Nový řídicí a dohledový systém pro experiment COMPASS

Martin Bodlák Vladimír Jarý[1] Josef Nový

[1]Fakulta jaderná a fyzikálně inženýrská
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
mailto:Vladimir.Jary@cern.ch

InstallFest 2012
Školicí centrum Silicon Hill, Praha
4. března 2012

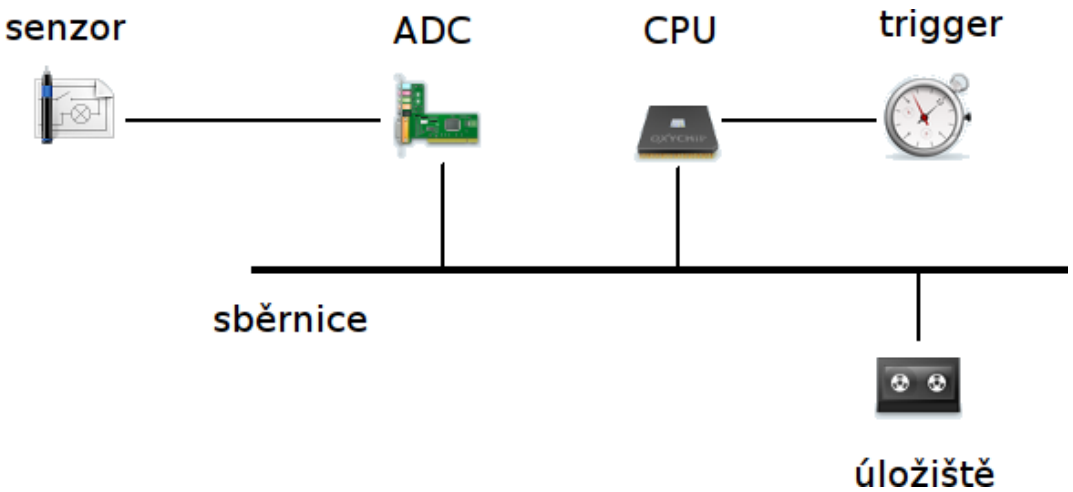
Přehled

Základní pojmy

- *událost*: data popisující průlet částice systémem detektorů
- role systému pro sběr dat (*data acquisition*, DAQ):
  1. načtení dat z detektorů (*readout*)
  2. sestavení úplných událostí z fragmentů pocházejících z různých kanálů (*event building*)
  3. zapsání událostí do trvalého úložiště (*data logging*)
  4. dohled a řízení (*monitoring*, *run control*)
- *trigger systém*: vybírá fyzikálně zajímavé události nebo zamítá nezajímavé události
- účinnost trigger systému:
  $\epsilon = N_{dobrych(vybranych)} / N_{dobrych} < 1$
- mrtvá doba (deadtime) systému:
  $D = t_{system\_je\_vytizen} / t_{celkovy}$ (je-li systém vytížen, nemůže přijímat žádné další události)

Příklad: systém s periodickým triggerem

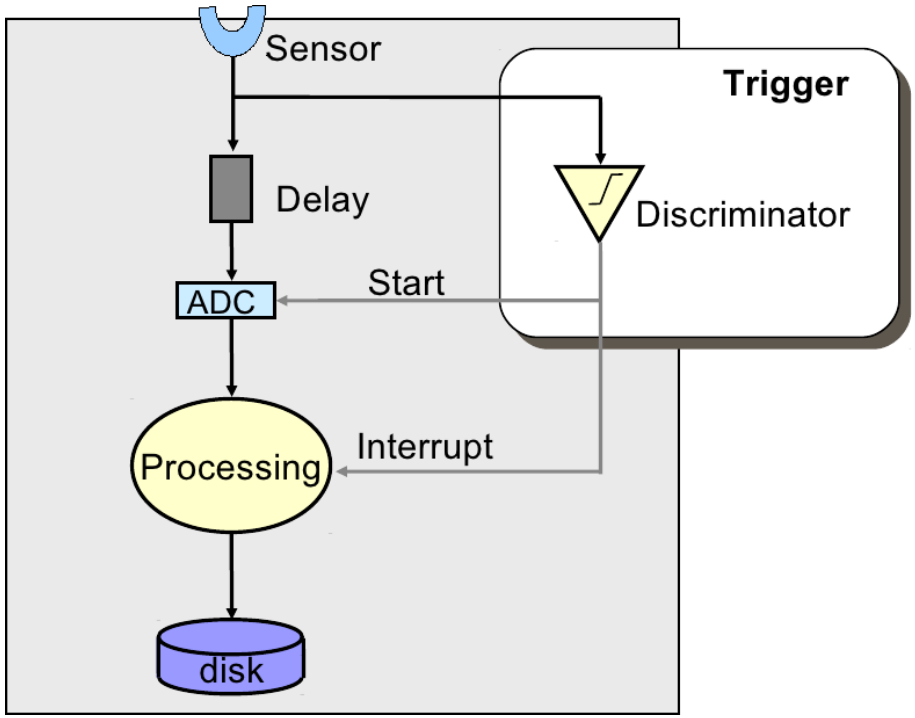
- v podstatě se jedná o vzorkování veličiny spojité v čase
- A/D převodník digitalizuje data, CPU je načítá a ukládá
- frekvence triggeru dána dobou zpracování události:
  - je-li potřeba na zpracování 1 ms $\Rightarrow$ $f_{trigger} \leq 1$ kHz



Systém pro sběr dat řízený periodickým triggerem

Fyzikální trigger

- data z detektorů přichází náhodně a nepředvídatelně
- potřeba mít fyzikální trigger



Systém sbírající data řízený fyzikálním triggerem podle [3]

Problémy s fyzikálním triggerem

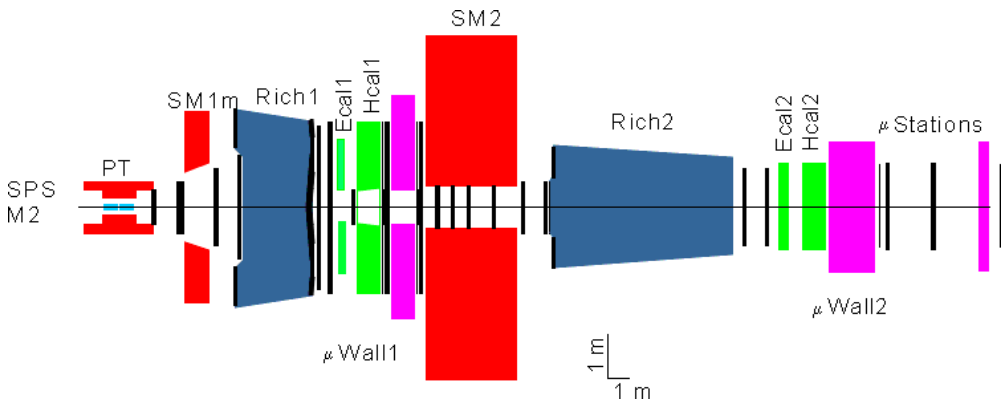
1. co dělat pokud nastane nová událost a systém je zaneprázdněný:
   - přidání busy logiky: je-li busy signál aktivní, systém nepřijímá další události
2. jak využít neaktivní dobu:
   - vyrovnávací paměti (FIFO): vyrovnávají fluktuace na vstupu a poskytují relativně stabilní datový tok na výstupu (*derandomizace*)
3. jak se vypořádat s velkým množstvím kanálů ($\sim 10^6$):
   - shromažďovací moduly (např. VME desky)
   - paralelní zpracování a ukládání dat
4. jak zajistit minimální deadtime:
   - A/D převodník pracuje na frekvenci $>> f$
   - zpracování a ukládání dat na frekvenci $\sim f$

