblick
  - Massively Parallel Supercomputer
    - Bedeutung Low-Power
      - Für einen Rechner mit einer Leistung im Bereich 380 TFlops mit konventionellen Hochleistungsprozessoren würde der Leistungsverbrauch bei etwa 10 MW – 20 MW liegen, was den Energieverbrauch einer 11000 Einwohner Stadt entspricht.
      - Ein Rack mit 1024 Dual-Prozessor Knoten
        - » Ausmaße: 0.9m x 0.9m x 1,9m
        - » Energieverbrauch: 27,5 kW
    - Bedeutung: Zuverlässigkeit, Verfügbarkeit, Sicherheit (Reliability, Availability, Security, RAS)
    - Bedeutung: Programmierunterstützung
      - Nachrichten-orientiertes Programmiermodell MPI,

- IBM Blue Gene/L Überblick
  - Massively Parallel Supercomputer
    - Anwendungsszenarios:
      - Simulation physikalischer Phänomene
      - Echtzeit-Datenverarbeitung
      - Off-line Datenanalyse
      - Anwendungen in den großen amerikanischen Forschungslabors

- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - 65536 Knoten
      - ASIC: Dual-Processor Chip
      - 18 SDRAM chips
    - Knoten über 5 Netzwerke verbunden
      - Wichtigstes Netzwerk mit höchster Bandbreite
        - » 64 x 32 x 32 3-D Torus

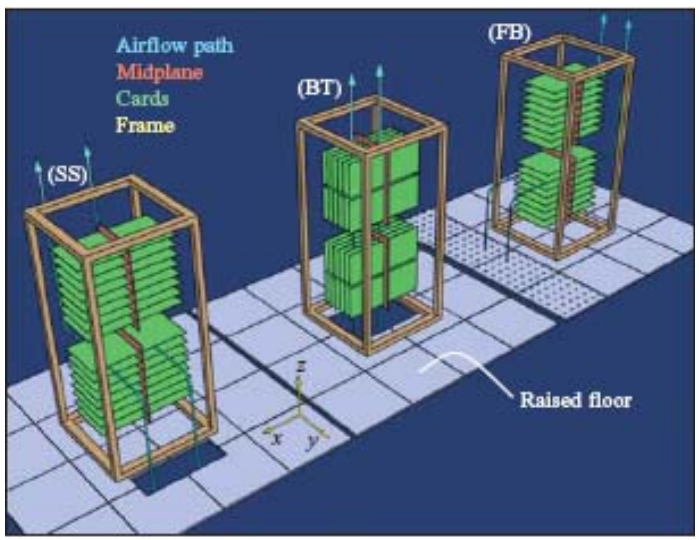
- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - 65536 Knoten bis zu 64 Racks, die auch so organisiert werden können, als wären es verschiedene Systeme, wobei auf jedem ein eigenes Single Software Image läuft
    - Knoten
      - 2 BG/L Compute ASIC (BLC)
        - » Dual Processor SoC ASIC
      - 9 Double data rate synchronous dynamic random access memory chips (DDR SDRAM chips) pro ASIC
    - Knoten über 5 Netzwerke verbunden
      - Wichtigstes Netzwerk mit höchster Bandbreite
        - » 64 x 32 x 32 3-D Torus
        - » Global Collective Network
        - » Global Barrier and Interrupt Network
        - » I/O Network (Gigabit Ethernet)
        - » Service Network

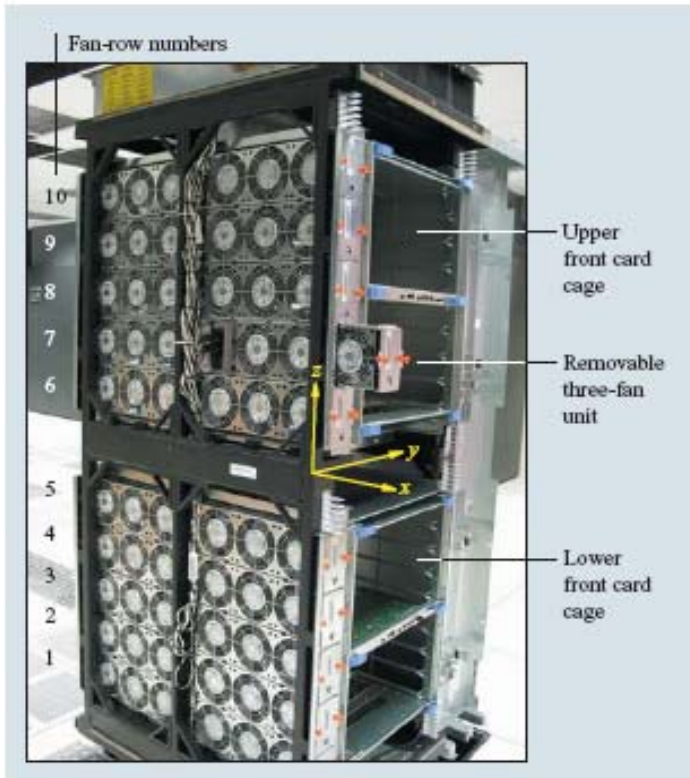
- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - System Rack



Quelle: P. Coteus, et.al.: Packaging the Blue Gene/L supercomputer. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

