



BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR
MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK

Parciális rekonfiguráció

Heterogén számítási

rendszerek

VIMIMA15

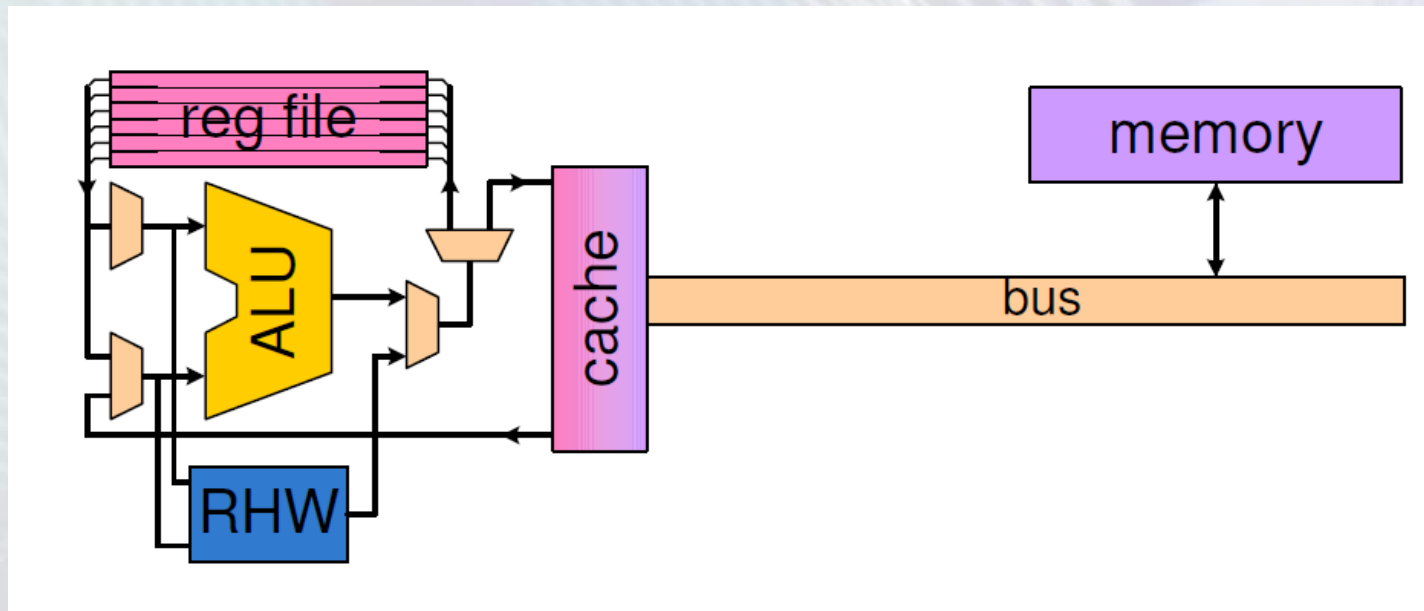
Fehér Béla
BME MIT

Konfigurálható CPU-k

- **FPGA használata jellemzően HW orientált feladatokra, közvetlen leképezéssel**
- **Speciális, egyedi funkcionális egységek, adatfolyam számítási modell**
- **Általános feladatok gyakran igényelnek komplex vezérlést → CPU (Mikrorendszerek tervezése...)**
- **Megjelentek a soft / lágy processzorok, szokásos felépítés, de néhány egyedi tulajdonság**
- **Pl. Utasításkészlet felhasználói kiegészítése**
- **USER1, USER2, ... utasítások**

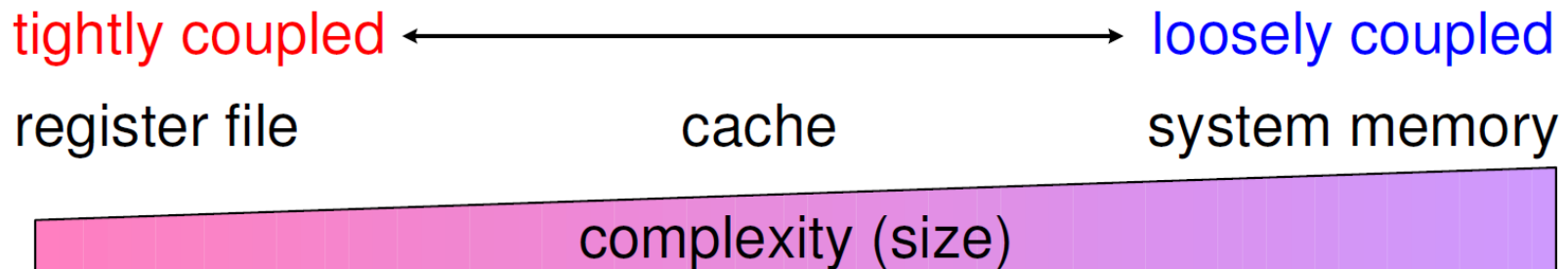
Konfigurálható CPU-k

- A hagyományos RISC típusú CPU mag kis HW-es kiegészítése
- Alkalmazás függő feladatokra egyedi, nem túl bonyolult HW végrehajtás (pl. POP_COUNT)



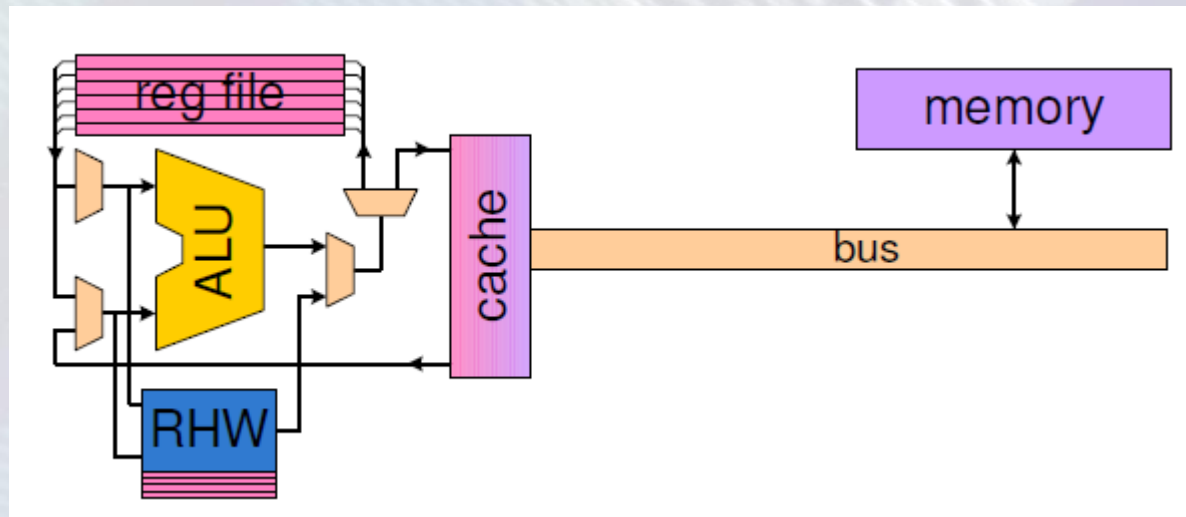
Konfigurálható CPU-k

- A szükséges HW komplexitás kicsi, előnye jelentős lehet, a szoftver rendszer (fordító) fel van készítve
- A végrehajtás szorosan kapcsolódik a CPU utasításvégrehajtásához, nem független attól
- Nincs saját regiszterkészlet, független adattárolás
- 1-2 funkcióra 10x – 20x gyorsítás, de nem párhuzamosítás, a CPU dolgozik, csak hatékonyabban (mint pl. a DSP MAC funkció)



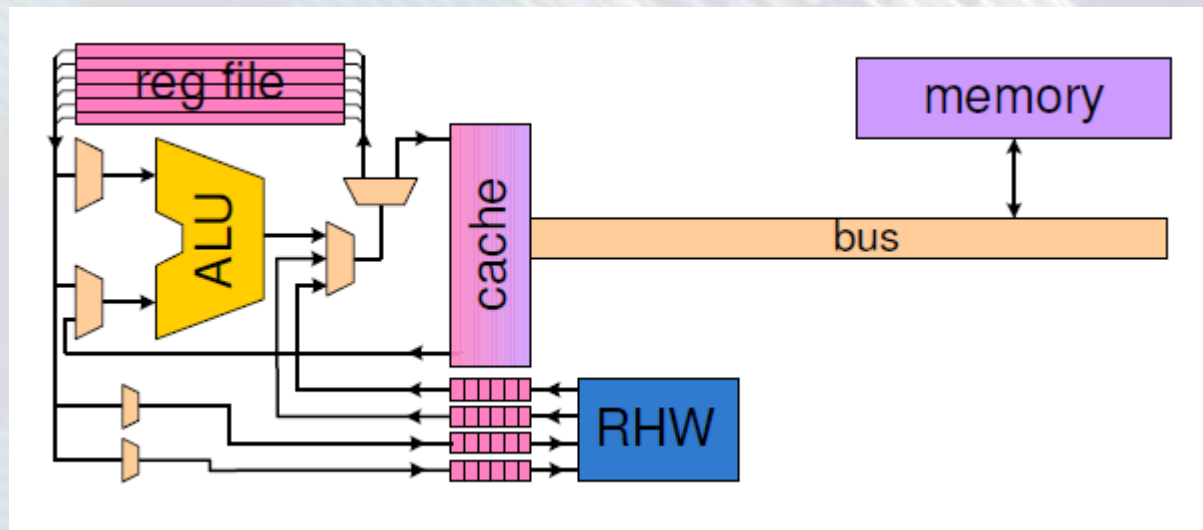
Konfigurálható CPU-k

- A kiegészítő HW tartalmazhat saját regisztereket, átmeneti adatok tárolására (pl. komplex szorzás $Im \cdot Re$ értékek)
- Adatmozgatás csak IN-OUT értékekre, hatékonyabb
- Még itt sincs párhuzamosítás, de jobb a végrehajtás



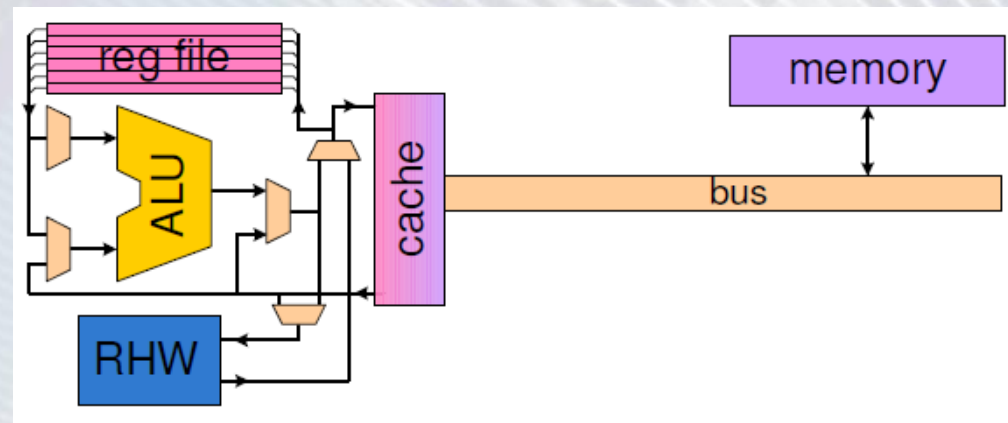
Konfigurálható CPU-k

- A kiegészítő HW egy FIFO IN-OUT kommunikációs csatornával kapcsolódik a CPU maghoz
- Ez még mindig szorosan csatolt, CPU utasítások illesztik (pl. MicroBlaze FSL) de már párhuzamosan működhet az ALU/REG FILE tömb mellett
- Összetettebb feladatok, részfunkciók megvalósítása



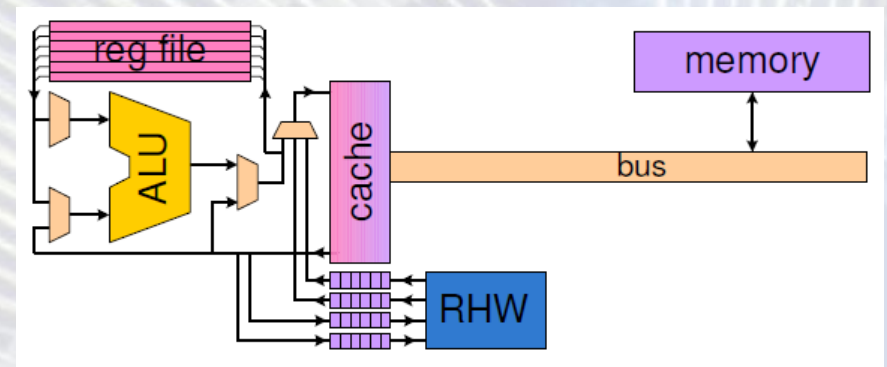
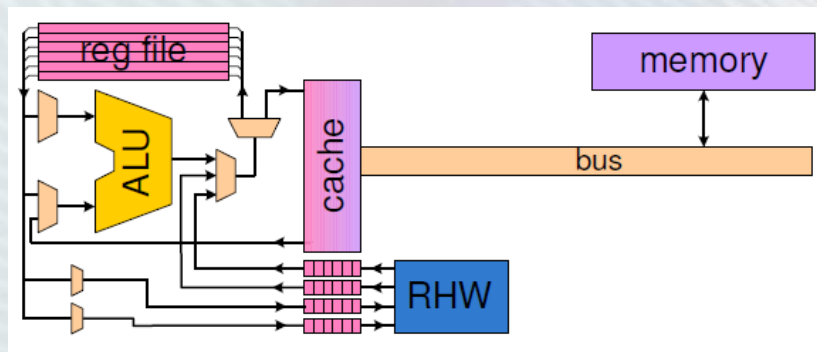
Konfigurálható CPU-k