Experiment COMPASS

- *COMPASS* = COmmon Muon and Proton Apparatus for Structure and Spectroscopy
- experiment s pevným terčem na urychlovači *SPS* v laboratoři *CERN*
- vědecký program schválen v roce 1997
  - studium struktury a spektroskopie hadronů
  - experimenty s mionovým svazkem
  - experimenty s hadronovým svazkem
- sběr dat od roku 2002
- momentálně začíná 2. fáze experimentu (COMPASS II)
- mezinárodní projekt: 250 vědců, 29 institucí, 11 zemí

Popis experimentu

- cyklus urychlovače SPS: svazek (*beam*) není spojitý, skládá se z úseků (*spills*, *bursts*)
  - systém pro sběr dat používá vyrovnávací paměti pro rozložení zátěže na celý cyklus urychlovače
- interakcí svazku s terčem vznikají sekundární částice
- průlet částic detekován systémem detektorů



Systém detektorů, svazek částic dopadá na terč zleva, převzato z [4]

Vrstvy systému pro sběr dat

1. *primární elektronika detektorů*
   - provádí předzpracování a digitalizaci analogových dat
   - celkem zhruba 250000 kanálů
2. moduly *GeSiCA*, *CATCH* (VME technologie)
   - provádí načítání a shromažďování dat
   - načítání aktivováno signály z TCS (*Trigger Control System*)
   - přidání hlavičky (identifikátor triggeru, časová značka)
3. ROB (*readout buffer*) servery
   - slouží jako vyrovnávací paměť pro efektivní využití cyklu SPS urychlovače
   - PCI karta spillbuffer (512 MB paměti)
4. EVB (*event builder*) servery
   - sestavení kompletních událostí
   - zapsání souborů s událostmi na trvalé úložiště
   - uložení metadat o událostech do Oracle DB
   - doplňkové úlohy: dohled na kvalitou dat, filtr

DATE (Data Acquisition and Test Environment)

- software navržený pro experiment ALICE na LHC
- řada úprav a doplňků pro COMPASS experiment
- základní dva procesory:
  1. lokální shromažďovač dat: provádí načítání dat z detektoru
  2. globální sběrač dat: sestavuje události z fragmentů vyprodukovaných lokálními shromažďovači dat
- dobře škálovatelný a flexibilní systém:
  - režim pp (vysoká frekvence interakcí, malé událostí)
  - režim PbPb (nízká frekvence interakcí, velké události)
  - DAQ experimentu ALICE $\times$ malé laboratorní experimenty s jedním procesorem
- testy výkonu:
  - načítání dat: 40 GB/s
  - sestavování událostí: 2.5 GB/s
  - záznam na úložiště: 1.25 GB/s

Problémy se současným systémem

Motivace:

- časem roste frekvence trigger systému, datový tok
- 260 TB dat zaznamenáno v roce 2002, v roce 2010 již 2 PB
- vyšší datový tok $\Rightarrow$ vyšší DAQ deadtime
- stárnoucí HW $\Rightarrow$ vyšší poruchovost
- vývoj PCI-Express verze spillbuffer karty nákladný
- chybějící vzdálené řízení

Návrh nového systému:

- nahradit síť ROB a EVB serverů vlastním HW
- tok dat, sestavování událostí řízeno HW
- software už pouze pro řízení a dohled
- možné použití i pro další experimenty (PANDA?)

Scintific Linux CERN 5

- založen na Red Hat Enterprise Linux 5
- CERN + Fermilab $\rightarrow$ Scientific Linux
- $\rightarrow$ Scientific Linux CERN
- stránky projektu `http://linuxsoft.cern.ch/`
- RPM balíčky, balíčkovací systém yum
- AFS klient
- repozitáře s vlastním softwarem
- RSS kanály organizace (např. CERN market)

Současná řídící místnost

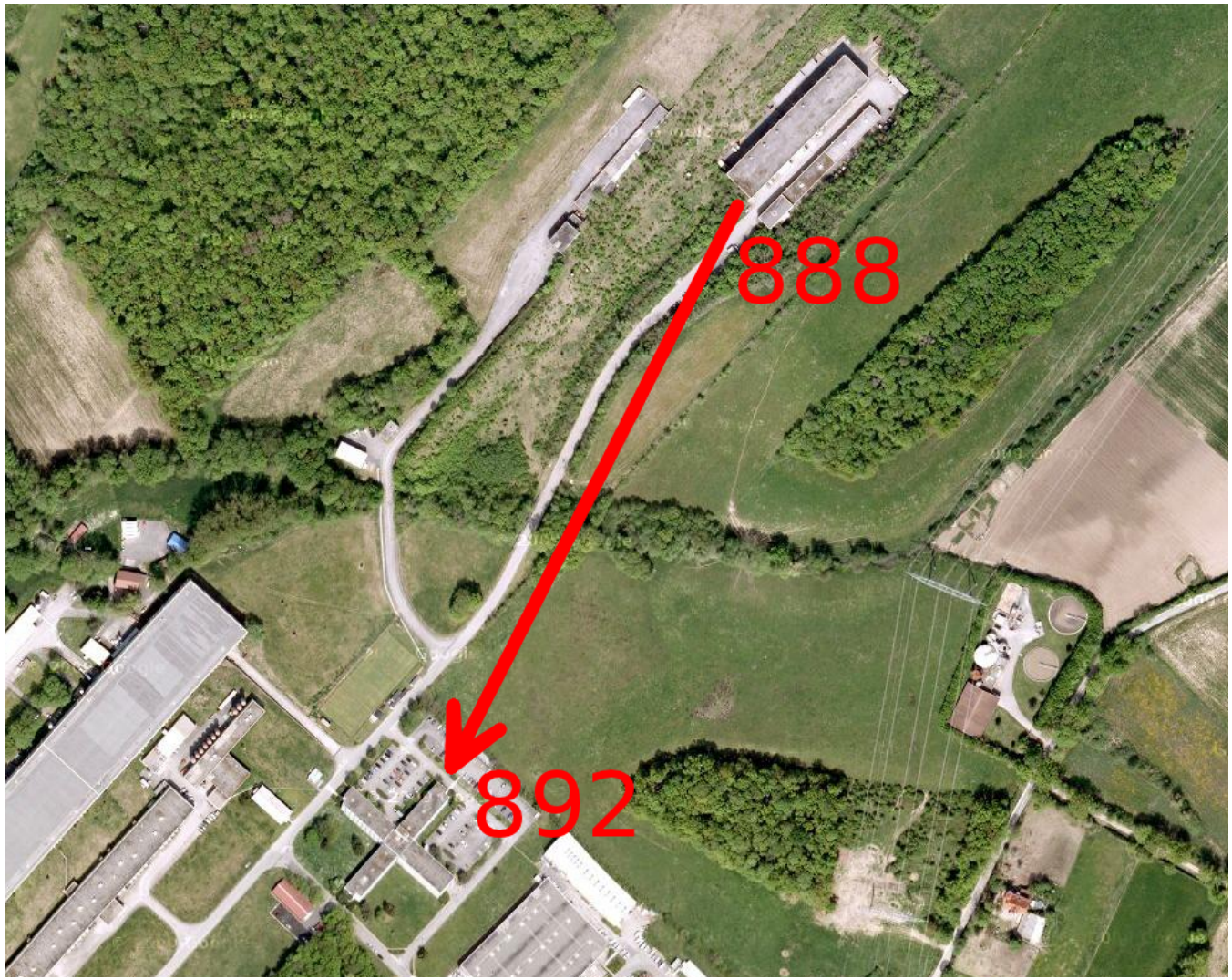
Současný velín umístěn přímo v hale experimentu COMPASS:

- Výhody
  - serverovna a detektory poblíž
  - možnost přímé fyzické kontroly plynových subsystémů
- Nevýhody
  - místo, kde prochází svazek $\Rightarrow$ problémy s radiací
  - horší dostupnost pro členy směny
  - horší ergonomie (hluk, . . .)

Technický koordinátor experimentu doporučil zřídit vzdálenou řídicí místnost.