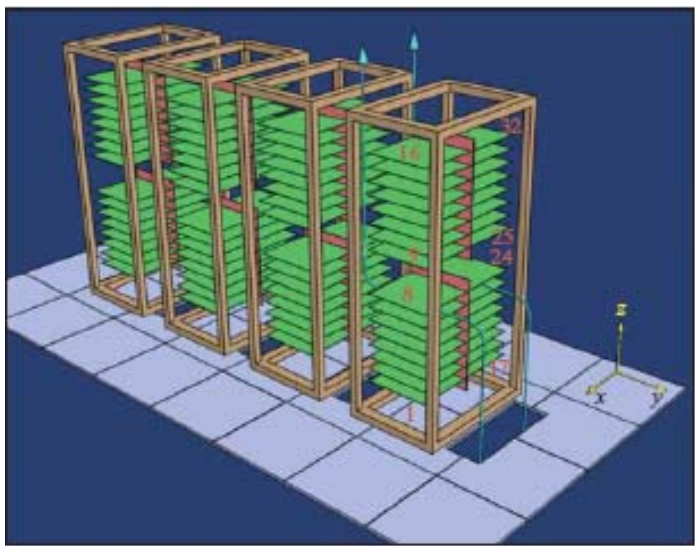
- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - Kühlung

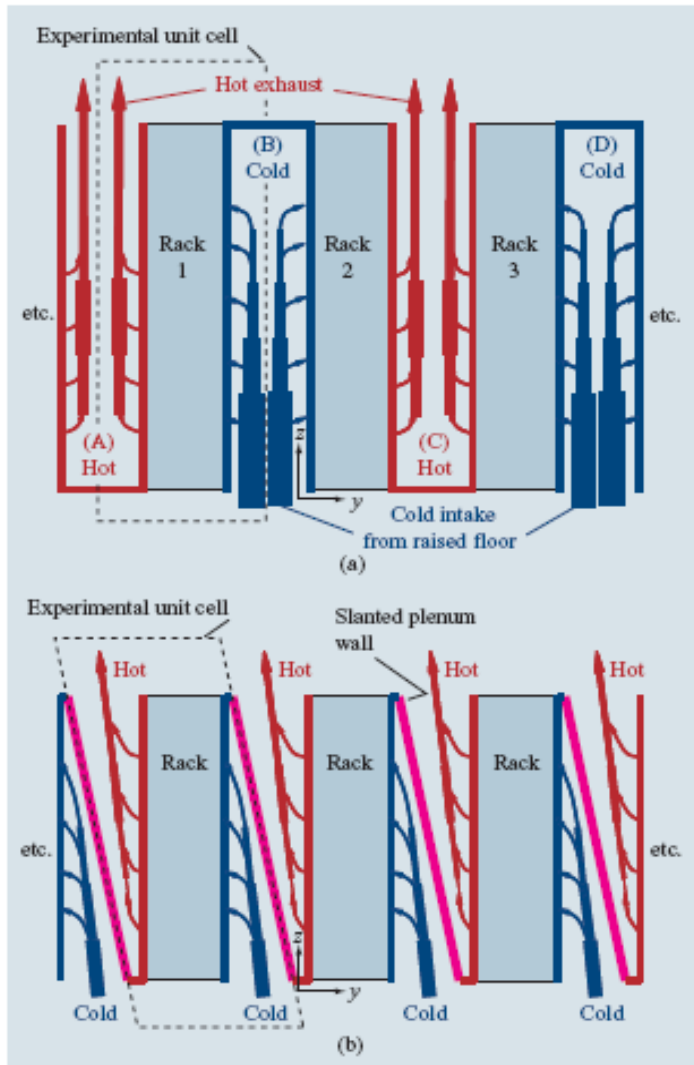




Quelle: P. Coteus, et.al.: Packaging the Blue Gene/L supercomputer. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

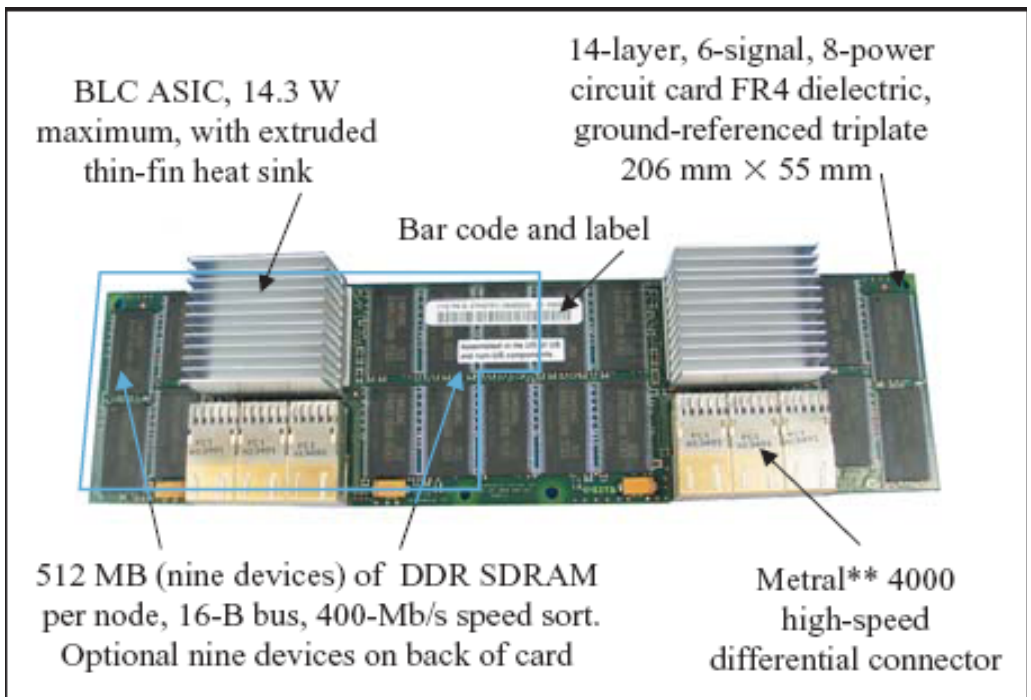
- IBM Blue Gene/L Überblick
  - Systemaufbau