- A kiegészítő HW már egy önálló társprocesszor jellegű funkció, a CPU kiegészítője
- Már valódi egyedi feladatokra, természetesen átkonfigurálható módon, akár több verzióban
- Saját memória hozzáférés (esetleg csak a TCM vagy az D\$ szinten), de a CPU-val konkurens módon
- Nincs közvetlen kapcsolat a CPU belső magjával



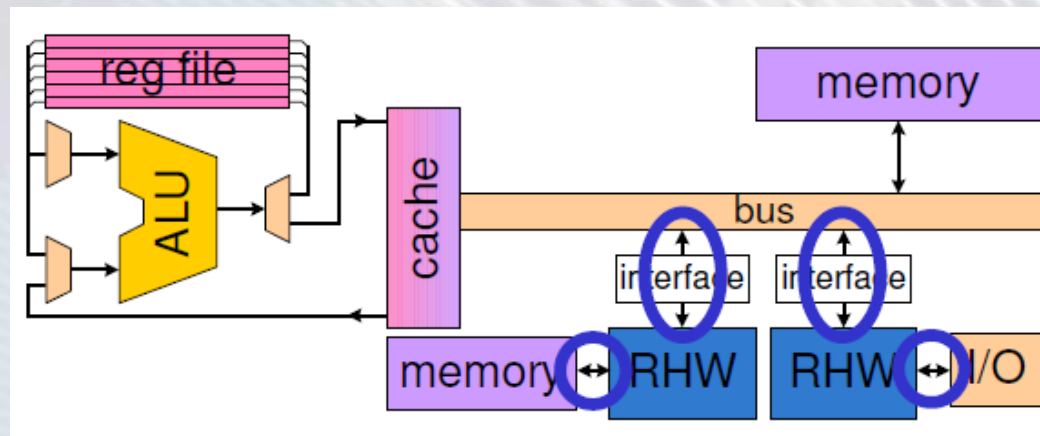
Konfigurálható CPU-k

- A korábbi Xilinx FPGA-kban a Hard-core PowerPC CPU-k ilyen módon voltak kiegészíthetők, FSL-FIFO, ill. TCM-FIFO interfészekkel
- Kedvező elrendezés:
 - 400MHz hard-core PPC CPU (1, 2, 4 db)
 - 200MHz dedikált HW feldolgozó egység(-ek)
 - Teljesen párhuzamos működés



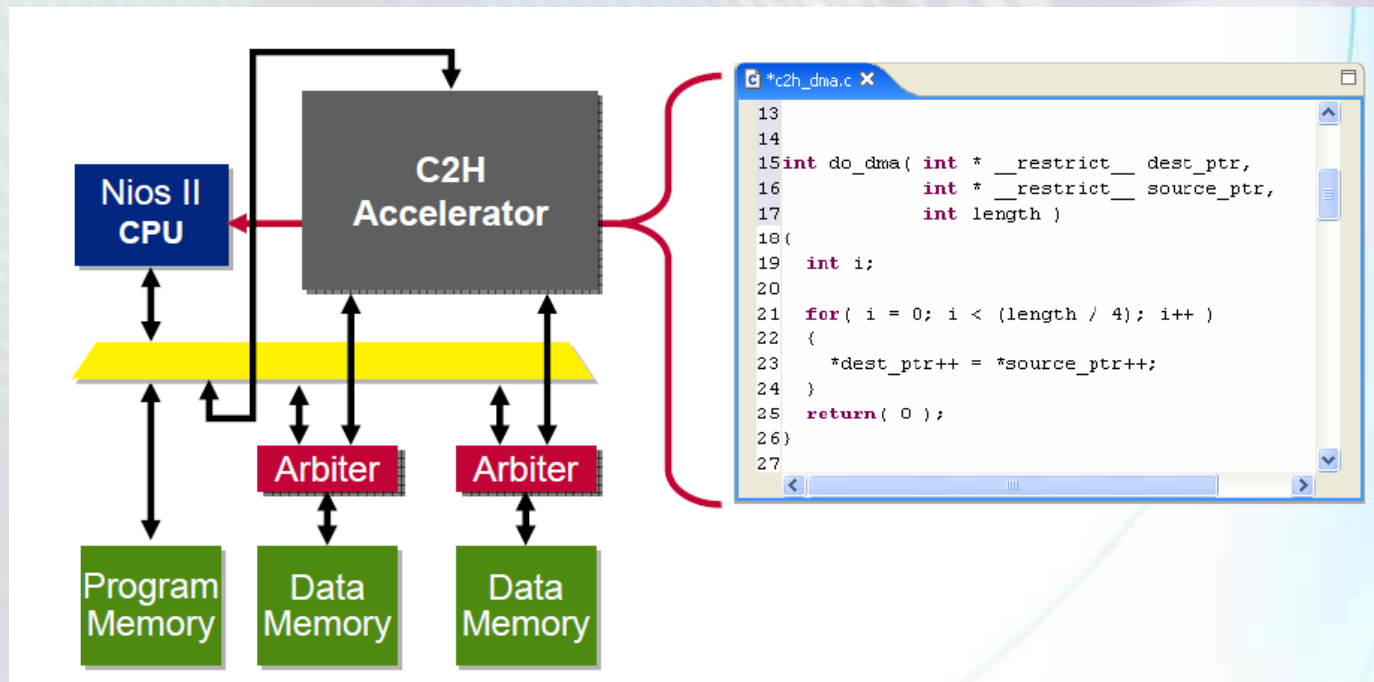
Konfigurálható CPU-k

- A következő lépés a konfigurálható HW távolítása a CPU-tól, attól nagyrészt független felépítéssel
- Itt a modul már önálló rendszerelem
- Szükséges a beépíthetőséghez egy szabványos busz interfész használata
- Tipikus alkalmazások: Nagysebességű adat/video interfészek, hálózati elemek busz master funkcióval



Konfigurálható CPU-k

- Egy szép példa: Altera C-to-HW technológia
- Az időigényes SW funkció azonosítható, HW-ben realizálható
- Persze kell hozzá egy ~HLS típusú fordító,

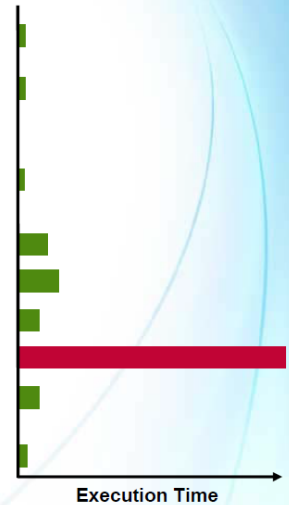


Konfigurálható CPU-k

- Időigényes SW funkció azonosítása
- A feladat konvertálása HW műveleti egységekre
- Egyedi adatfeldolgozó hálózat felépítése (pipeline, ütemezés, műveleti pontosság)
- Egyedi vezérlő és busz interfész kialakítása
- CPU SW adaptálása

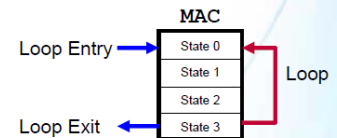
```
main ()
{ ...variable declarations...
  init();

  while (!error && got_data())
  {
    do_user_interface();
    gather_statistics();
    if (got_new_data())
      d_transform(in_buf, out_buf);
    check_for_errors();
  }
  cleanup();
}
```



```
long long MAC
(int *a, int *b, int len)
{
  long long result = 0;
  while (len > 0) {
    result += *a++ * *b++;
    len--;
  }
  return result;
}
```

- 1 - 32x32 multiplier
- 3 - 32-bit incrementers
- 1 - 64-bit adder
- 1 - 32-bit comparator
- 2 - read-masters
- Nominal control logic



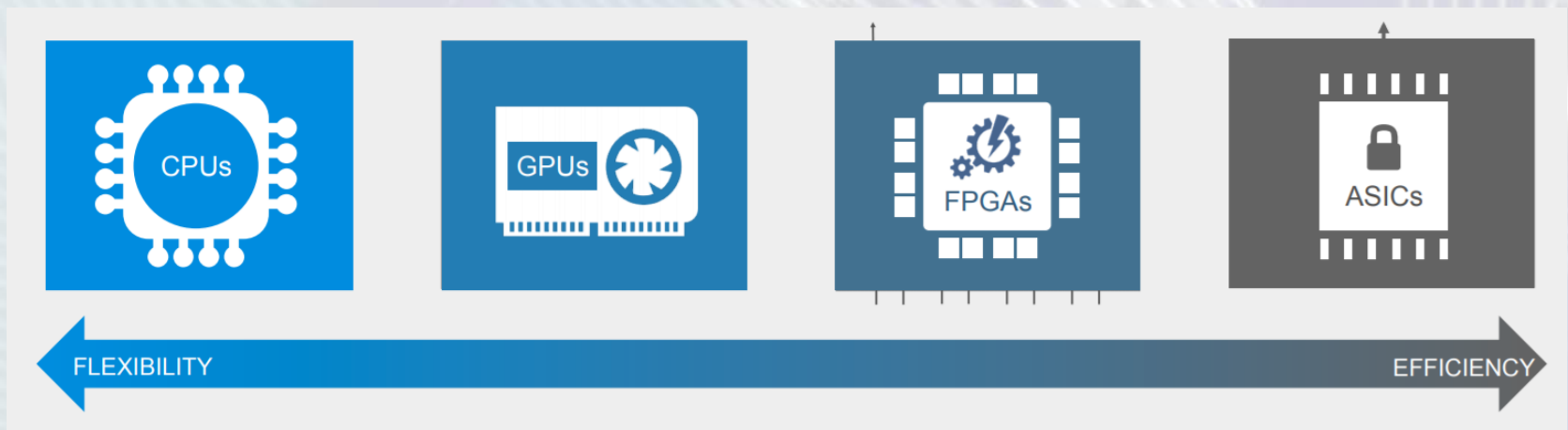
Every converted software function has its own dedicated hardware state machine

Konfigurálható HW gyorsítók

- **A HW műveletvégzés előnyei sok esetben nyilvánvalóak**
- **Adott rendszerbe FPGA alapú gyorsító kártyák helyezhetők el**
- **Manapság a legelterjedtebb a PCIe interfész használata, de létezhet ennél közvetlenebb kapcsolat**
- **Az egyedi rendszerek gyorsítása mellett megjelentek a felhő alapú szolgáltatások is, ahol FPGA alapú gyorsító eszközök találhatóak**
- **Ezek használata a dinamikus részleges vagy teljes átkonfigurálási technológián alapul**

Felhőszolgáltatások igényei

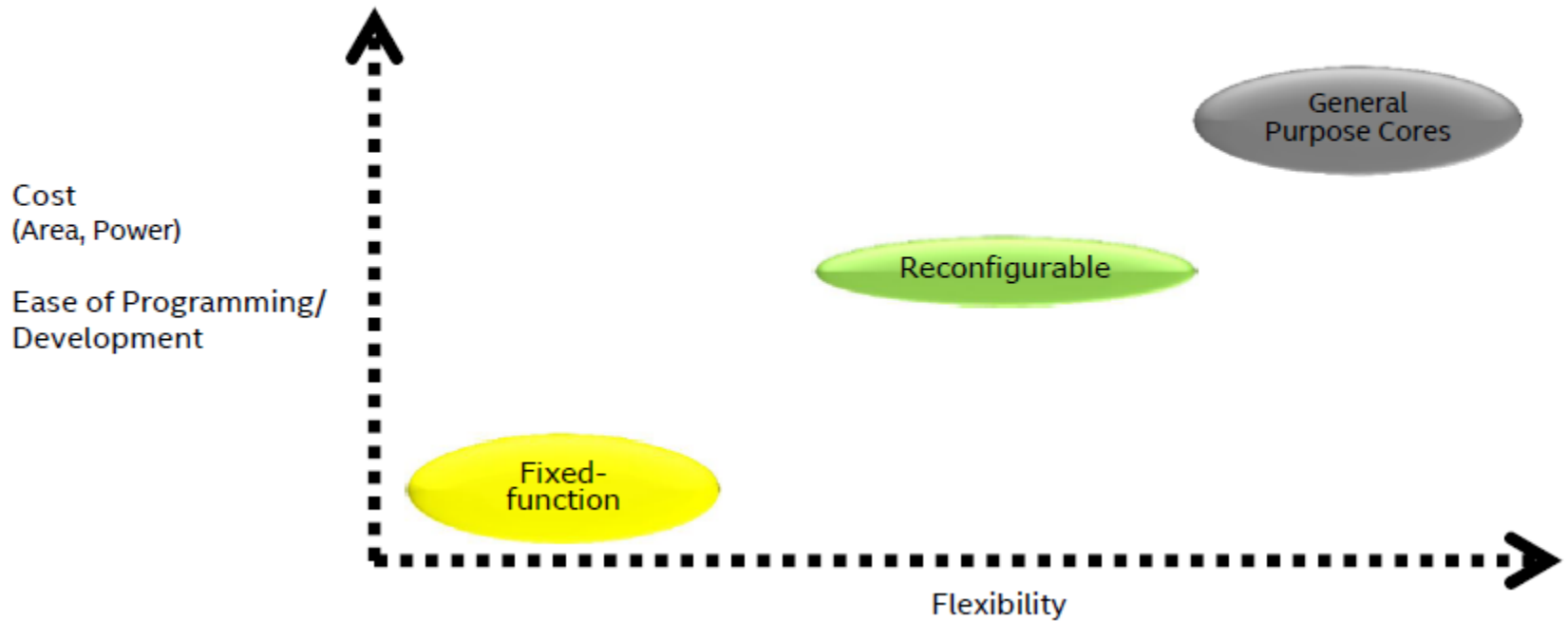
- Nagyságrendi növekedés az adatforgalomban
- Magasabb követelmények a válaszidők, késleltetés (latency) csökkentésére
- Nagyobb feldolgozási sebességigény
- **Alternatívák: CPU-GPU-FPGA-ASIC**