Původní velín v hale experimentu

Přesun velínu



Vzdálená řídicí místnost

- vzdálený dohled a řízení experimentu COMPASS
- kancelářská budova
- vybavení:
  - 8x PC
  - 4x 24” LCD
  - 8x 22” LCD
  - IP kamery v hale experimentu
- napojeno na síť COMPASSu
- nový velín otestován
- ušetřeny finanční prostředky, které by bylo nutné investovat do přídavného stínění spektrometru
- před nasazením zbývá nainstalovat klimatizaci

Instalace nových stanic

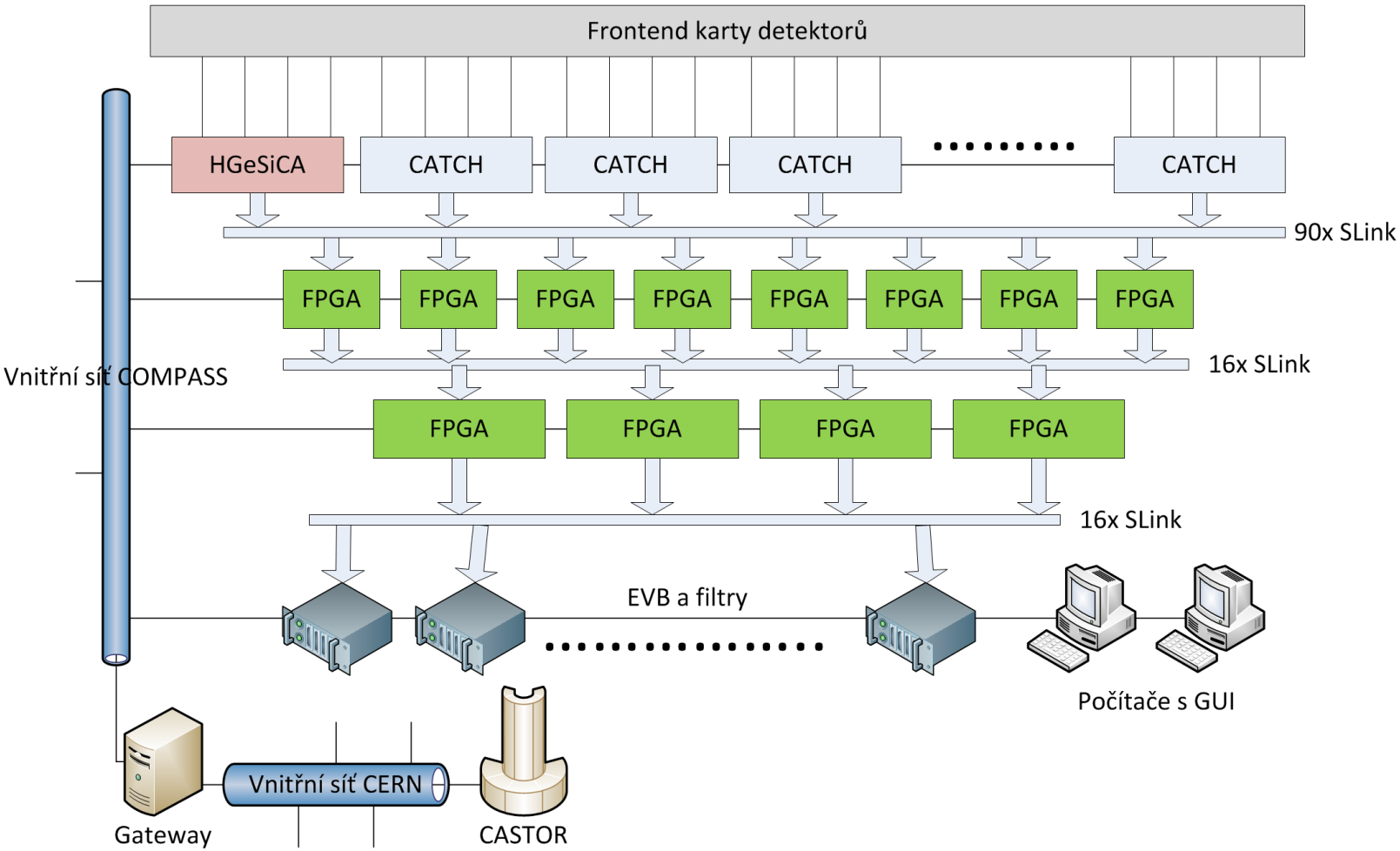
- Scientific Linux CERN 5, 32bit
- instalační program *Anaconda*
- bezobslužná instalace pomocí *kickstart* skriptů
- různé parametry instalace podle rolí
  - **řízení**
  - sestavování událostí
  - souborový server
  - databázový server
  - ...
- kickstart soubory publikované v centrální databázi *AIMS*
- Průběh:
  1. boot po síti, stažení kickstart skriptu z databáze
  2. předání skriptu programu Anaconda
  3. načtení parametrů, pokus o bezobslužnou instalaci

Nový velín v kancelářské budově

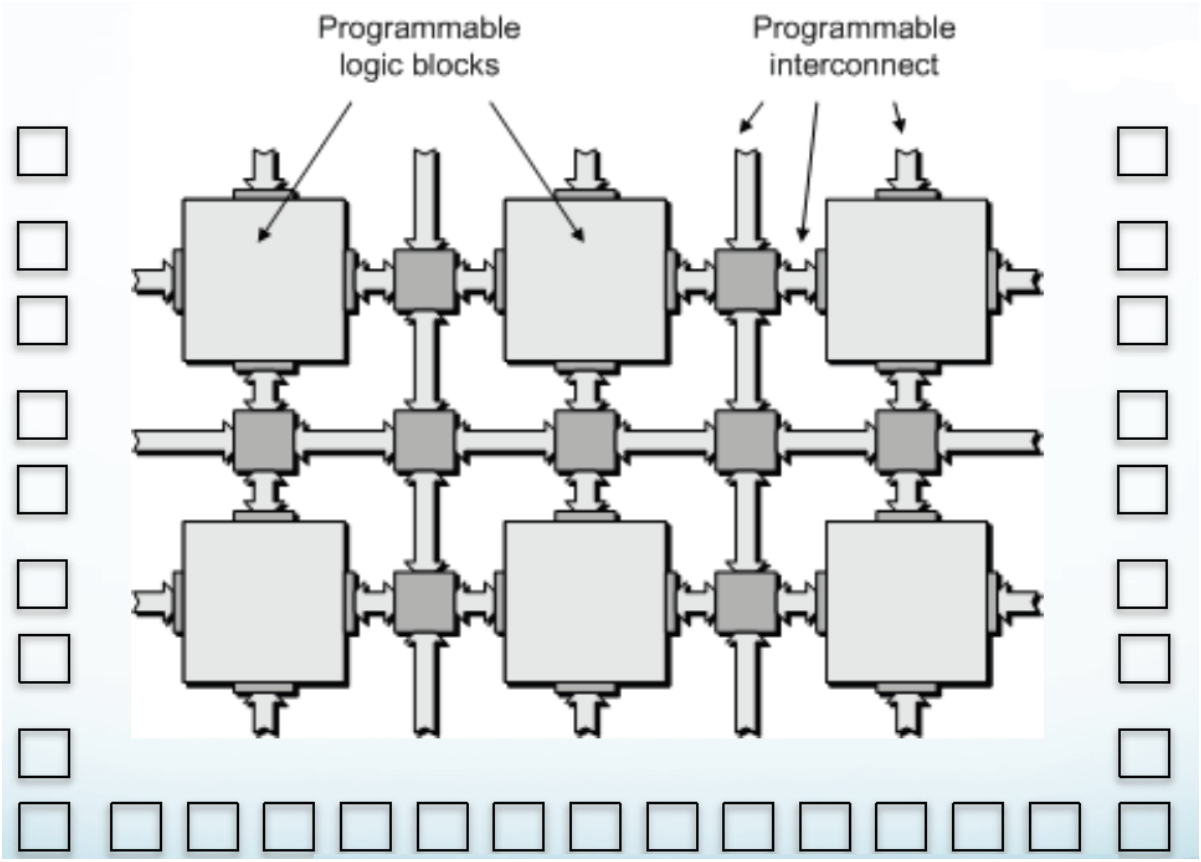
Definice požadavků na nový systém

- řízení a dohled nad systémem pro sběr dat
- řízení toku dat
- jednodušší systém
- zachování stávajícího formátu dat
- použití knihovny DIM
- použití některých modulů z DATE (Murphy TV, COOOL, log book, ...)
- využití specializovaného hardwaru (FPGA karty)
- řízení v reálném čase není vyžadováno