Quelle: P. Coteus, et.al.: Packaging the Blue Gene/L supercomputer. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - BG/L Compute card

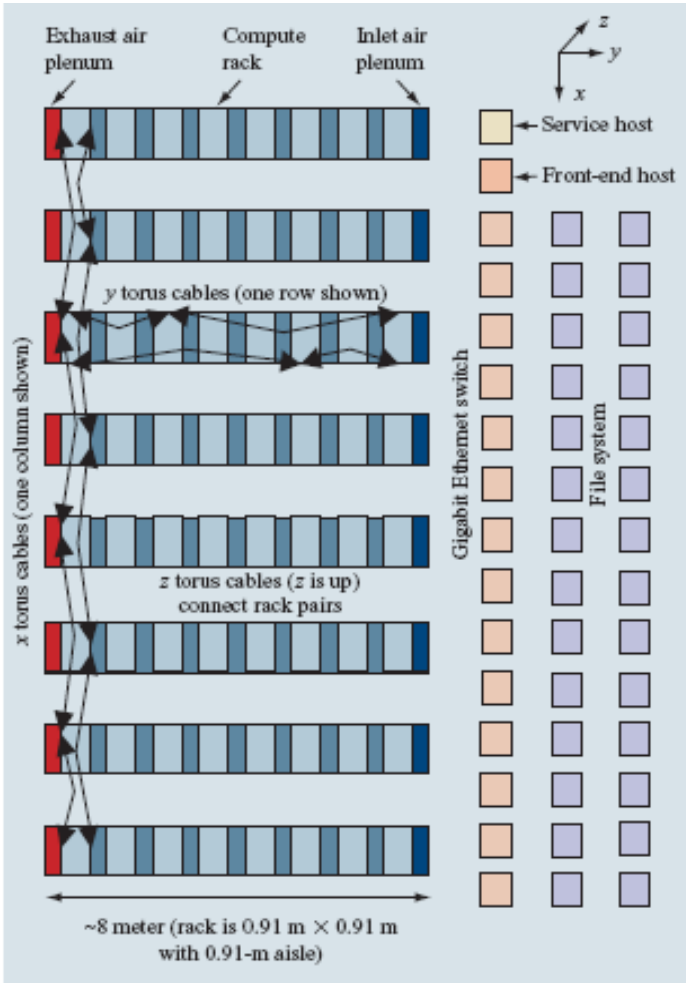


Quelle: P. Coteus, et.al.: Packaging the Blue Gene/L supercomputer. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Systemkomponenten
    - BG/L Node Card

Quelle: P. Coteus, et.al.: Packaging the Blue Gene/L supercomputer. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

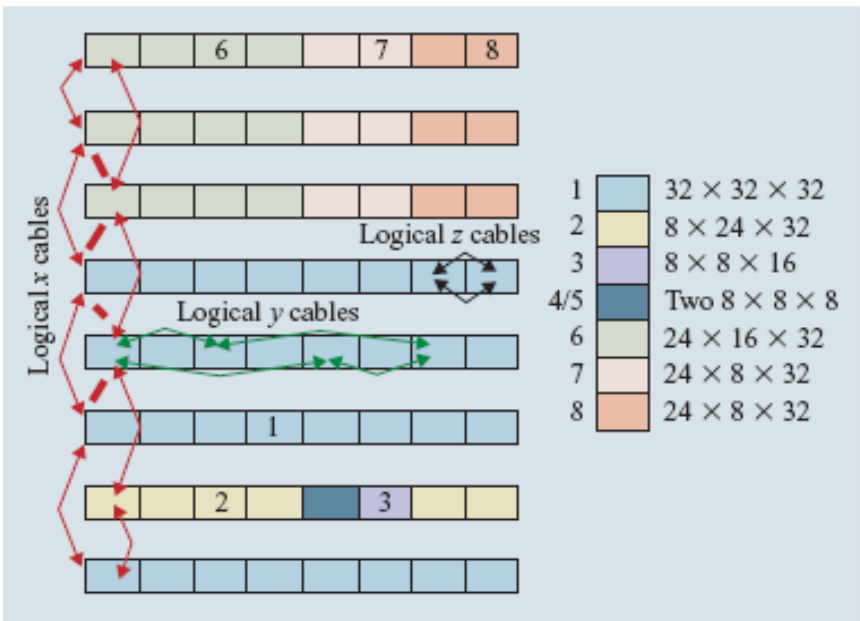
- IBM Blue Gene/L Überblick
  - Ein BG/L System kann für eine Anwendung konfiguriert werden