Alternatívák és opciók

- **Működési hatékonyság: Művelet/W, Művelet/\$**
- **Programozási hatékonyság: Erőforrás, Költség**

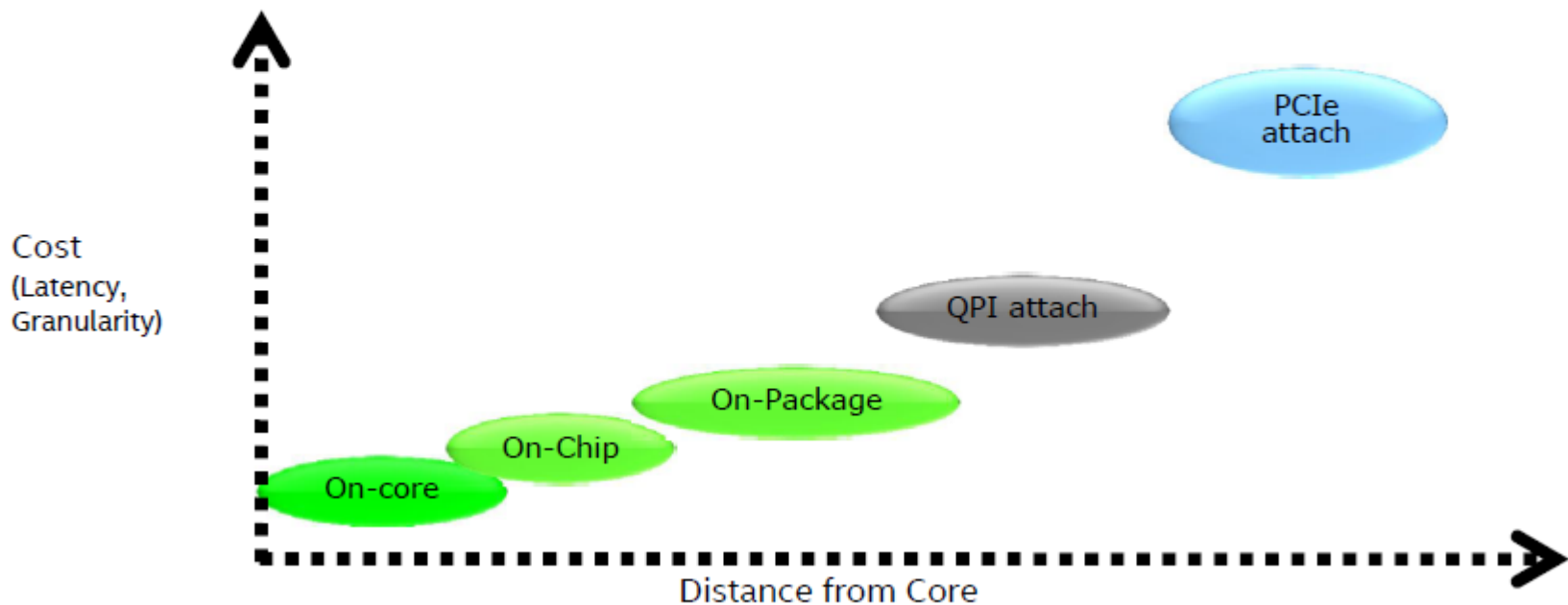
Accelerator Architecture



Alternatívák és opciók

- Rendszerbeillesztés helye, módja,
- Kapcsolat a Host és a gyorsító között

Accelerator Attach

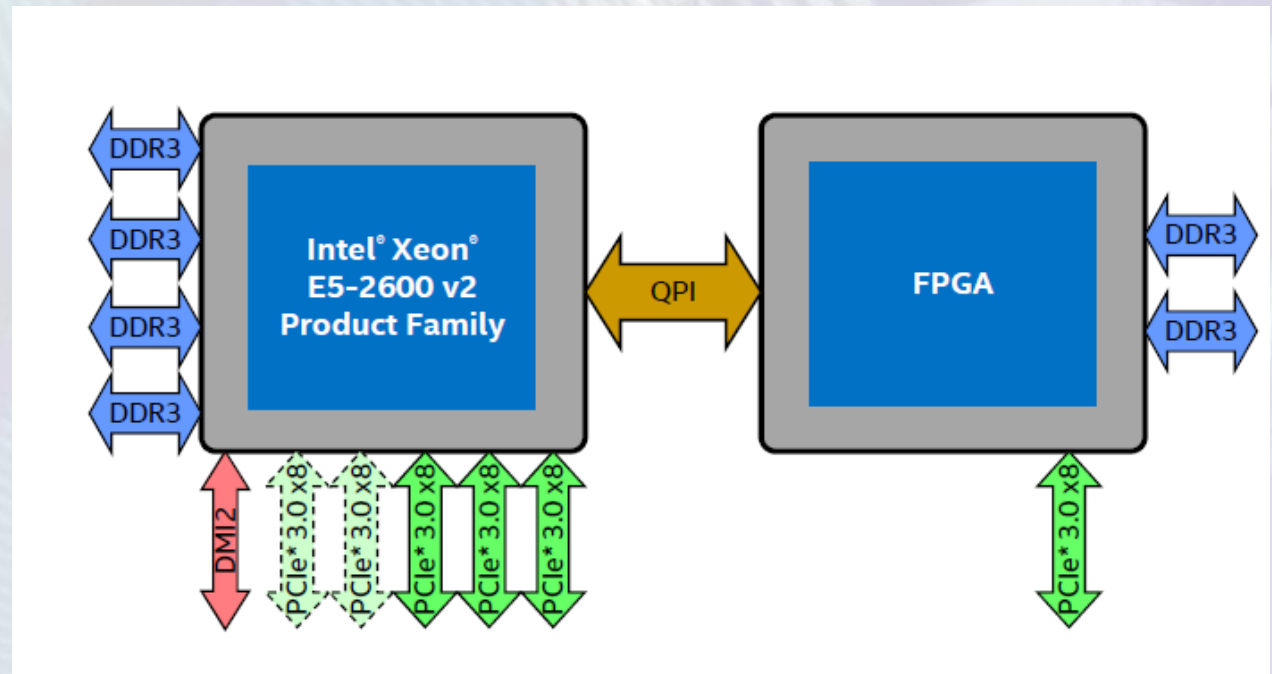


Alternatívák és opciók

- **Programozási és használati modell**
- **Adathozzáférési modell**
 - Közvetlen IO csatornán keresztül
 - Megosztott virtuális memóriában
 - Nem kell adat buffereket másolgatni...)
- **Együttműködési modell**
 - Off-load: független működés
 - Hibrid: az algoritmus részei megosztva működnek a host és a gyorsító erőforrásain

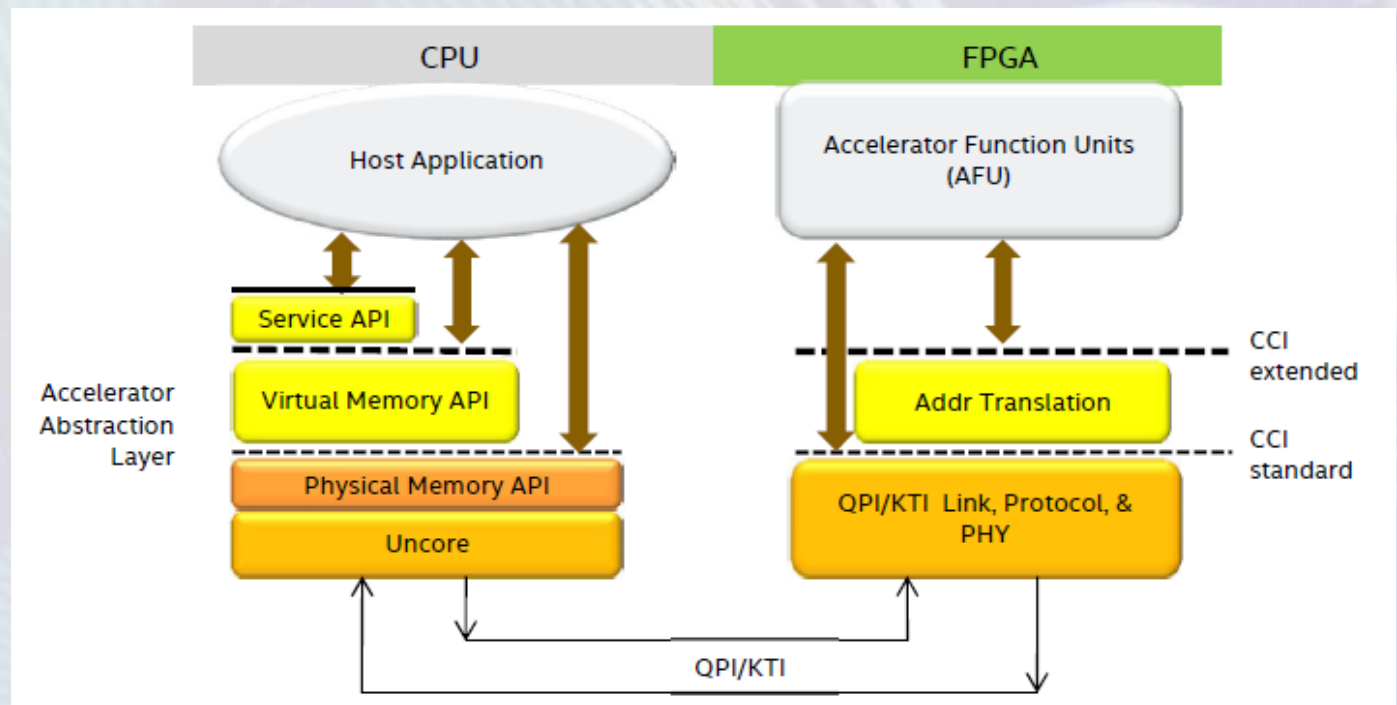
Intel megoldás

- **Xeon + FPGA QPI interfésszel**
 - Külső kapcsolat QPI 6,4GT/s sebességgel
 - Lokális memória, bufferelt adatkezelés
 - AAL Acceleration Abstraction Layer



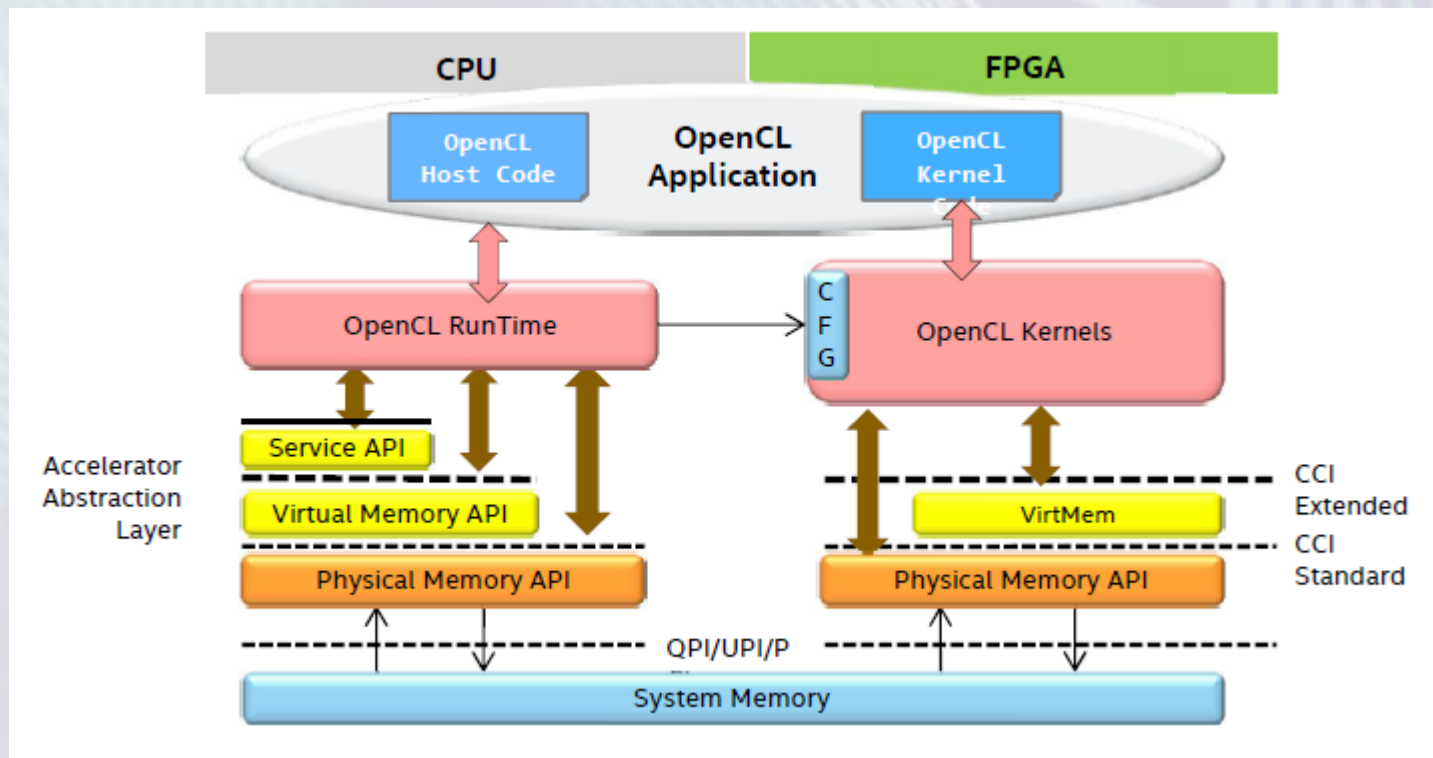
Intel megoldás

- **Programozási interfész**
 - AFU Accelerator Functional Units
 - CCI Core Cache Interface (másoknál CCIX)
 - Egységes memória kezelés a teljes rendszerben