Nová hardwarová architektura pro sběr dat

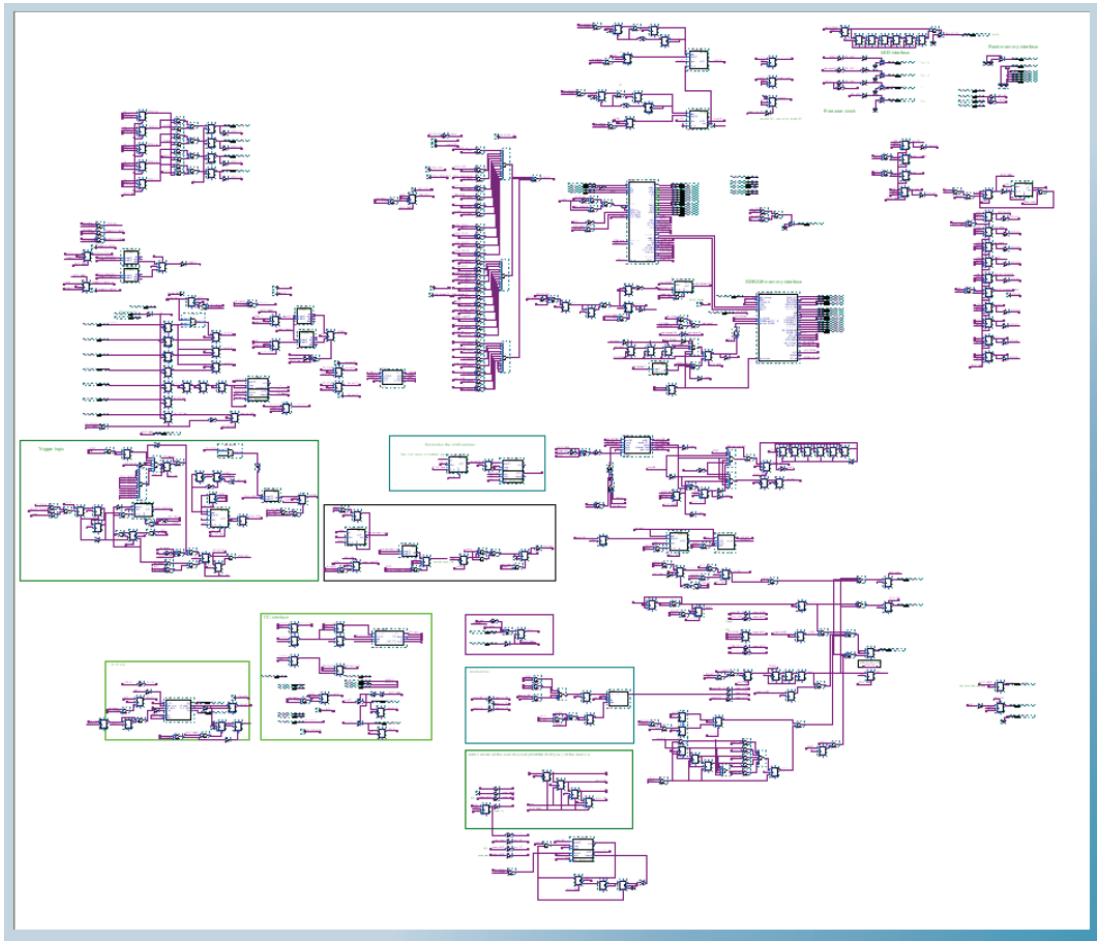


FPGA (Field-programmable gate array)



Čip programovatelný v poli (mimo továrnu), převzato z [3]

FPGA (Field-programmable gate array)



Zadání chování čipu pomocí schématu, převzato z [3]

FPGA (Field-programmable gate array)

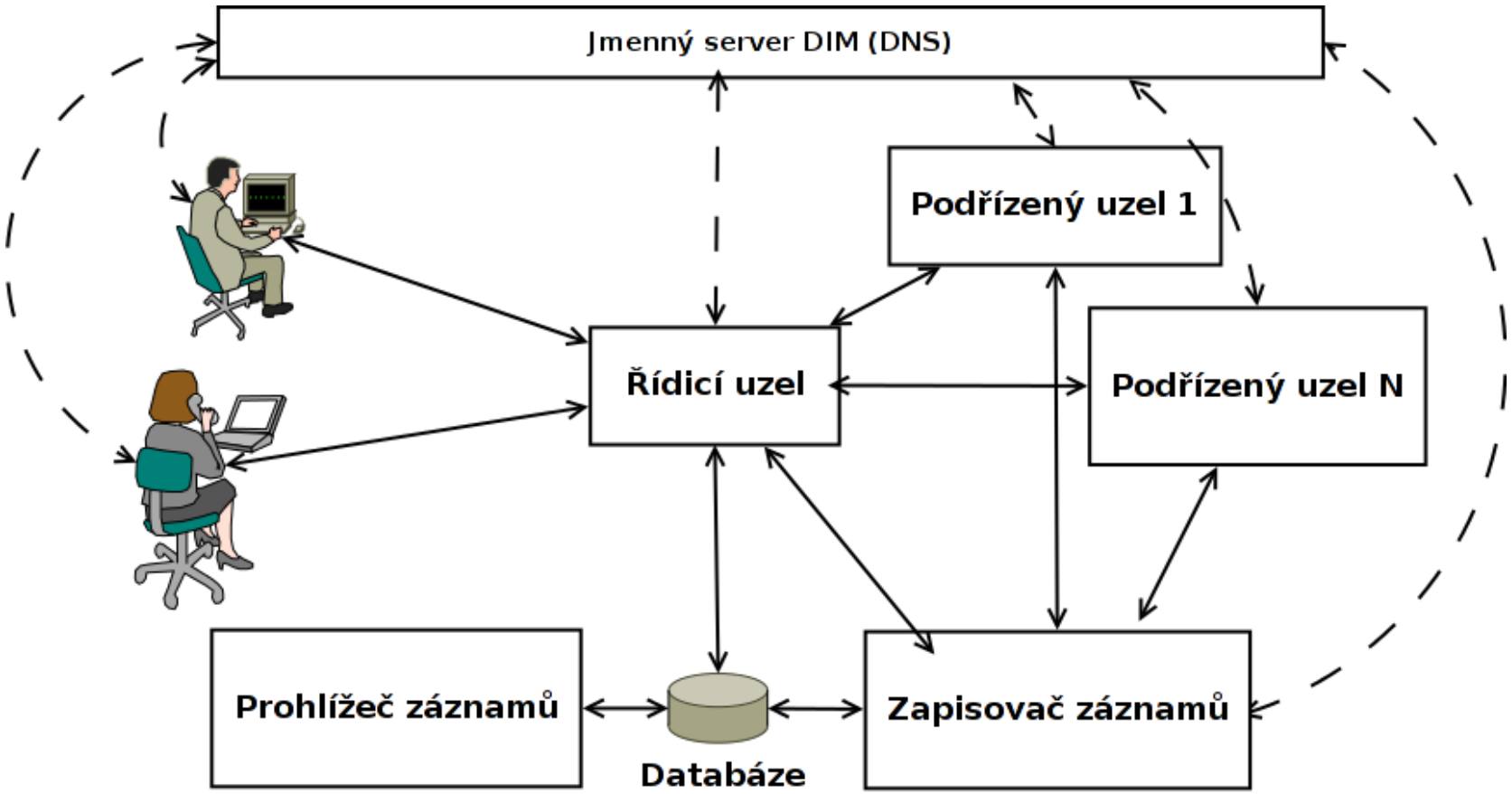
```
architecture behavioral of VMEReg is

  signal vme_en_i   : std_logic;
  signal Q : std_logic_vector(15 downto 0);

begin  -- behavioral

  vme_addr_decode : process (vme_addr, vme_en) is
    variable my_addr_vec : std_logic_vector(vme_addr'high downto 0);
    variable selected    : boolean;
  begin  -- process vme_addr_decode
    my_addr_vec := std_logic_vector( TO_UNSIGNED ( my_vme_base_address, vme_addr'high+1 ) );
    selected    := my_addr_vec(vme_addr'high downto 1) = vme_addr(vme_addr'high downto 1);
    vme_en_i <= '0' ;
    if selected then
      vme_en_i <= vme_en;
    end if;
  end process vme_addr_decode;

  reg: process (vme_clk, reset) is
  begin  -- process reg
    if reset = '1' then                 -- asynchronous reset
        Q <= init_val;
        vme_en_out <= '0';
    elsif vme_clk'event and vme_clk = '1' then  -- rising clock edge
      vme_en_out <= vme_en_i;
      if vme_en_i = '1' and vme_wr = '1' then
        Q <= vme_data;
      end if;
    end if;
  end process reg;

  data <= Q;
  vme_data_out <= Q;

end behavioral;
```