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

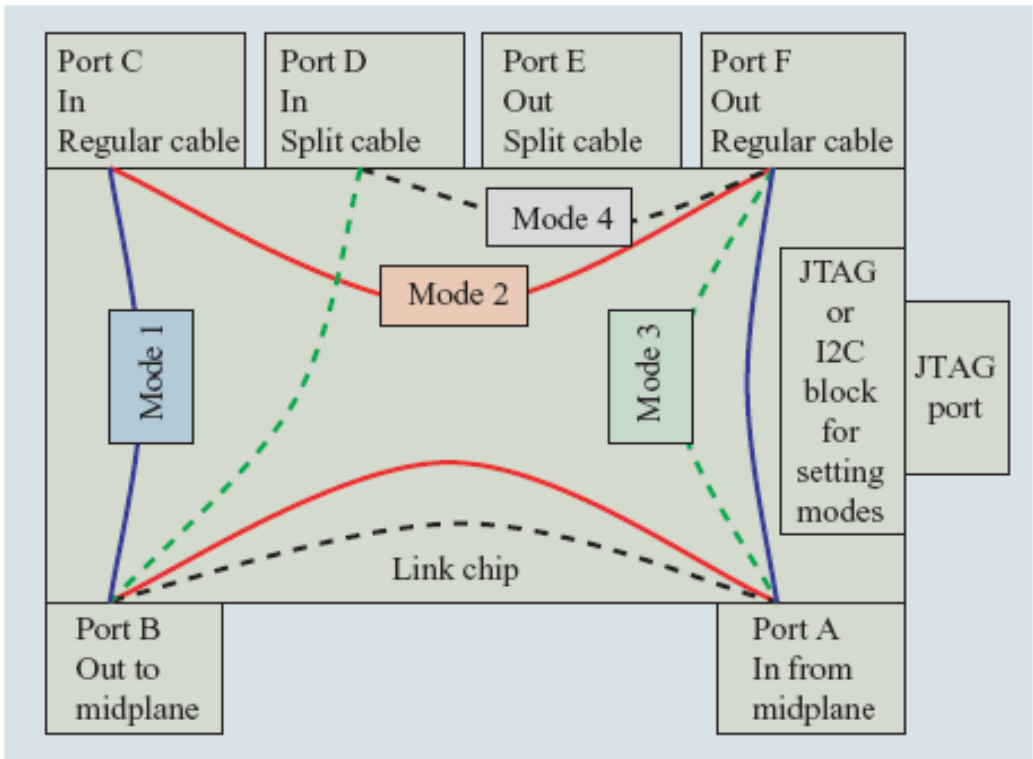
- IBM Blue Gene/L Überblick
  - Systempartitionierung
    - Partitionierung in kleinere Systeme
      - Beispiel System mit 20K Knoten (20 Rack-System)
        - » 4 Reihen mit 4 Compute Racks (16 K Knoten)
        - » Mit Stand-by Menge von 4 Racks für Fail-over
    - 2 Host-Rechner
      - Verwaltung des Rechners
      - Vorbereiten der Jobs
    - I/O Racks mit RAIDs
    - Switch Racks
      - Mit Gigabit Ethernet für die Verbindung der Compute Nodes, I/O Nodes, und Host-Rechner

- IBM Blue Gene/L Überblick
  - Systempartitionierung
    - Partitionierung für 8 Benutzer



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

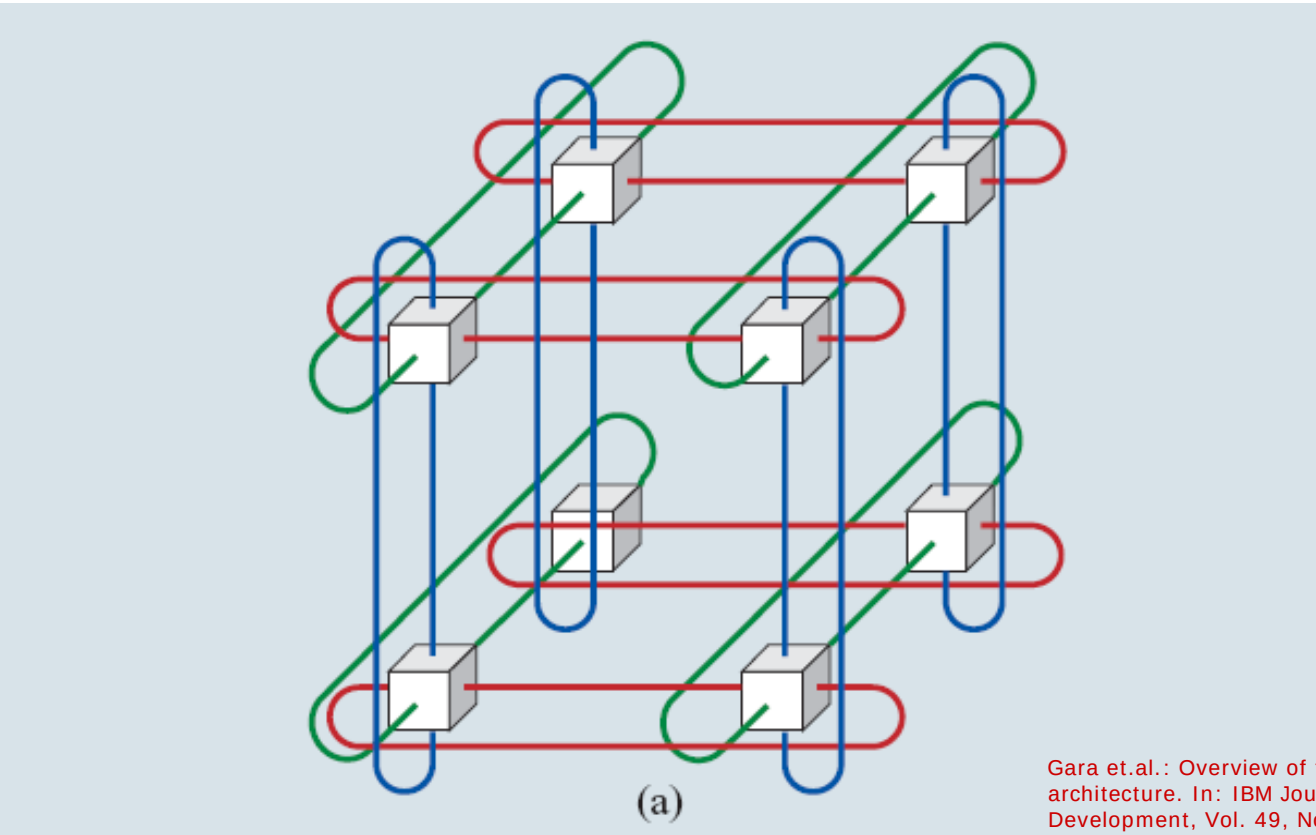
- IBM Blue Gene/L Überblick
  - BG/L Link chip