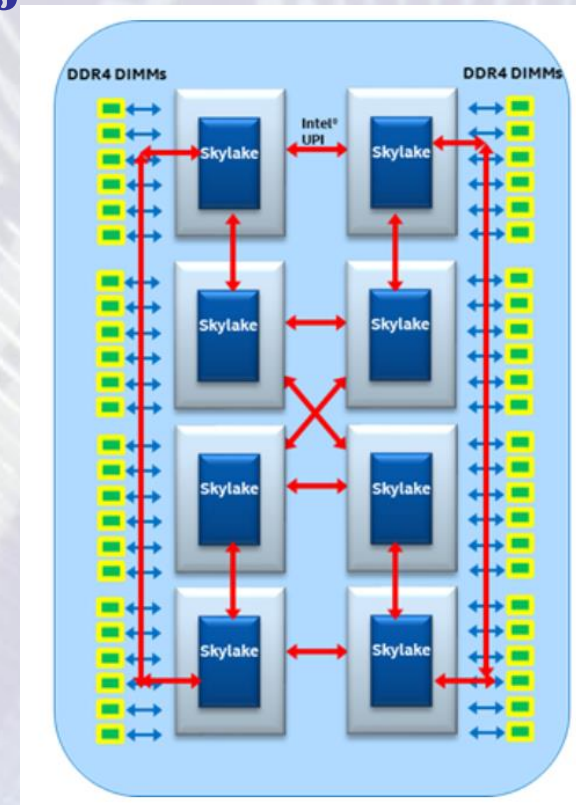
Intel megoldás

- Programozási interfész OpenCL esetén
 - Egységes alkalmazói kód
 - Hordozható CPU+FPGA generációk között



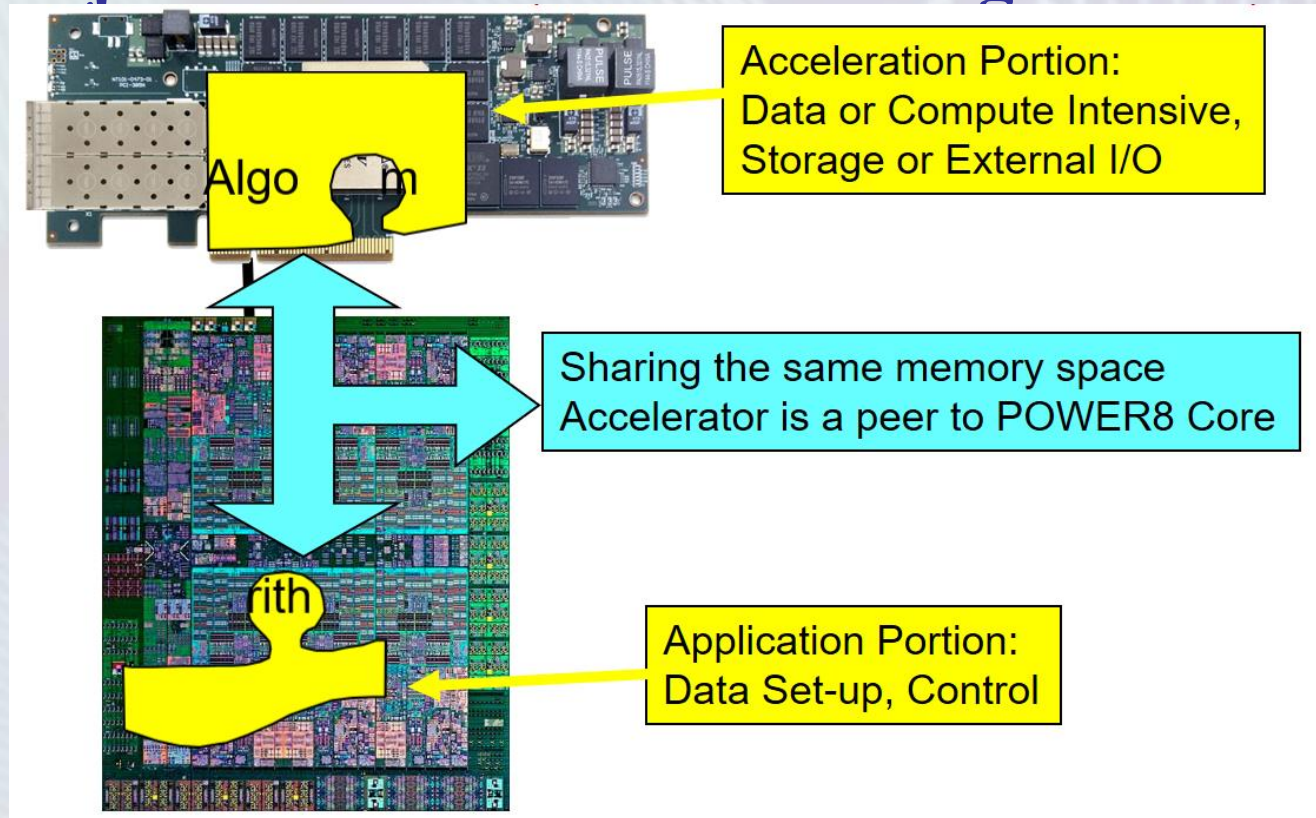
Intel UPI

- Ujabb generációs CPU-k Skylake és utána
- UPI Ultra Path Interconnect – Teljes 2D hálózat
- Cache-koherens kapcsolat
- Elsősorban a core-ok között
- De külső egységek is felfűzhetők
 - Gyorsítók
 - Speciális tároló egységek



IBM CAPI

- CAPI Coherent Accelerator Processor Interface
- Szabványos PCIe fizikai kapcsolat, a POWER8/9 CPU biztosítja a rendszerbe illeszthetőséget



IBM CAPI

- **A cél a gyorsító használatát előkészítő felesleges műveletek (pl. memória adatmásolás) kiiktatása**
- **Koherens memória elérés nélkül az eszközzel felelős a memória leképezéséért, változók, bemeneti adatok mozgatásáért**
- **Az adatok egyidőben több példányban léteznek**
 - Eredeti (OS) adatok a rendszermemóriában
 - Átmeneti adatok az eszközzel puferrében
 - Munka adatok a gyorsító lokális memóriájában
- **Sok felesleges adminisztráció a driver-ben**

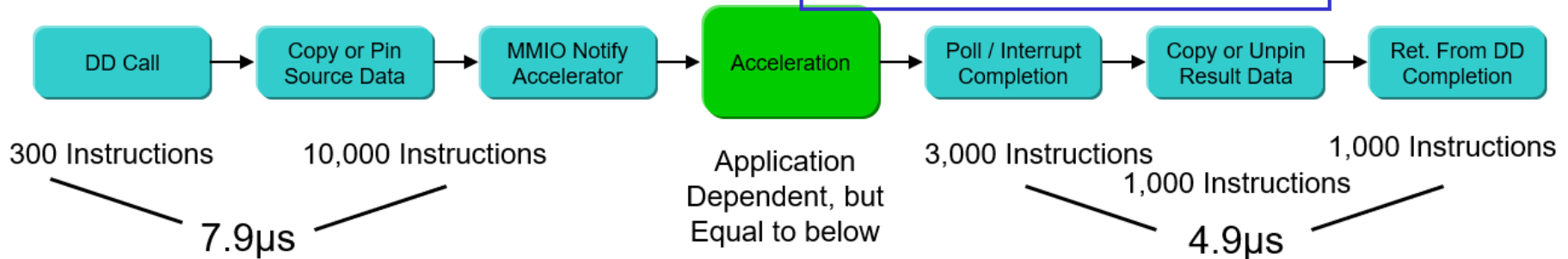
IBM CAPI

- **A CAPI megoldásban definiáltak egy PSL POWER Service Layer-t**
- **Ez a HW modul az FPGA-ban kerül kialakításra, közvetlenül a POWER8/9 CPU végrehajtó egységével/memória vezérlőjével áll kapcsolatban**
- **Nincs felesleges adatmozgatás, a gyorsító a változókat, bemeneti adatokat közvetlenül a rendszer memóriából éri el, és az eredményt oda írja vissza**
- **Az eszközközkezelő a működés során nincs használva**

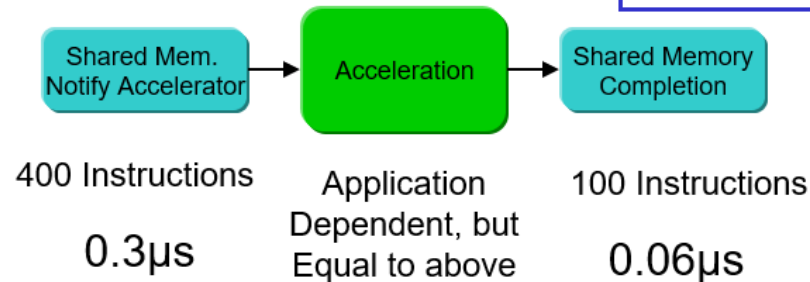
IBM CAPI

- Az adatelőkészítés folyamata

Typical PCIe Model Flow: Total ~13 μ s for data prep

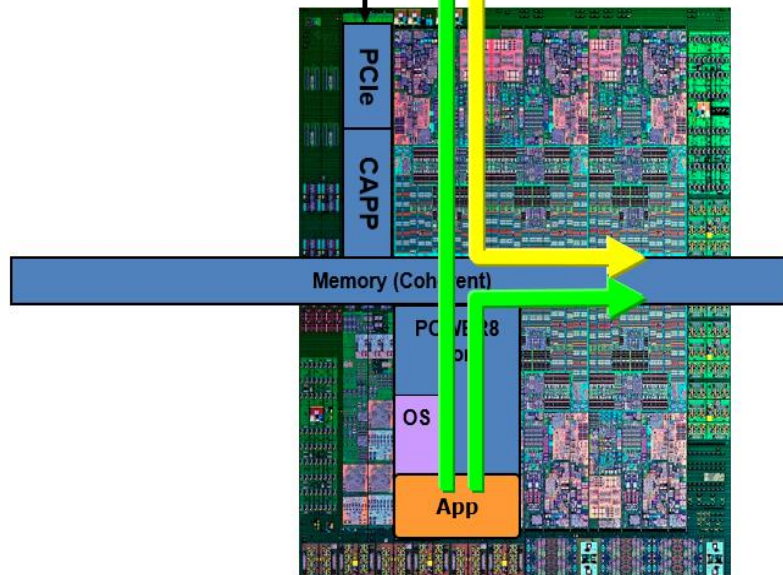
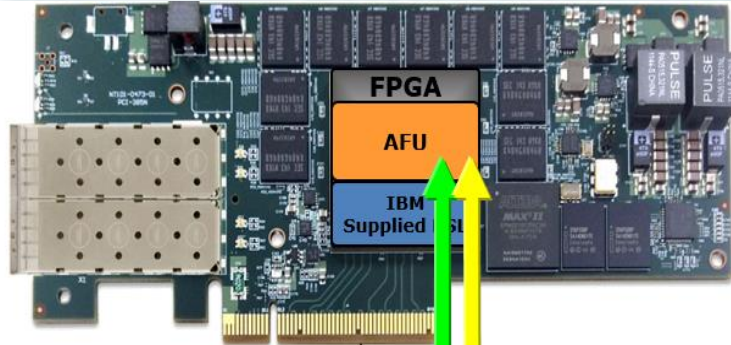


Flow with a Coherent Model: Total 0.36 μ s

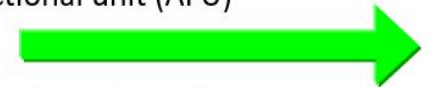


IBM CAPI

- A működésben részt vevő funkcionális egységek



- Proprietary hardware to enable coherent acceleration
- Operating system enablement
 - Ubuntu LE
 - Libcxl function calls
- Customer application and accelerator
- Application sets up data and calls the accelerator functional unit (AFU)
- AFU reads and writes coherent data across the PCIe and communicates with the application
 - PSL cache holds coherent data for quick AFU access