Zadání chování čipu pomocí VHDL, převzato z [3]

Struktura systému pro sběr dat

- Řídicí proces
  - ovládá podřízené procesy
  - komunikuje s databází
  - dostává příkazy od uživatelského rozhraní
- Podřízený proces
  - běží na specializovaném hardwaru (FPGA karta)
  - přijímá příkazy od řídicího procesu
  - poskytuje informace o stavu FPGA karty
- GUI
  - 1 řídicí rozhraní, *n* monitorovacích
  - přijímá informace od řídícího procesu
  - přes řídicí proces odesílá řídicí příkazy podřízeným procesům
- Message logger
- Message browser

Struktura systému pro sběr dat

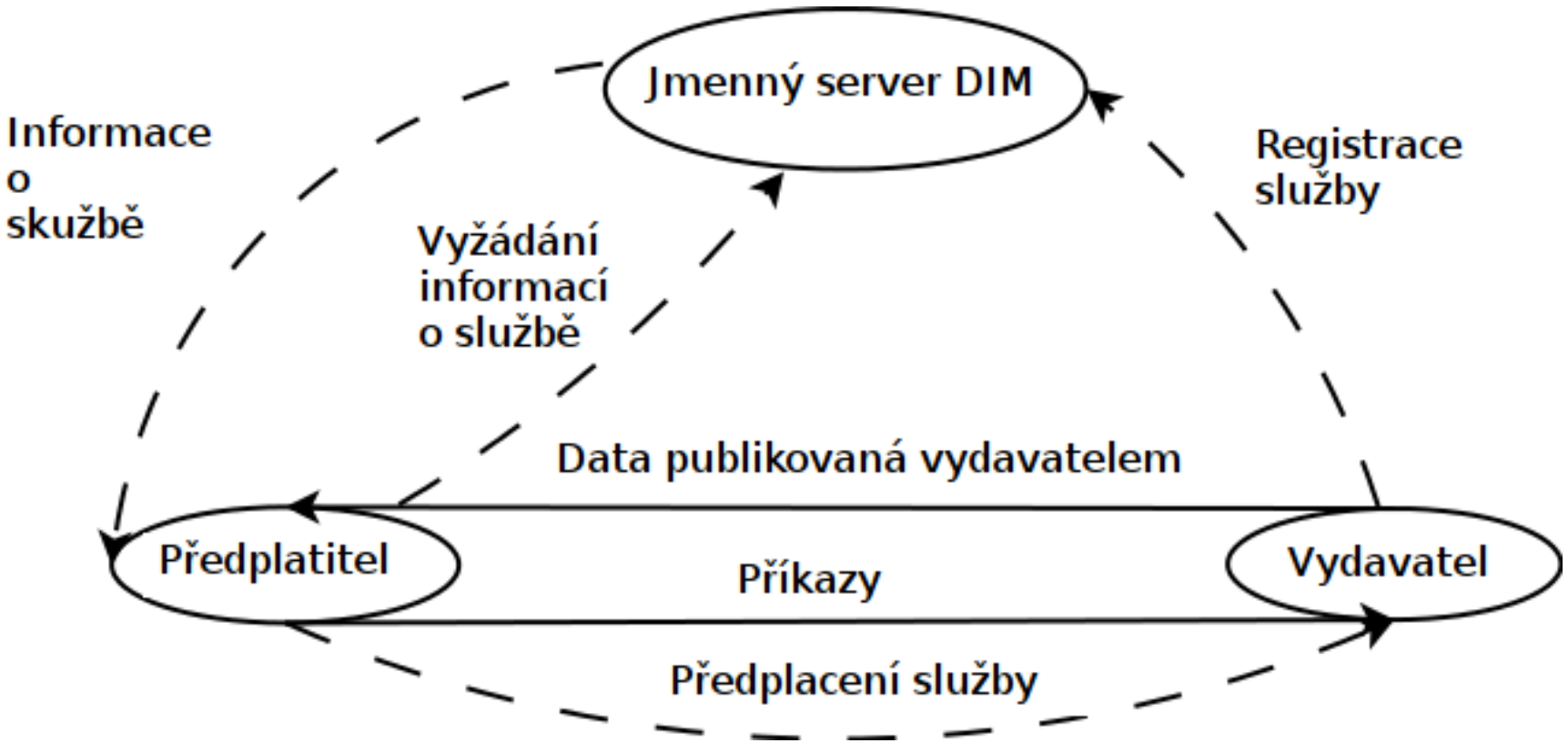


Role v systému

Knihovna DIM

- *Distributed Information Management*
- vývoj v CERN (původně pro experiment DELPHI)
- zajišťuje asynchronní 1 k *N* komunikaci po síti
- rozšíření paradigmatu klient–server o jmenný server
- postaveno na standartu TCP/IP
- rozhraní pro C/C++, Javu a Python
- multiplatformní knihovna
- používáno i v rámci DATE