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - BG/L Link chip
    - Ports A und B direkt mit „midplane“ verbunden
    - Ports C,D,E und F sind mit Kabeln verbunden
    - Statisches Routing, das vom Host bei der Partitionierung festgelegt wird
      - Bleibt bis zu einer Neukonfigurierung bei einer neuen Partitionierung fest
    - Jeder Link chip Port bedient 16 unidirektionale Torus Links
    - Weitere Signale für Collective und Barrier Network
    - Jede Midplane enthält 24 Link Chips
    - Jede Midplane bildet ein 8x8x8 Gitter

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - 3-D Torus (Beispiel 2 x 2 x 2 Torus)

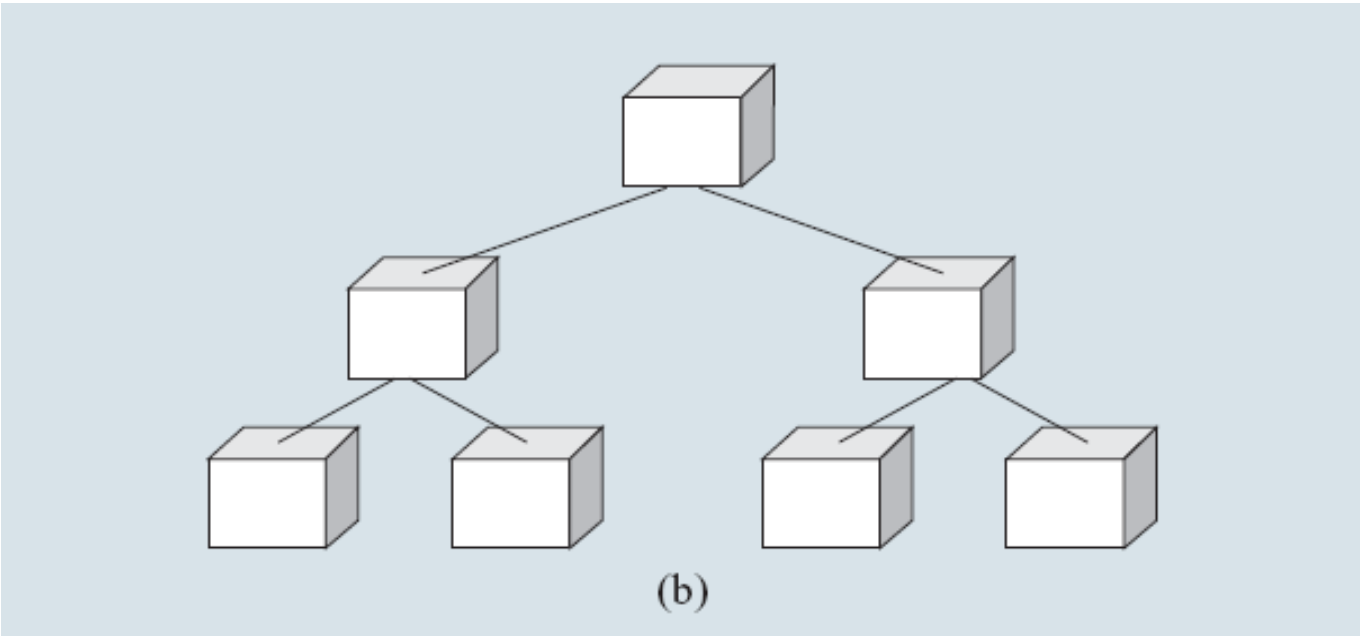


Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - 3-D Torus
      - Jeder Knoten kann mit jedem Knoten kommunizieren
      - Jeder Knoten teilt seine Kommunikationsbandbreite mit Cut-through-Verkehr von anderen Knoten
      - Kommunikationsabhängige effektive Bandbreite
      - Algorithmenentwurf
        - » Möglichst lokale Kommunikation
      - Cut-Through Routing
      - Adaptive Routing
        - » Erlaubt jeden minimalen Pfad zu wählen
        - » Möglichst blockierungsfrei
        - » Dynamische Wahl der Route für die Pakete
    - Multicast-Unterstützung in jede Richtung
    - Kommunikationslatenz für für die am weitesten entfernten Knoten: 6,4µs (64Hops)

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Collective Network
      - Erstreckt sich über gesamte Maschine
      - Daten können von jedem Knoten zu allen anderen verschickt werden (broadcast)
        - » 5µs Latenz
      - Zusätzliche Arithmetik-Reduktionsoperationen
        - » Min, max, sum, OR, AND, XOR Operationen
        - » Z.B. für globale Summation
      - Statisches Routing