IBM CAPI

- **CAIA Coherent Accelerator Interface Architecture**

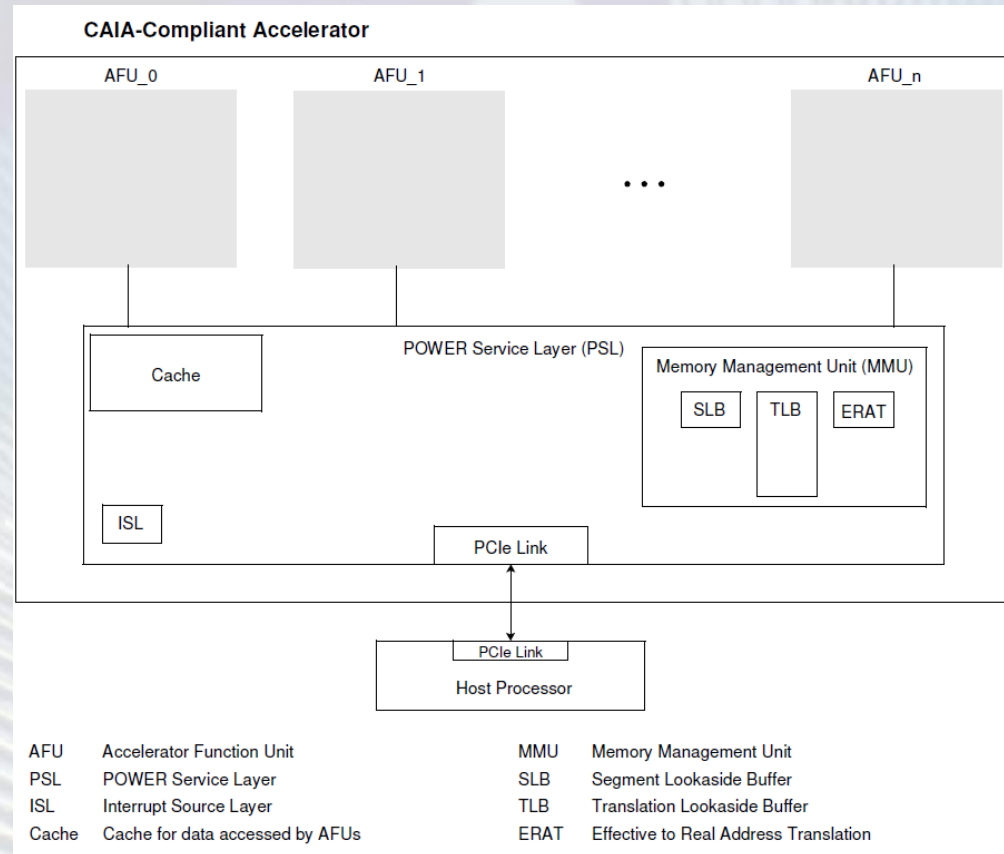
- Illeszkedik a CAPI Coherent Accelerator Processor Interface-hez

- A lényeg a PSL egység, amiben CACHE, MMU, és ISL is van !

Ezt az IBM biztosítja

- A CAIA egyidőben több Accelerator Functional Unit-ot (AFU) is tartalmazhat

- Valódi dinamikus rekonfiguráció lehetősége

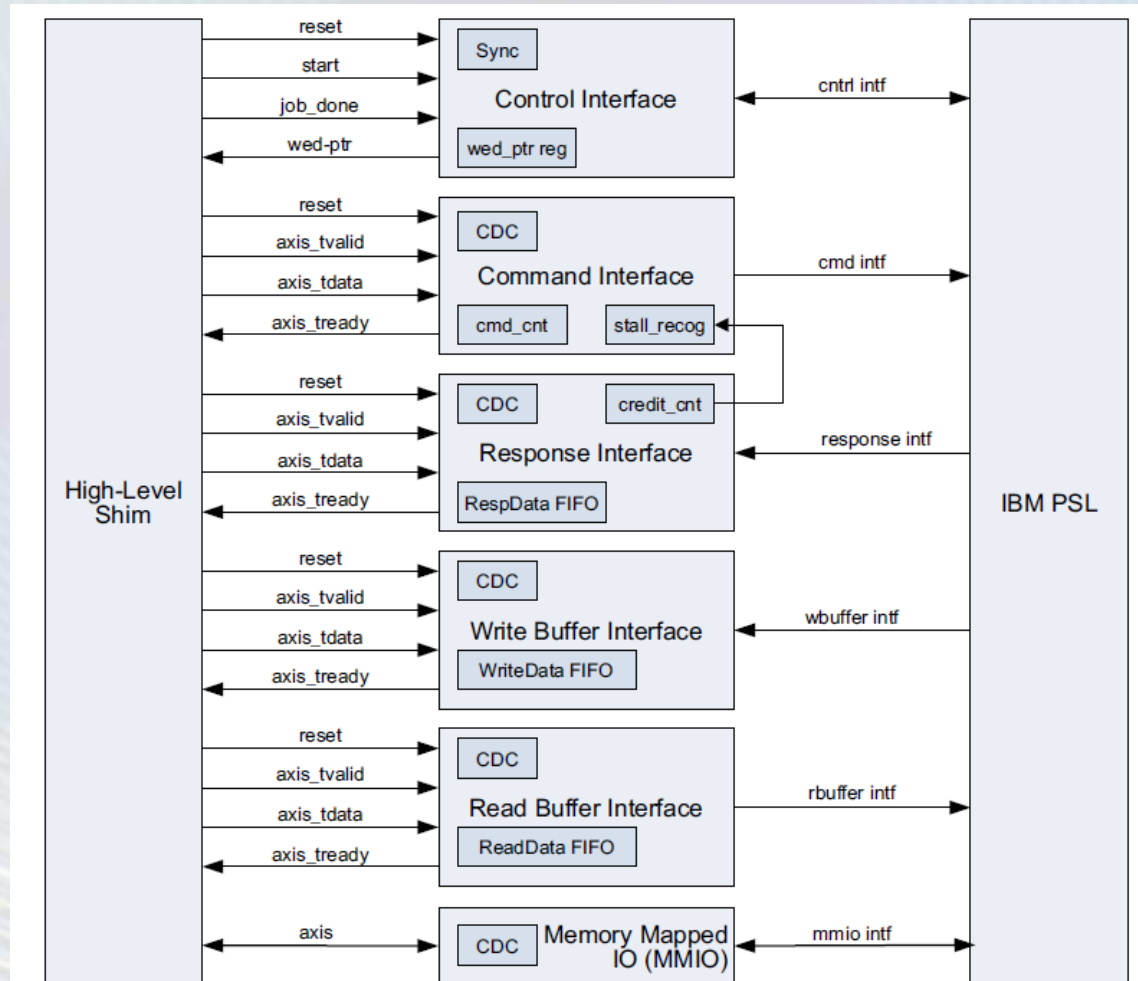


Xilinx IBM CAPI interfész

- **Kapcsolatot teremt a PSL és az AFU logika között**
- **Az FPGA-n belül a szabványos AXI interfészre épül**
- **A PSL 5 interfész csatornát használ:**
 - Vezérlő interfész
 - Parancs interfész
 - Válasz interfész
 - Írasi és olvasási adatbuffer interfészek
 - MMIO interfész (a gyorsító regisztereinek kezelésére)
 - Megjegyzés: az adatátvitel történhet out-of-order módon, az érvényes adatokat

Xilinx IBM CAPI interfész

- Az interfészek egy-egy AXI Streamre képeződnek
- Kivéve MMIO, ami AXI-Lite



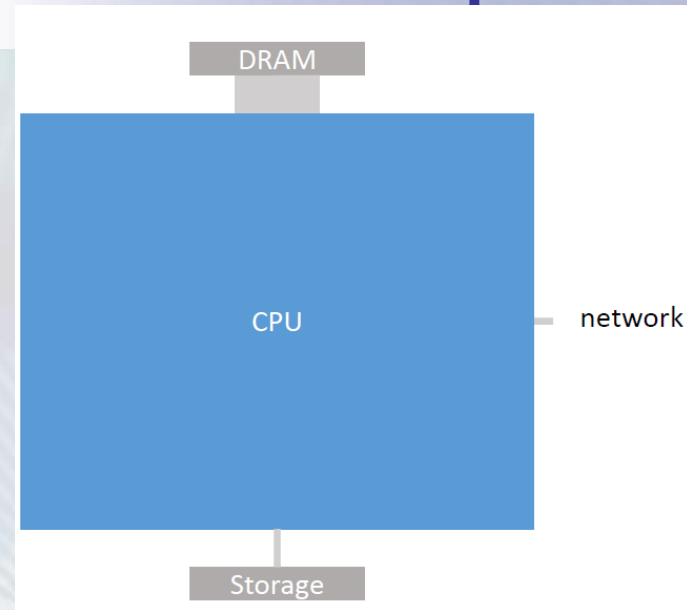
X17224-072016

CAPI továbblépés

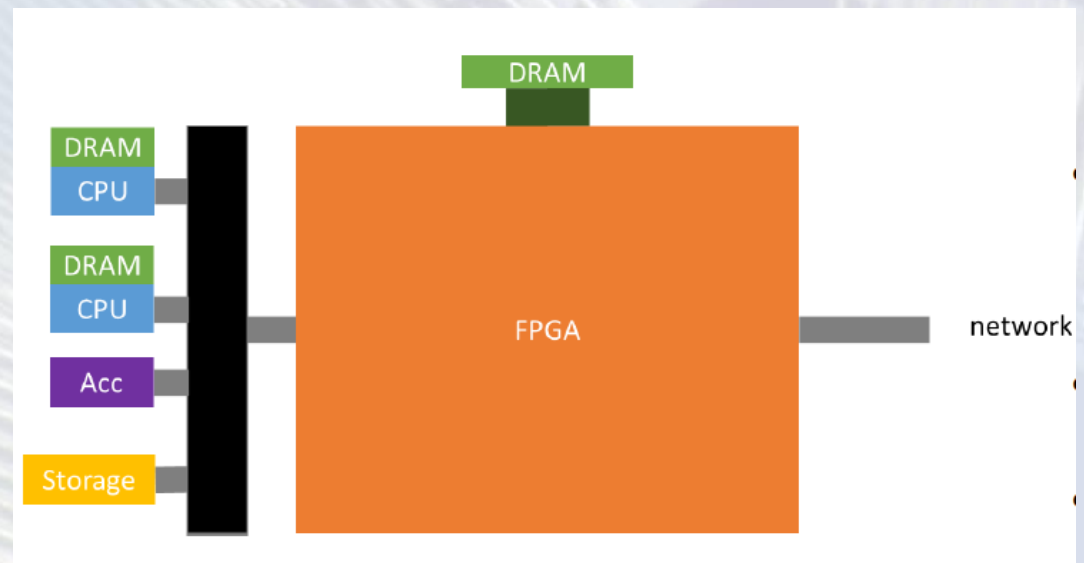
- **CCIX Cache Coherent Interconnect for Accelerator**
- **Általános célú interfész specifikáció tetszőleges CPU + gyorsító közé**
- **Nyílt ipari szabvány kezdeményezés (AMD, ARM, Huawei, IBM, Mellanox, Xilinx)**
- **Nagy sávszélesség: 25Gb/s**
- **Bővítés memóriák (HMC, HBM) és tárolóeszközök (NVMe) felé**
- **Egyelőre kísérleti technológia**