Jmenný server DNS

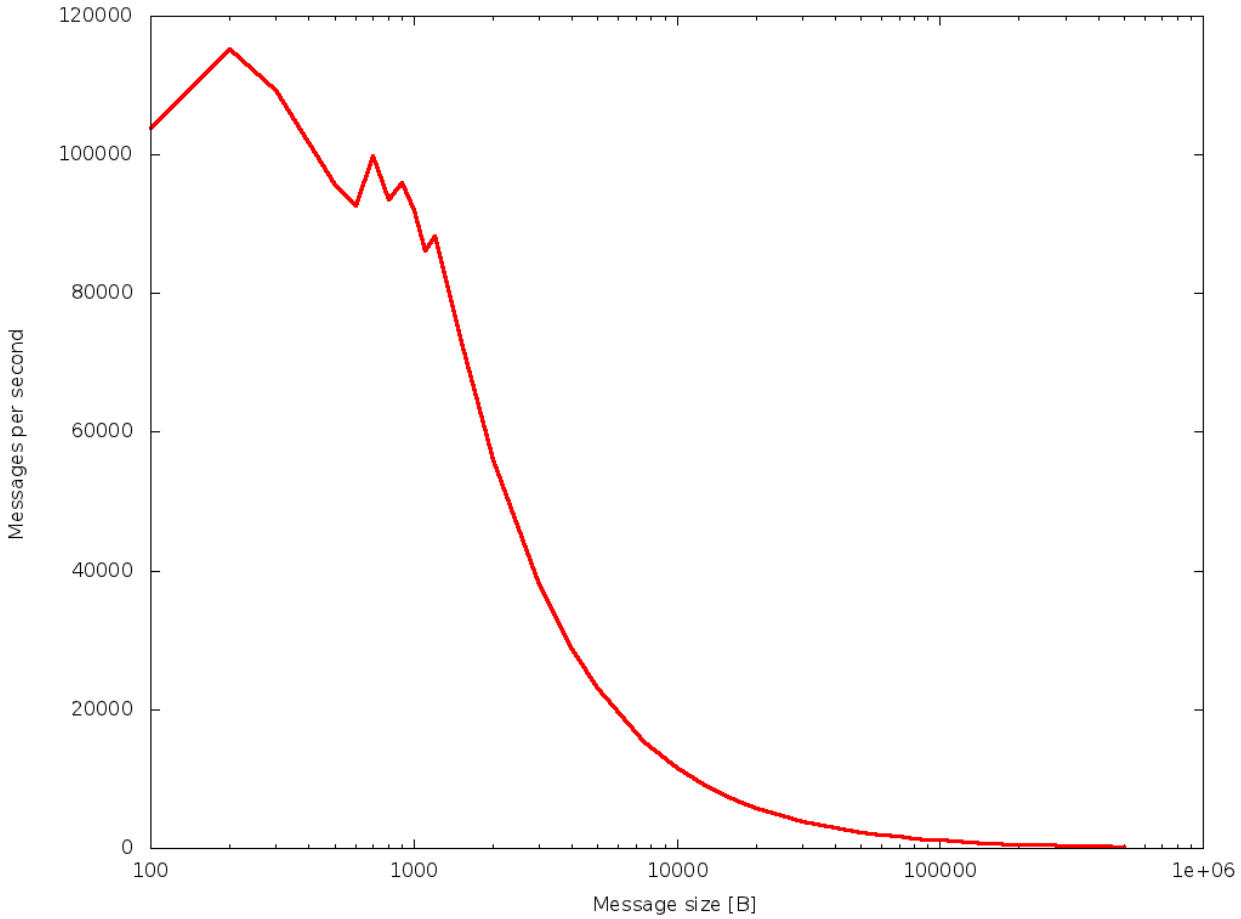


Komunikace prostřednictvím DIM knihovny

Charakteristika testovací verze

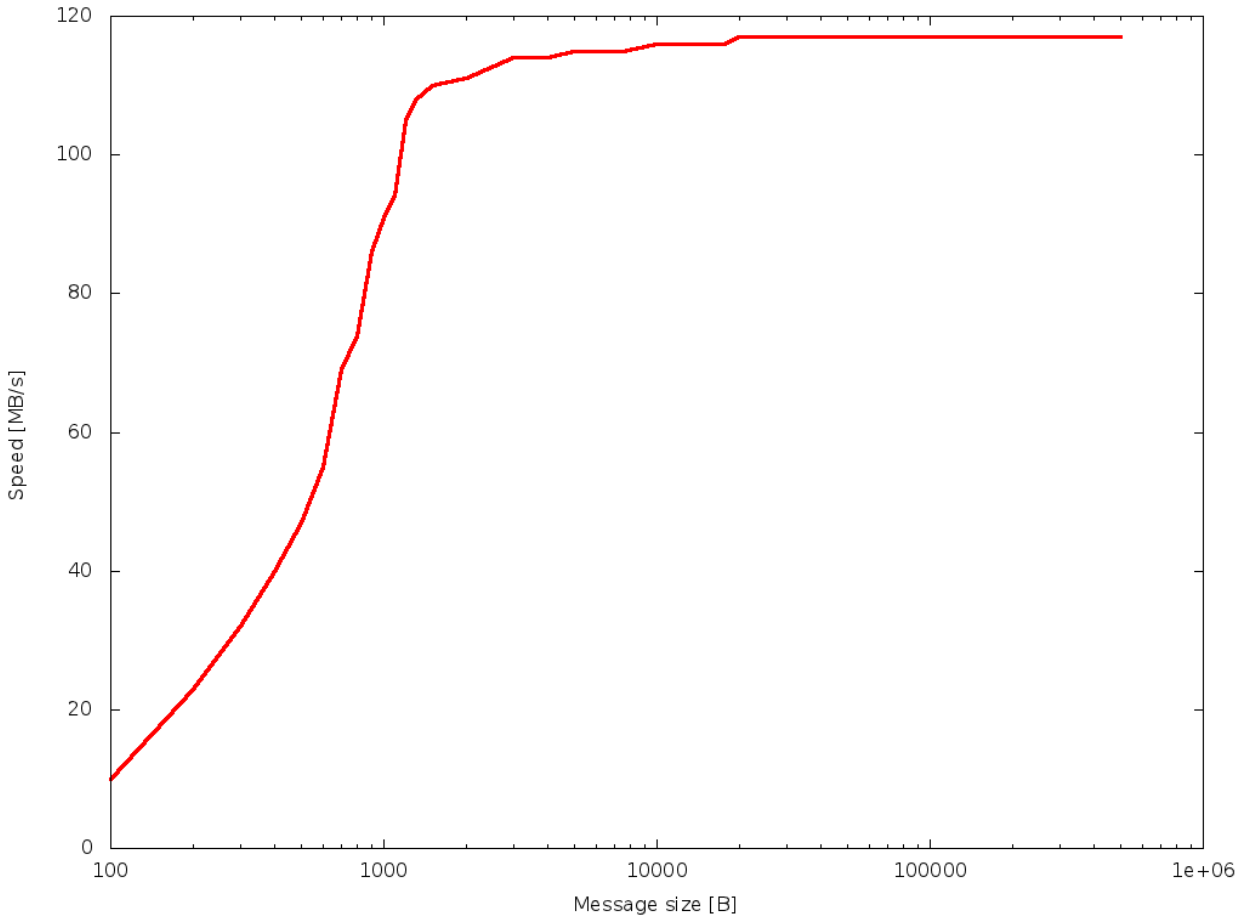
- hlavní část v QT frameworku
- nastavení a zprávy v MySQL databázi
- komunikace pomocí knihovny DIM (C++ rozhraní)
- pomocné skripty v Pythonu
- testování během zimní odstávky experimentu
- testy provedeny s parametry:
  - Gigabit Ethernet
  - 2-16 podřízených procesů na počítačích pro sestavování událostí
  - různá velikost zprávy od 100 B do 500 kB

Výsledky testů (1/3)