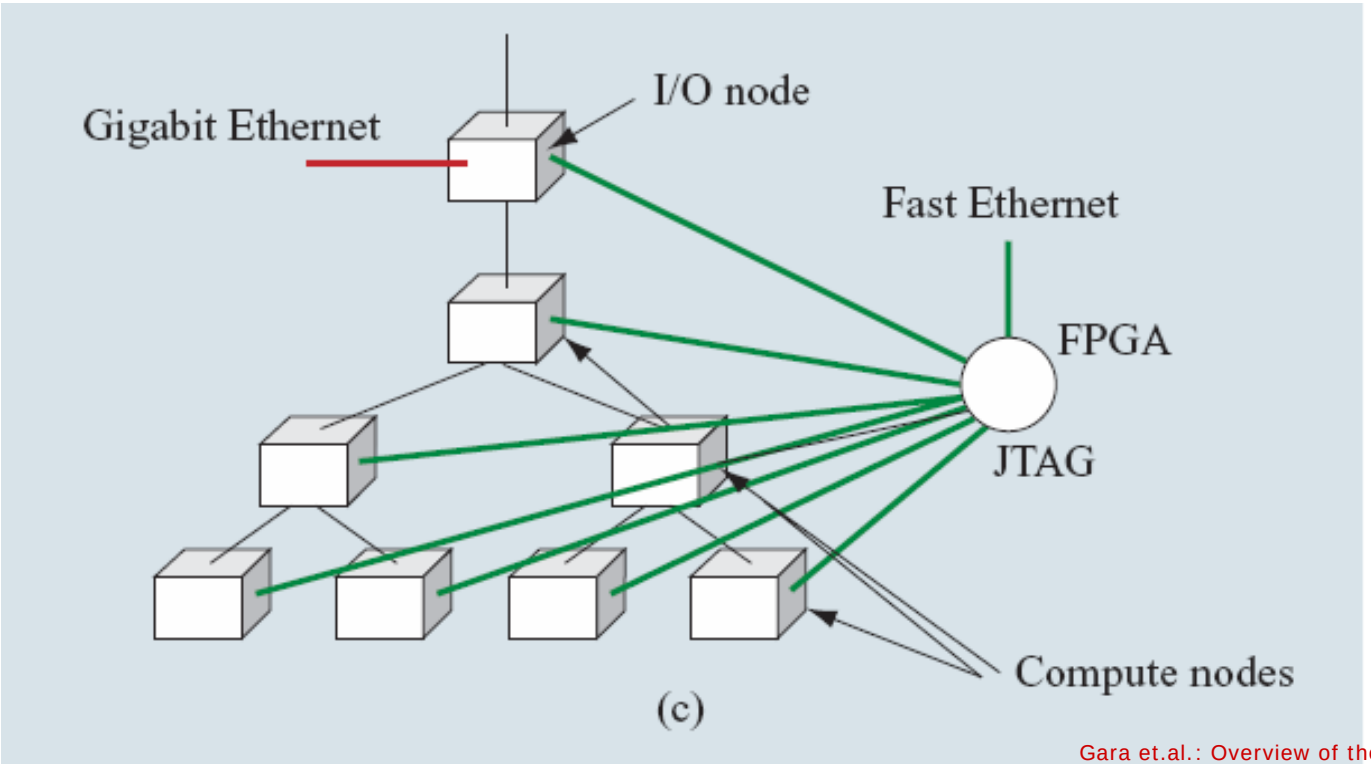
- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Collective Network



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Barrier Network
      - Verbesserung der Latenz für globale Operationen
      - 4 unabhängige Kanäle
        - » Globales OR über alle Knoten: globaler Interrupt, wenn die Maschine oder eine Partition angehalten werden muss, z.B. für Diagnose-Zwecke
        - » Individuelle Signale werden in Hardware verknüpft und an die physikalische Wurzel eines Baums weitergeleitet
        - » Das Ergebnis-Signal wird an alle Knoten im Baum verteilt (broadcast)
        - » Globale AND-Operation mit Hilfe Inverter-Logik: globaler Barrier
        - » Round-Trip-Latenz: 1,5µs bei 64K Knoten