Microsoft Azure adatközpont

- **Hagyományos számítógép modell**

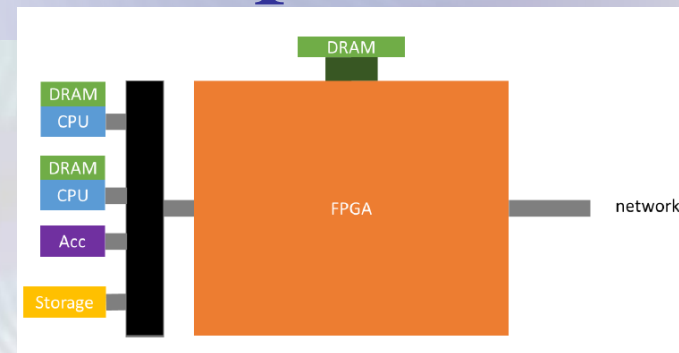


- **Microsoft DataCenter Computer**



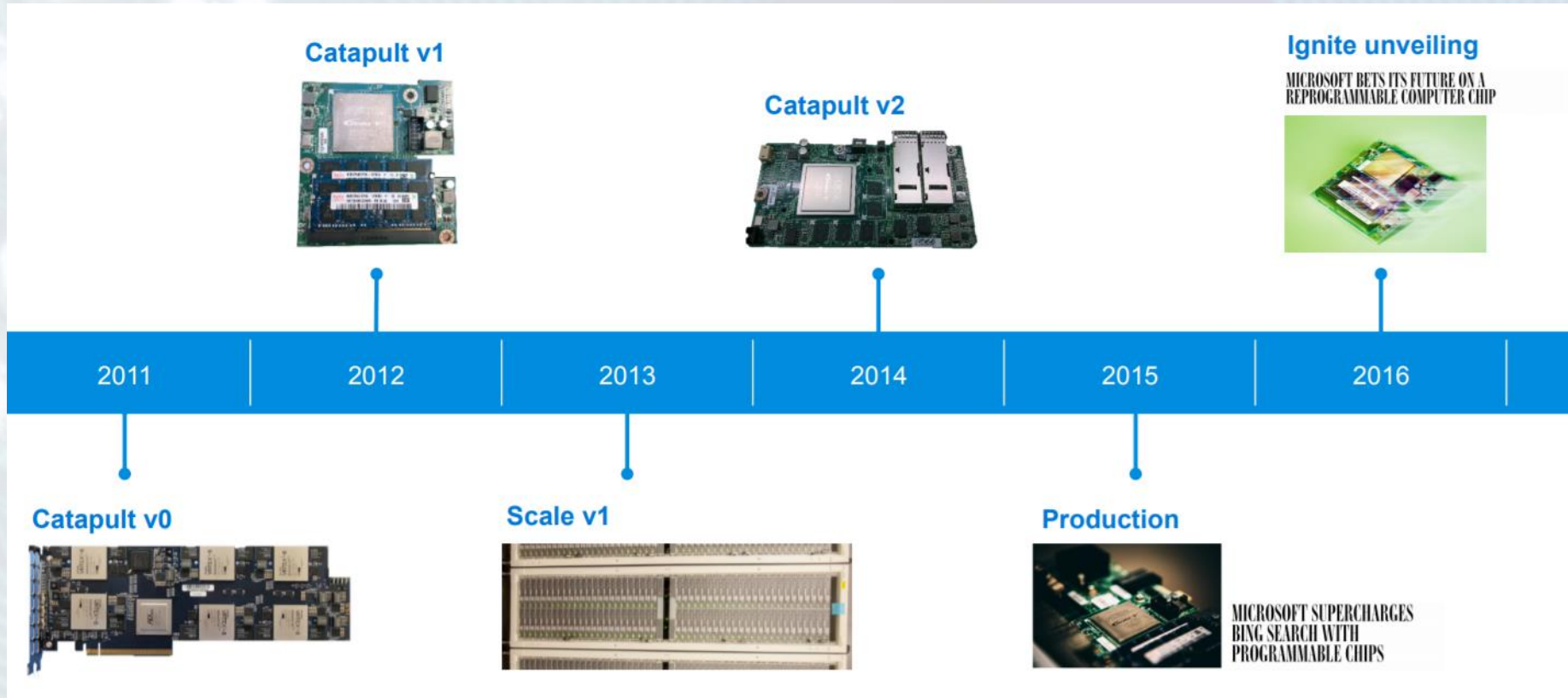
Microsoft Azure adatközpont

- **Előnyök, tulajdonságok**
- **Az FPGA minden csomagot feldolgoz, „látja a forgalmat”**
- **Egyszerű feladatokat teljesen ellát**
 - A csomagok a CPU-ba el sem jutnak
 - Közvetlenül kezeli a rendszerememóriát
- **A CPU szerepe csak a nagykomplexitású feladatok ellátása**
- **Jellemző feladatok**
 - Elosztott gépi tanulás
 - Software Defined Networking



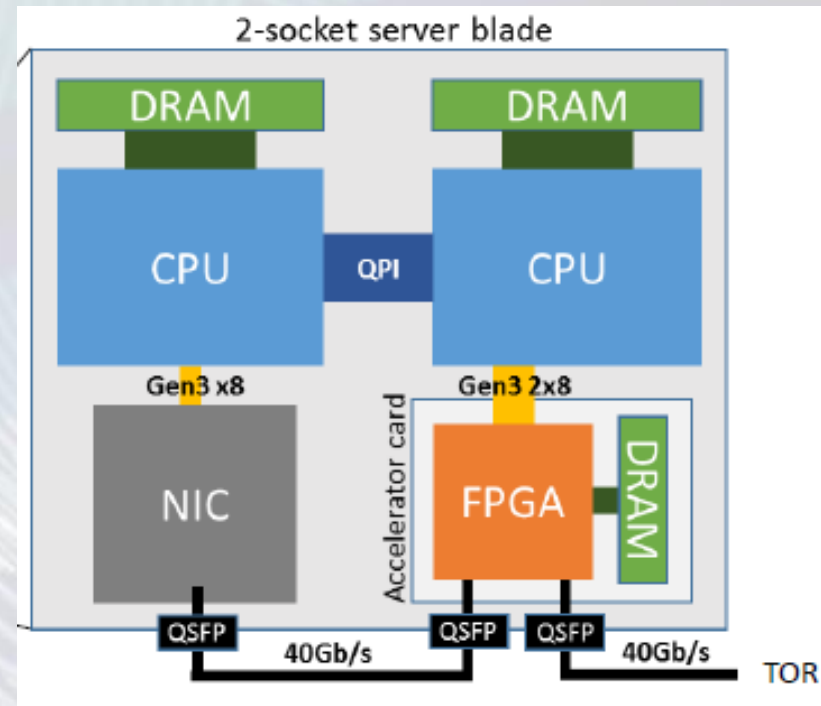
Microsoft Azure adatközpont

- A Catapult projekt eddigi fejlődése



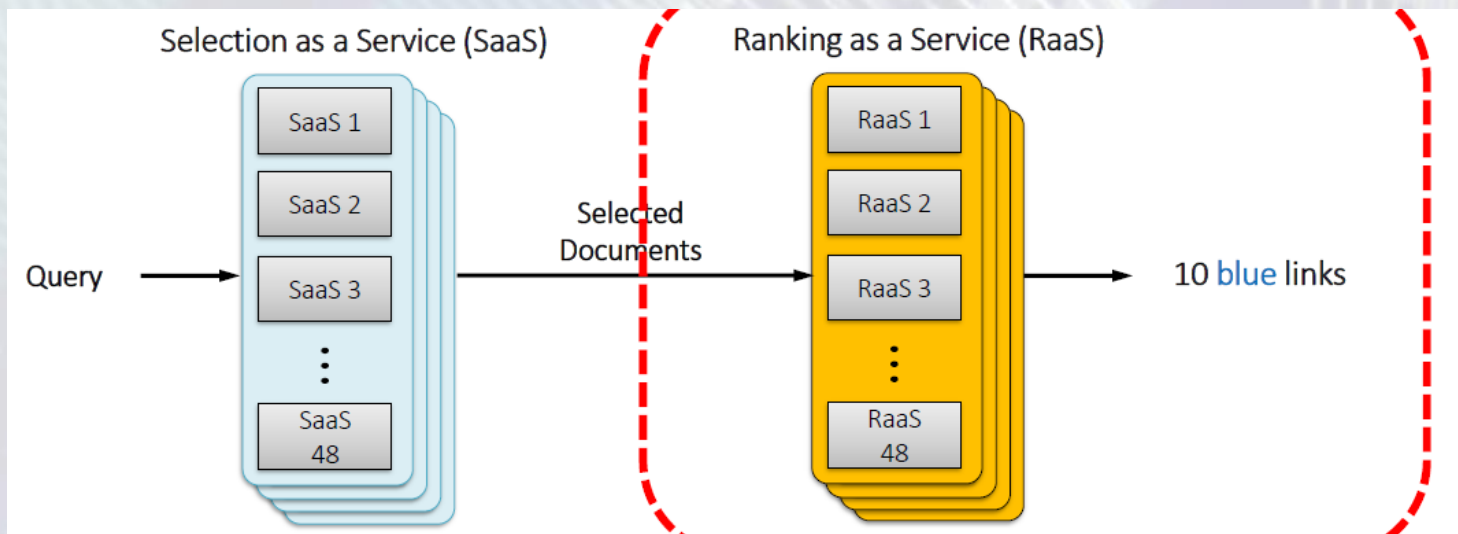
Microsoft Azure adatközpont

- A kialakított architektúra
- Teljesen flexibilis felépítés
- Lokális gyorsítóeszköz
- Távoli gyorsítóeszköz
- Hálózati/tárolási gyorsító
- **ToR = Top-of-Rack** kommunikációs hálózat



Microsoft Azure adatközpont

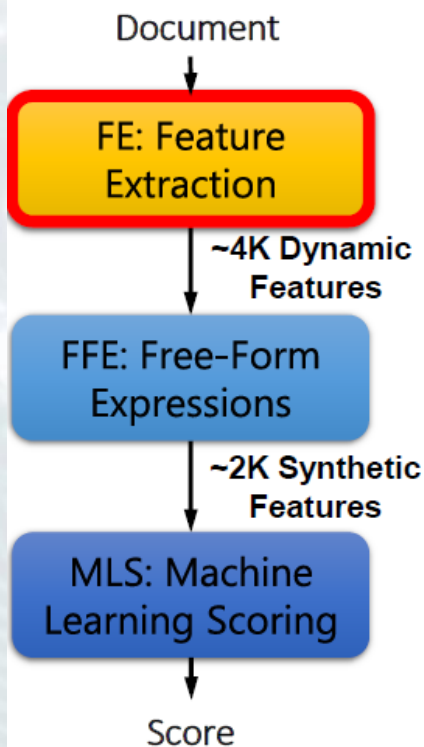
- **Lokális gyorsító alkalmazás**
 - A korábban már említett Intel CPU + FPGA megoldástól eltérően PCIe kapcsolat (a QPI foglalt)
 - A rendszermemória elérés PCIe-en keresztül
 - Bing keresésekhez, kiválasztás, sorbarendezés



Microsoft Azure adatközpont

FE: Feature Extraction

Query: "FPGA Configuration"



NumberOfOccurrences_0 = 7

NumberOfOccurrences_1 = 4

NumberOfTuples_0_1 = 1

Field-programmable gate array

From Wikipedia, the free encyclopedia
(Redirected from **FPGAs**)

A **field-programmable gate array** **FPGA** is an **integrated circuit** designed to be configured by the customer or designer after manufacturing—hence **user-programmable**. The **FPGA configuration** is generally specified using a **hardware description language** (HDL), similar to that used for an **application-specific integrated circuit** (ASIC) (**circuit diagrams** were previously used to specify the configuration as they were for ASICs, but this is increasingly rare). **FPGAs** can be used to implement any logical function that an ASIC could perform. The ability to update the functionality after shipping, **partial re-configuration** of a portion of the design^[1] and the low non-recurring engineering costs relative to an ASIC design (notwithstanding the generally higher unit cost), offer advantages for many applications.^[2]

FPGAs contain **programmable logic** components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together"—somewhat like many (changeable) logic gates that can be inter-wired in (many) different **configurations**. Logic blocks can be configured to perform complex **combinational functions**, or merely simple **logic gates** like **AND** and **XOR**. In most **FPGAs**, the logic blocks also include memory elements, which may be simple **flip-flops** or more complete blocks of memory.^[1]

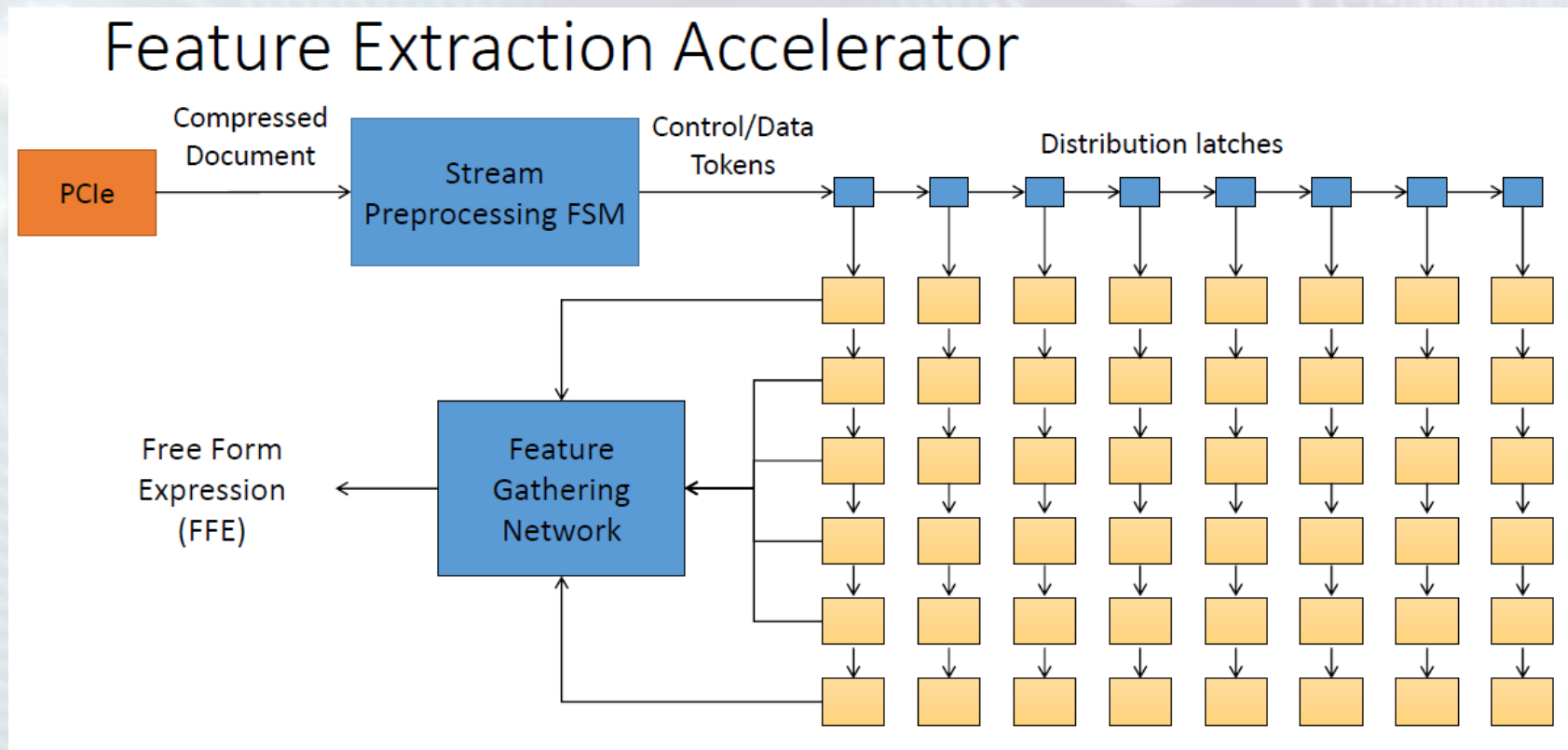
In addition to digital functions, some **FPGAs** have analog features. The most common analog feature is programmable **slew rate** and drive strength on each output pin, allowing the engineer to set slow rates on lightly loaded pins that would otherwise **ring** unacceptably, and to set stronger, faster rates on heavily loaded pins on high-