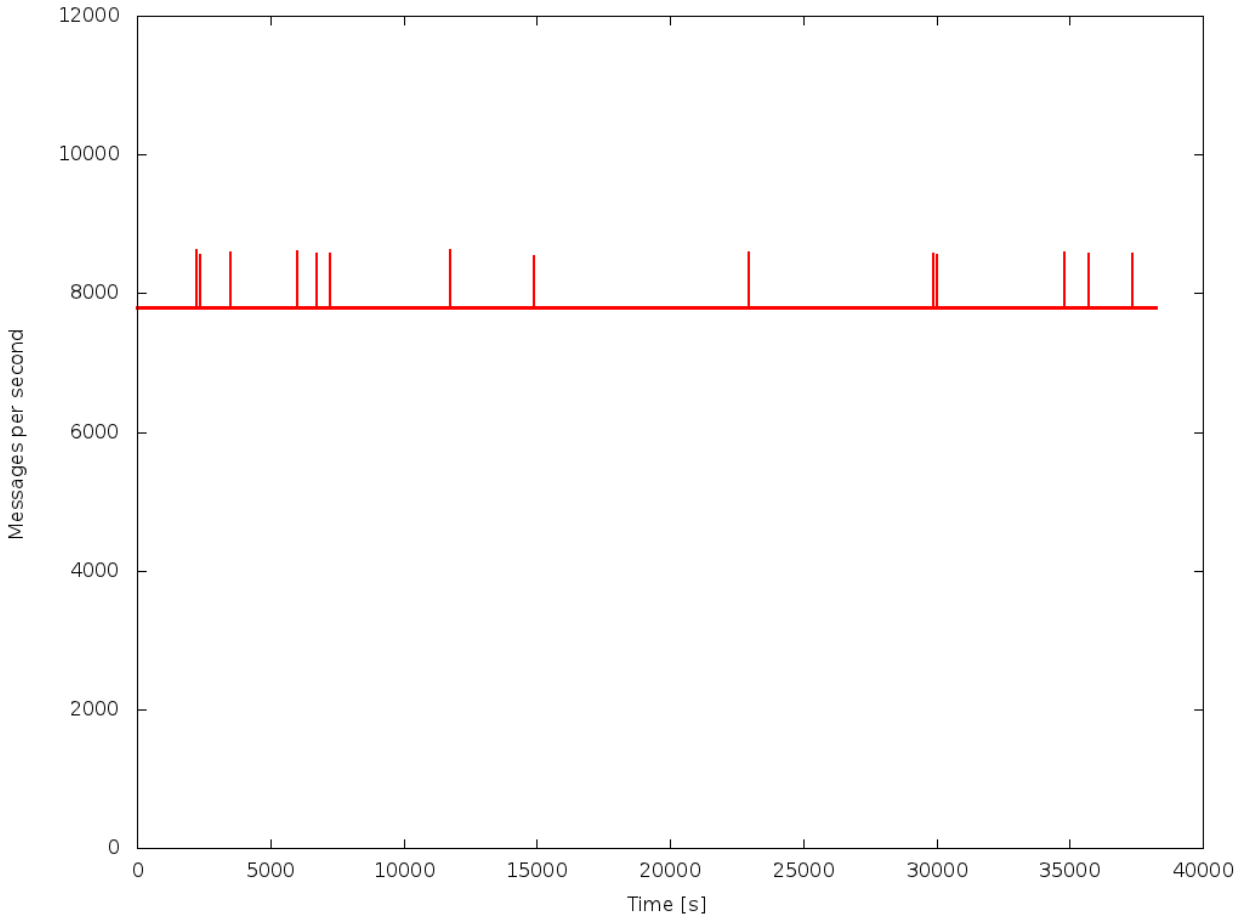
Počet zpráv za sekundu v závislosti na velikosti zprávy

Výsledky testů (2/3)



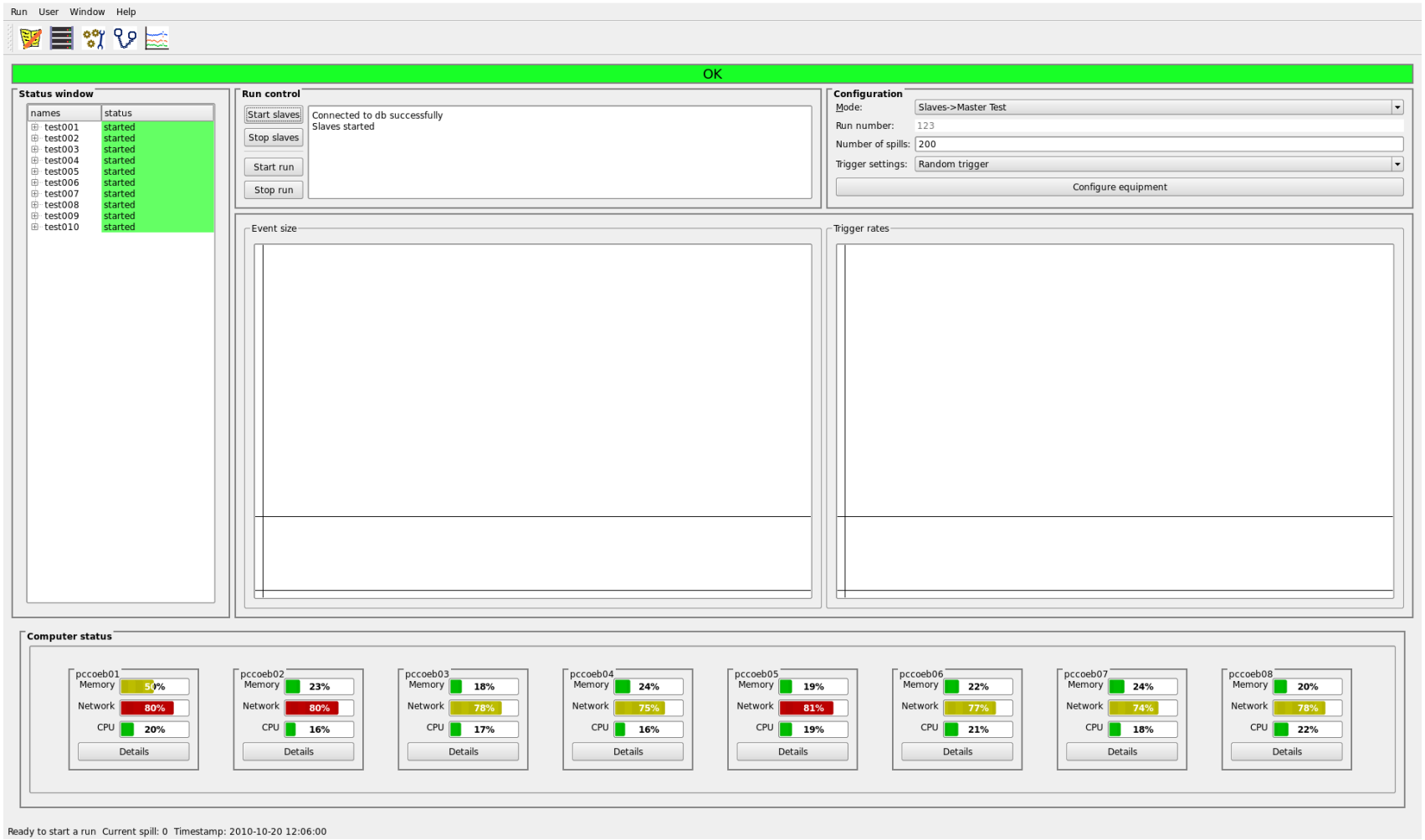
Rychlost přenosu v závislosti na velikosti zprávy

Výsledky testů (3/3)