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Control system network



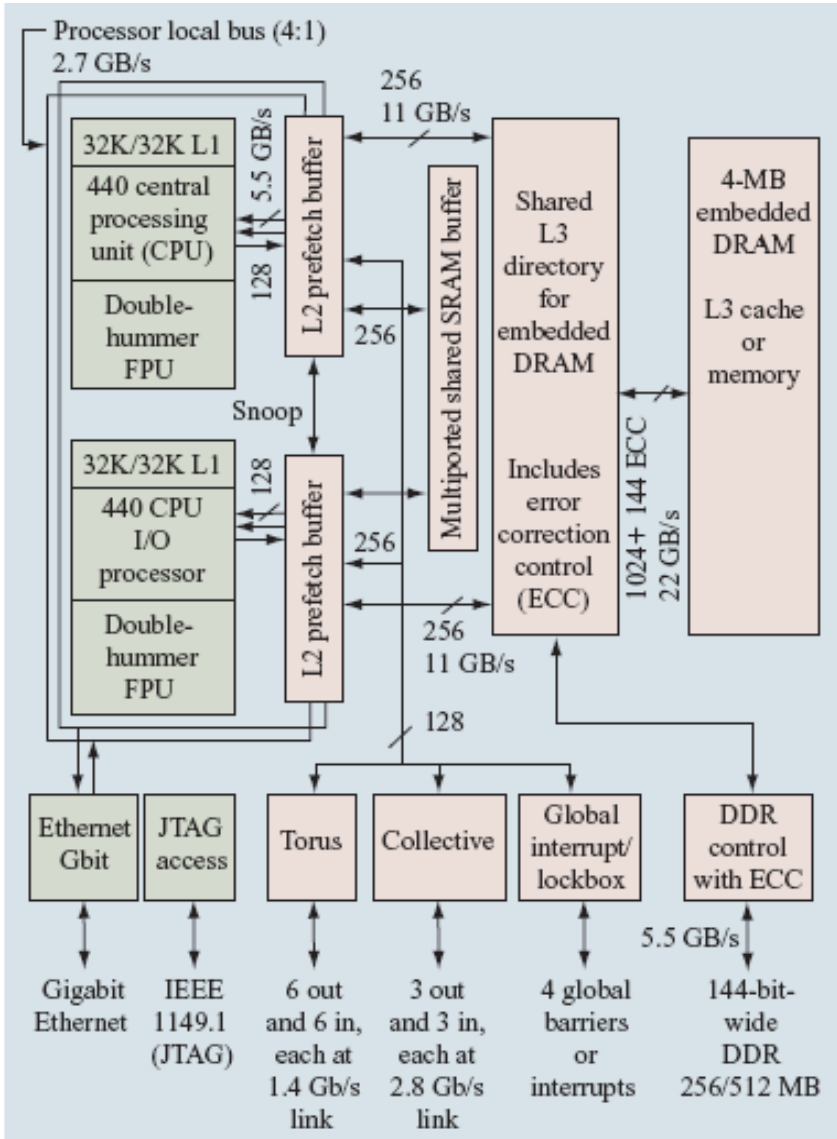
Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Control system network
      - Eine BG/L Maschine enthält eine Menge von 250000 Endpunkten in Form von ASICs, Temperatursensoren, Spannungsversorgung, Taktversorgung, Kühler, Status-Leuchtdioden, etc., die alle initialisiert, gesteuert und beobachtet werden müssen
      - Diese Aktionen werden von Service Node durchgeführt
      - Zugriff auf Endknoten über ein Intranet auf Ethernet-Basis
      - Control-FPGA übernimmt Protokoll-Umsetzung in verschiedene Netzwerkprotokolle

- IBM Blue Gene/L Überblick
  - Verbindungsnetzwerke
    - Gigabit-Ethernet
      - I/O-Knoten haben Gigabit-Ethernet-Schnittstelle für den Zugriff auf externe Ethernet-Switches
      - Verbindung zwischen I/O-Knoten und dem externen parallelen File-System sowie zum externen Host
      - Anzahl I/O-Knoten ist konfigurierbar
        - » Maximales I/O zu Compute-Node-Verhältnis ist 1:8

- IBM Blue Gene/L Überblick
  - Blue Gene/L Node
    - BLC ASIC
      - SoC, integriert die wesentlichen Funktionen eines Rechners auf einem Chip
        - » 2 PowerPC 440
        - » FP-Core für jeden Prozessor
        - » Embedded DRAM
        - » DDR Memory Controller für externen Speicheranschluss
        - » Gigabit Ethernet-Adapter
        - » Alle Puffer für die Torus-Netzwerk-Schnittstelle

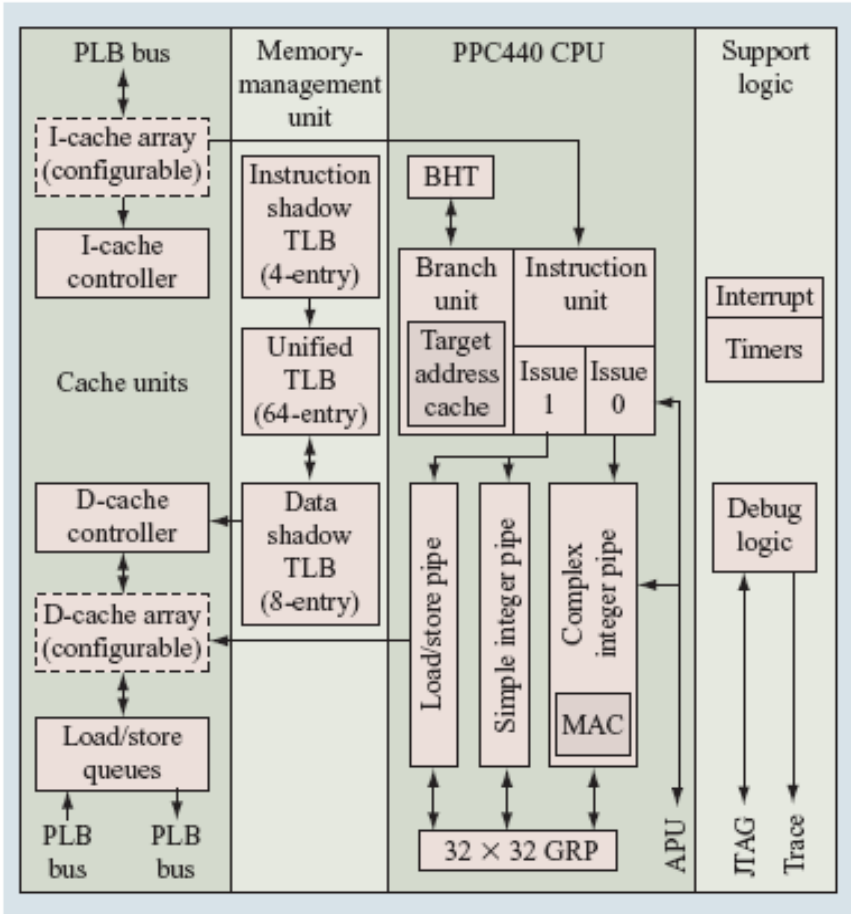
- IBM Blue Gene/L Überblick
  - Blue Gene/L Node
    - BLC ASIC