H2RC Nov 14, 2016

14

Microsoft Azure adatközpont

- Lehet egyetlen FPGA teljes belső konfigurációja, vagy az erőforrások egy része

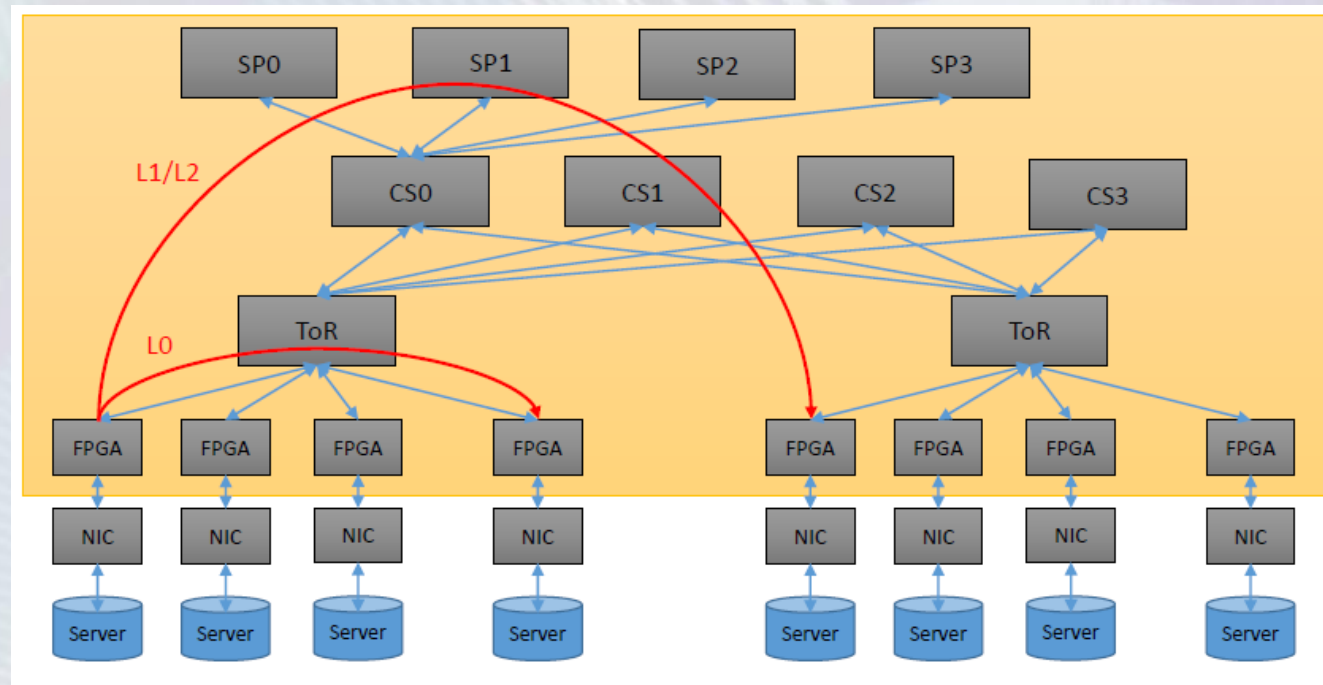


Microsoft Azure adatközpont

- **Távoli gyorsító alkalmazás**
 - A terhelések megvalósíthatósága nem mindig pontosan illeszkedik az FPGA méretéhez
 - A kihasználatlan területek elvesztegetett FPGA erőforrások vagy és elvesztegetett teljesítmény
 - Opció 1. Csökkentett teljesítmény/erőforrásigény, hogy elférjen egy FPGA-ba
 - Opció 2. Hozzáférés több szervertől hálózaton
 - Megoldás: ToR hálózati hozzáférés

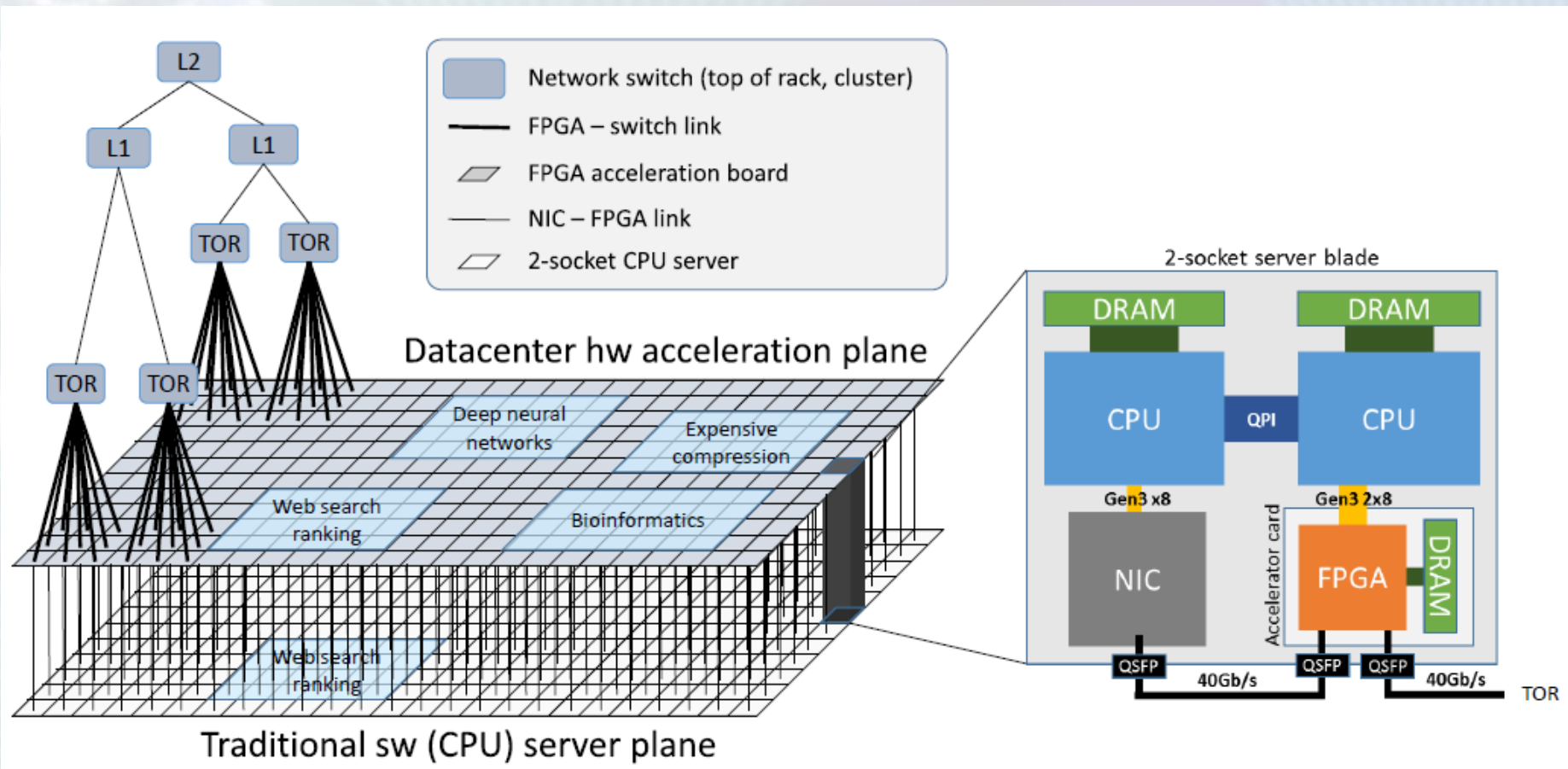
Microsoft Azure adatközpont

- **FPGA-k közötti közvetlen hálózati kapcsolat**
- **Egy ToR alatt (6x8 = 48 FPGA) közvetlen nagysebességű L0 szintű kapcsolat**
- **Ezek felett L1 és L2 szintű hálózati eszközök vannak**



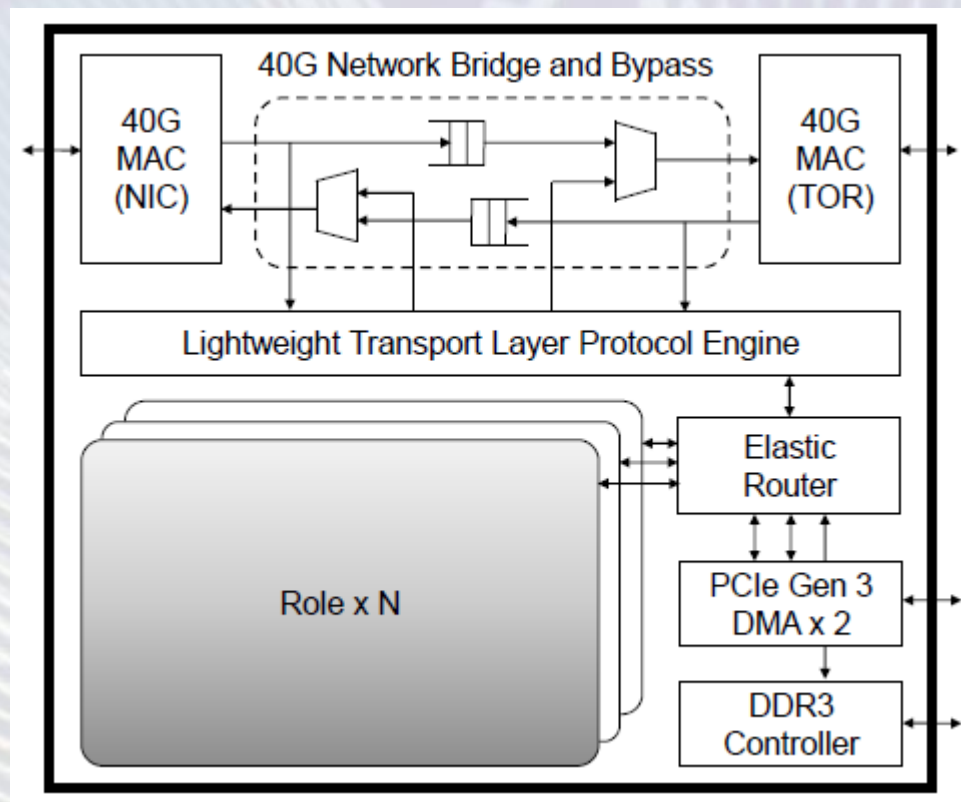
Microsoft Azure adatközpont

- **FPGA alapú konfigurálható felhő adatközpont**



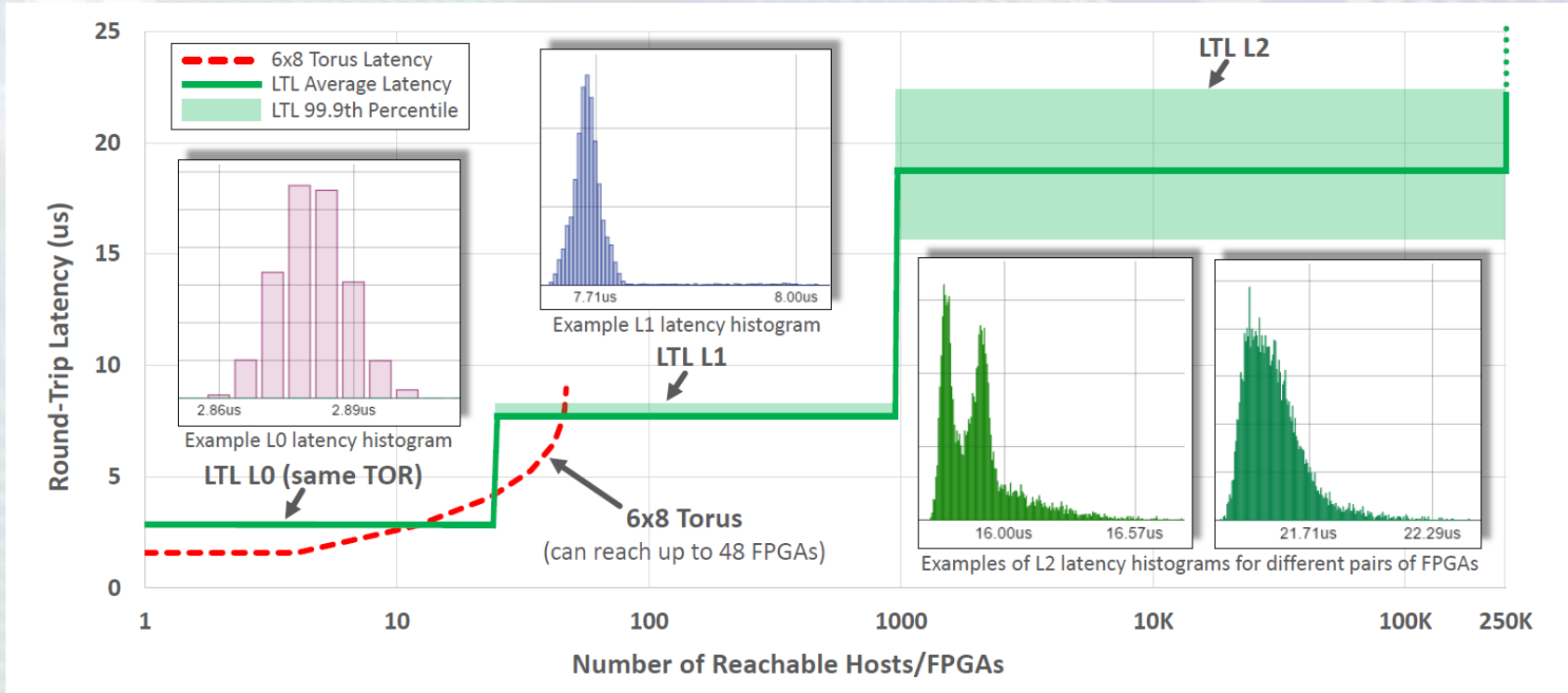
Microsoft Azure adatközpont

- Az FPGA-k belső felépítése
- Két komponens: Role, az alkalmazói logika, és a közös I/O és vezérlés
- Lokális memória 4GB
- Két 40Gb interfész
- PCIe IF DMA-val, közvetlen rendszer memória elérés
- Belső adatutak, belső hálózat