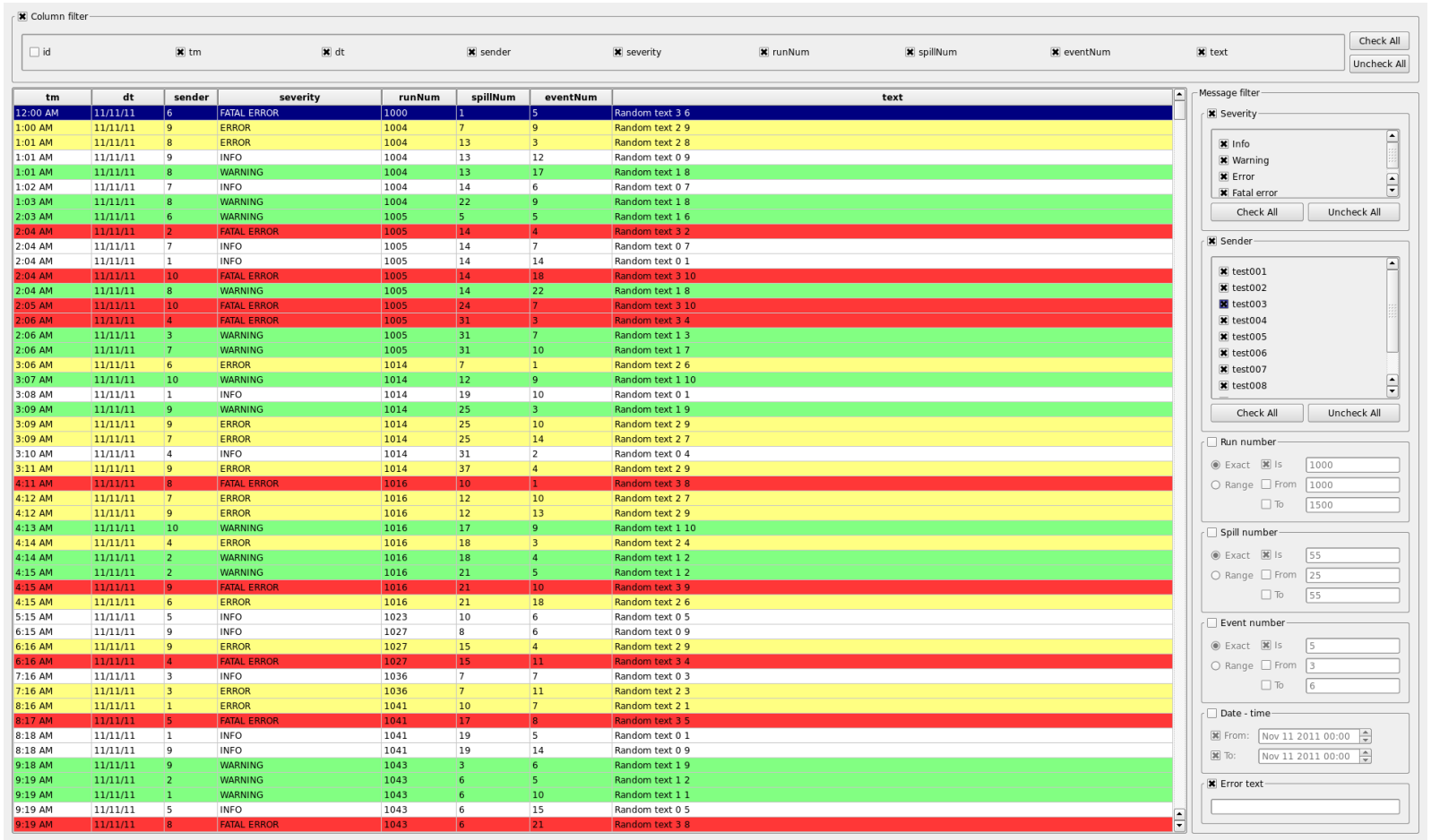


Test stability

Uživatelské rozhraní



Prohlížeč záznamů



tm	dt	sender	severity	runNum	spillNum	eventNum	text
12:00 AM	11/11/11	6	FATAL ERROR	1000	1	5	Random text 3 6
1:00 AM	11/11/11	9	ERROR	1004	7	9	Random text 2 9
1:01 AM	11/11/11	8	ERROR	1004	13	3	Random text 2 8
1:01 AM	11/11/11	9	INFO	1004	13	12	Random text 0 9
1:01 AM	11/11/11	8	WARNING	1004	13	17	Random text 1 8
1:02 AM	11/11/11	7	INFO	1004	14	6	Random text 0 7
1:03 AM	11/11/11	8	WARNING	1004	22	9	Random text 1 8
2:03 AM	11/11/11	6	WARNING	1005	5	5	Random text 1 6
2:04 AM	11/11/11	2	FATAL ERROR	1005	14	4	Random text 3 2
2:04 AM	11/11/11	7	INFO	1005	14	7	Random text 0 7
2:04 AM	11/11/11	1	INFO	1005	14	14	Random text 0 1
2:04 AM	11/11/11	10	FATAL ERROR	1005	14	18	Random text 3 10
2:04 AM	11/11/11	8	WARNING	1005	14	22	Random text 1 8
2:05 AM	11/11/11	10	FATAL ERROR	1005	24	7	Random text 3 10
2:06 AM	11/11/11	4	FATAL ERROR	1005	31	3	Random text 3 4
2:06 AM	11/11/11	3	WARNING	1005	31	7	Random text 1 3
2:06 AM	11/11/11	7	WARNING	1005	31	10	Random text 1 7
3:06 AM	11/11/11	6	ERROR	1014	7	1	Random text 2 6
3:07 AM	11/11/11	10	WARNING	1014	12	9	Random text 1 10
3:08 AM	11/11/11	1	INFO	1014	19	10	Random text 0 1
3:09 AM	11/11/11	9	WARNING	1014	25	3	Random text 1 9
3:09 AM	11/11/11	9	ERROR	1014	25	10	Random text 2 9
3:09 AM	11/11/11	7	ERROR	1014	25	14	Random text 2 7
3:10 AM	11/11/11	4	INFO	1014	31	2	Random text 0 4
3:11 AM	11/11/11	9	ERROR	1014	37	4	Random text 2 9
4:11 AM	11/11/11	8	FATAL ERROR	1016	10	1	Random text 3 8
4:12 AM	11/11/11	7	ERROR	1016	12	10	Random text 2 7
4:12 AM	11/11/11	9	ERROR	1016	12	13	Random text 2 9
4:13 AM	11/11/11	10	WARNING	1016	17	9	Random text 1 10
4:14 AM	11/11/11	4	ERROR	1016	18	3	Random text 2 4
4:14 AM	11/11/11	2	WARNING	1016	18	4	Random text 1 2
4:15 AM	11/11/11	2	WARNING	1016	21	5	Random text 1 2
4:15 AM	11/11/11	9	FATAL ERROR	1016	21	10	Random text 3 9
4:15 AM	11/11/11	6	ERROR	1016	21	18	Random text 2 6
5:15 AM	11/11/11	5	INFO	1023	10	6	Random text 0 5
6:15 AM	11/11/11	9	INFO	1027	8	6	Random text 0 9
6:16 AM	11/11/11	9	ERROR	1027	15	4	Random text 2 9
6:16 AM	11/11/11	4	FATAL ERROR	1027	15	11	Random text 3 4
7:16 AM	11/11/11	3	INFO	1036	7	7	Random text 0 3
7:16 AM	11/11/11	3	ERROR	1036	7	11	Random text 2 3
8:16 AM	11/11/11	1	ERROR	1041	10	7	Random text 2 1
8:17 AM	11/11/11	5	FATAL ERROR	1041	17	8	Random text 3 5
8:18 AM	11/11/11	1	INFO	1041	19	5	Random text 0 1
8:18 AM	11/11/11	9	INFO	1041	19	14	Random text 0 9
9:18 AM	11/11/11	9	WARNING	1043	3	6	Random text 1 9
9:19 AM	11/11/11	2	WARNING	1043	6	5	Random text 1 2
9:19 AM	11/11/11	1	WARNING	1043	6	10	Random text 1 1
9:19 AM	11/11/11	5	INFO	1043	6	15	Random text 0 5
9:19 AM	11/11/11	8	FATAL ERROR	1043	6	21	Random text 3 8

Dosažené cíle a další kroky

1. Analyzován současný systém pro sběr dat
   - založen na balíku DATE
   - problémy s výkonem a stabilitou
2. Nainstalován vzdálený velín experimentu
   - velín připraven k nasazení
   - ušetřeny finanční prostředky za přídavné stínění
3. Vývoj nového systému pro sběr dat
   - připraven návrh řídicího a dohledového systému
   - minimální verze tohoto systému implementována a otestována
4. Další kroky
   - testy na reálném HW (embedded linux na softcore procesoru)
   - rozšiřování funkcionality

Literatura


P. Abbon et al. (the COMPASS collaboration): *The COMPASS experiment at CERN*, In: Nucl. Instrum. Methods Phys. Res., A 577, 3 (2007) pp. 455–518


H. Sakulin: *Field Programmable Gate Arrays*, In: International School of Trigger and Data Acquisition, Krakow, February 2012


W. Vandeli: *Introduction to Data Acquisition*, In: International School of Trigger and Data Acquisition, Roma, February 2011


COMPASS page [online]. 2010. Available at: `http://wwwcompass.cern.ch`