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Blue Gene/L Node
    - PowerPC 440
      - Taktfrequenz: 700 MHz
      - Superskalartechnik
      - 32-Bit Book-E Enhanced PowerPC Befehlssatz-Architektur
      - 7-stufige Pipeline

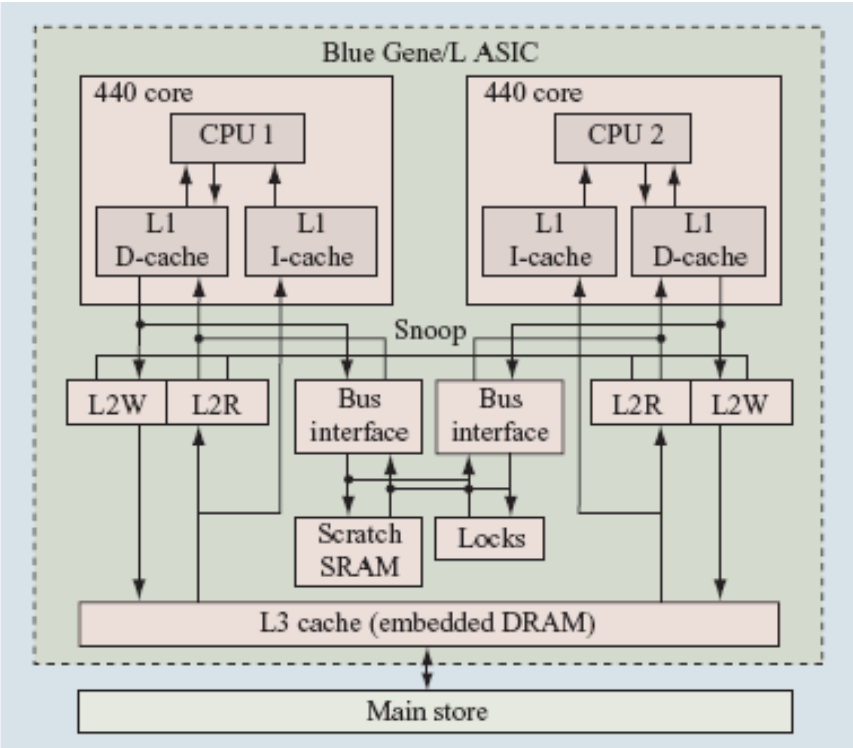
- IBM Blue Gene/L Überblick
  - Blue Gene/L Node
    - PowerPC 440



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Blue Gene/L Distributed Memory Architektur
    - Hierarchie:
      - On-chip Cache-Hierarchie
      - Off-Chip Hauptspeicher
      - On-Chip-Logik für Synchronisation und Kommunikation der beiden Prozessoren auf dem Chip
    - Verteilte Speiche-Architektur
      - Jeder Knoten hat 512 MB physikalischen Speicher
        - » Gemeinsamer Speicher für die beiden Prozessoren auf dem Chip
      - Insgesamt: 32 TBytes Speicher

- IBM Blue Gene/L Überblick
  - Blue Gene/L Distributed Memory Architektur



Gara et.al.: Overview of the Blue Gene/L system architecture. In: IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, pp.195-212

- IBM Blue Gene/L Überblick
  - Blue Gene/L Distributed Memory Architektur
    - Kohärenz
      - PPC440 Core keine Kohärenz-Unterstützung
        - » SW unterstützt Kohärenz auf L1-Ebene
      - L2 und L3 sind sequentiell konsistent mit Hardware-Unterstützung
      - Kein Inklusions-Eigenschaft für L1 und L2 sowie L1 und L3

- IBM Blue Gene/L Überblick
  - Blue Gene/L Distributed Memory Architektur
    - Kohärenz
      - PPC440 Core keine Kohärenz-Unterstützung
        - » SW unterstützt Kohärenz auf L1-Ebene
      - L2 und L3 sind sequentiell konsistent mit Hardware-Unterstützung
      - Kein Inklusions-Eigenschaft für L1 und L2 sowie L1 und L3
      - Communiaction coprocessor mode
        - » Ein Prozessor übernimmt Kommunikationsaufgaben
        - » Der andere Übernimmt die Berechnungen
        - » L1 Kohärenz wird auf System-Ebene mit Hilfe von Bibliotheken erreicht
      - Virtual Node Mode
        - » Knoten wird logisch in zwei Knoten mit jeweils einen Prozessor und dem halben physikalischen Speicher aufgeteilt
        - » Jeder Prozessor kann auf seinen eigenen Speicherbereich lesend und schreibend zugreifen und auf den anderen lesend
        - » Vermeidet Duplizieren von Anwendungsdaten
        - » Auf Knoten laufen zwei Anwendungsprozesse

- Literatur:
  - IBM Journal of Research and Development, Vol. 49, No. 2/3, 2005, Special Issue