Microsoft Azure adatközpont

- **Lightweight Transport Layer Protocol Engine**
- **UDP és IP protokolt használ a hálózaton belül**
- **Bonyolult belső egység, megbízható adatátvitellel**

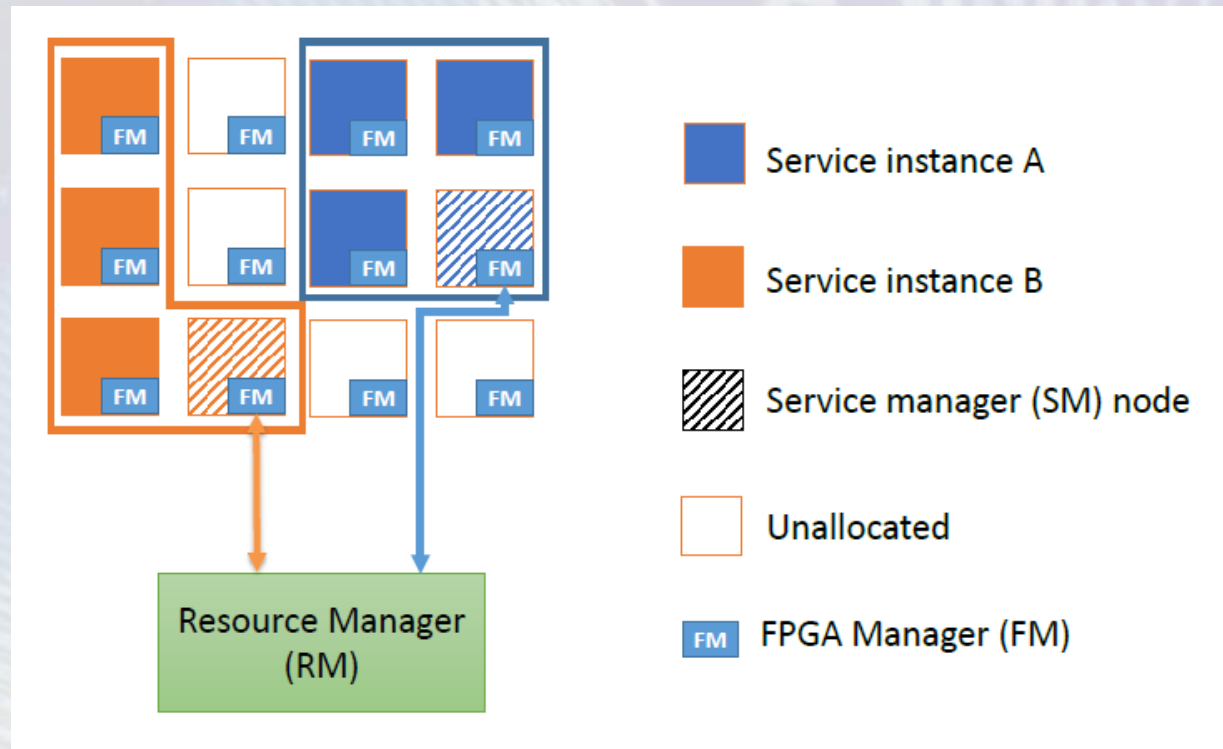


Microsoft Azure adatközpont

- **Lightweight Transport Layer Protocol Engine**
- **UDP és IP protokolt használ a hálózaton belül**
 - Bonyolult belső egység, megbízható adatátvitellel
- **Késleltetések**
 - L0 szinten, a ToR-on belül 2,88us (48 FPGA közvetlen elérése)
 - L1 szinten 7,7us, ezzel kb. 1000 FPGA érhető el
 - L2 szinten 16 us – 20 us, kb. 100 000+ FPGA felé
- **Valódi távoli és hálózati funkcionálisok**

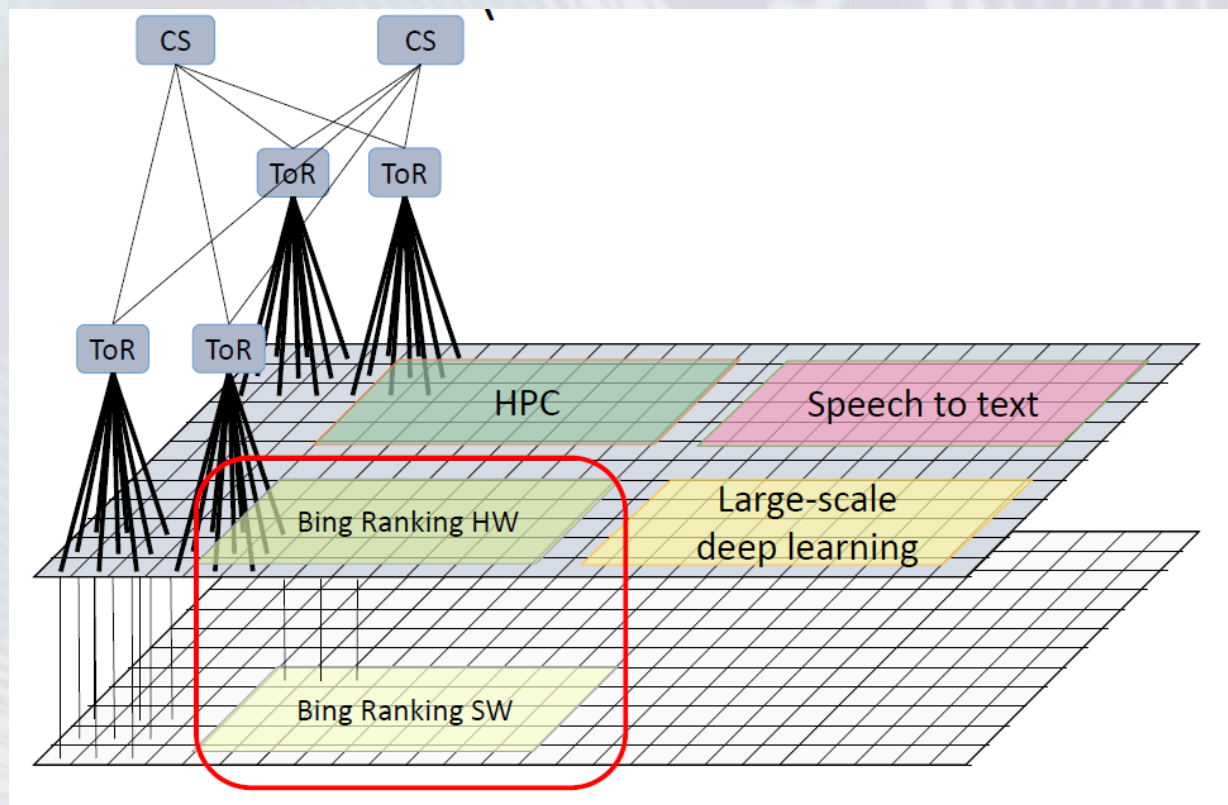
Microsoft Azure adatközpont

- **Hardware-as-a-Service rendszer kialakítás**
- **Különböző szolgáltatások telepíthetők az FPGA-kra**
- **Egy-egy komponens egy vagy több FPGA-t használ**
- **Hierarchikus felügyelet**
- **RM Resource Manager**
- **SM Service Manager**
- **FM FPGA Manager**



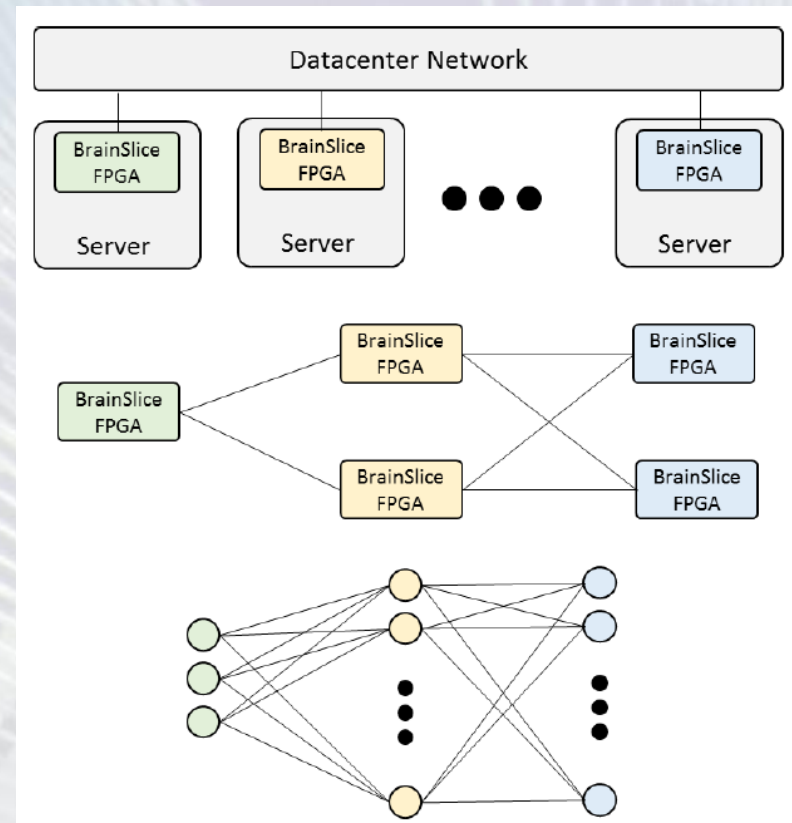
Microsoft Azure adatközpont

- **Hardware-as-a-Service rendszer kialakítás**
- **Különböző szolgáltatások telepíthetők az FPGA-kra**



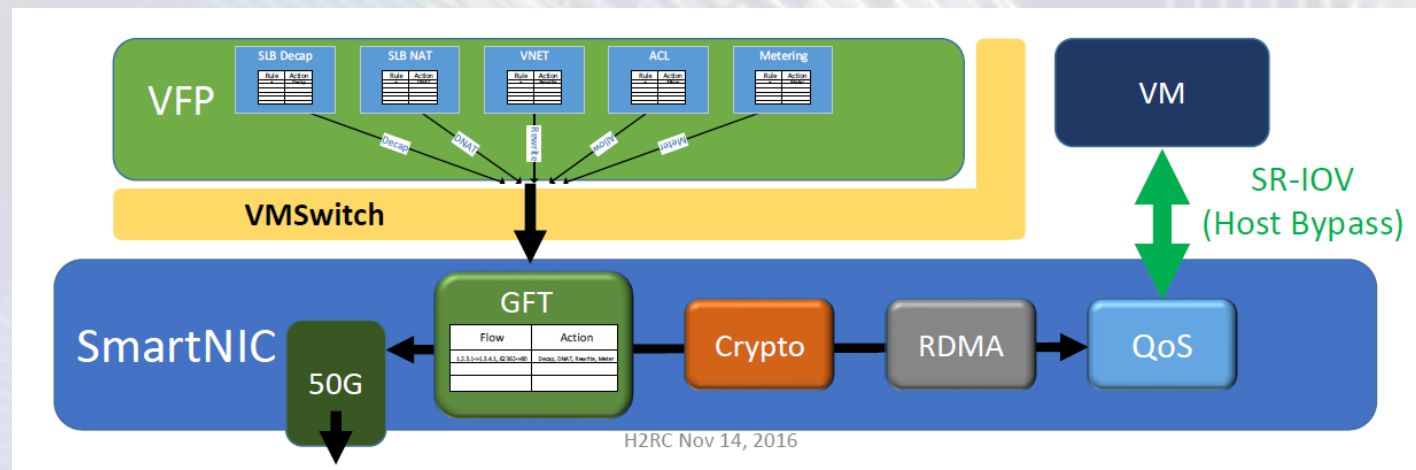
Microsoft Azure adatközpont

- **BrainWave: Extra méretű DNN hálózat**
- **Elosztott működés, akár 1000 FPGA között**
- **Optimális kapcsolati háló kapacitásra, késleltetésre, átviteli sáv szélességre**



Microsoft Azure adatközpont

- Hálózati intelligens NIC felhő szolgáltatásokra
- Az Azure a jelenlegi leggyorsabb publikált hálózat
 - 25Gb/s 25us késleltetéssel
- Az FPGA minden csomagot kezel, a fejléc módosításával (sebessége gyakorlatilag a hálózati sebességgel azonos „bare metal”)



Összegzés

- **Az adatközpontok területe egy intenzív fejlesztési ág**
- **Előtérbe került a heterogén számítási rendszerek használata**
- **Célok: Energiahatékonyság, adatátviteli sebesség, minimális válaszidő késleltetés**
- **Mindez felhasználóbarát fejlesztési technológiákkal**
- **Az FPGA-k használata fejlődik, egyre jobban beépülnek a rendszerekbe (Amazon, Nimbix, Baidu)**