XML processing using GPGPU
Research proposal

Jordan Vincent
University of Tsukuba
February 2, 2011
Academic achievements

- Engineering degree with emphasis on *Software development*.
- Research master degree with emphasis on *Parallel algorithms*.

from University of Technology of Belfort-Montbeliard (France).

Internship

Final project assignment (6 months) at Kitagawa Data Engineering laboratory (University of Tsukuba, Japan).
1 Research project
   - Background
   - Master project
   - Next challenge

2 Research plan
   - Schedule
   - Scope of research
Outline

1 Research project
   - Background
   - Master project
   - Next challenge

2 Research plan
   - Schedule
   - Scope of research
XML

Semi-structured data format for exchanging data in a textual form.

```xml
<a>
  <b arg1="value1">
    <c /></b>
  <b arg1="value2">
    <c>text</c>
  </b>
</a>
```

Worldmap:
171 GB
(Sept 2010)

English articles:
27 GB
(Sept 2010)

XPath

Core retrieval language for XML doc. XPath is a subset of XQuery.

```
/a/b[@arg1=value2]/c
```
TwigStack

TwigStack[1] is a famous algorithm to perform XML pattern matching.
Manycore processor family

Heterogenous parallel processor architectures
- Nvidia (GPU)
- ATI/AMD (GPU/CPU)
- Intel (Larrabee project)

CUDA
Nvidia specific toolkit for general purpose development on GPU.

OpenCL
"The open standard for parallel programming of heterogeneous systems" includes some GPU, CPU but also some DSP chips.
- Sony/IBM/Toshiba Cell
- Apple iPhone
Research works about GPGPU and DB processing

- Fast computation of database operations using graphics processors [Govindaraju, SIGMOD’04]
- GPUQP: Query Co-Processing Using Graphics Processors [Fang, SIGMOD’07]
- Relational Joins on Graphics Processors [He, SIGMOD’08]
- Data Monster: Why graphics processors will transform database processing? [Di Blas, 2009]
- Accelerating SQL Database Operations on a GPU with CUDA [Bakkum, GPGPU’10]
- Exploring utilisation of GPU for database applications [Walkowiak, ICCS’10]
- Accelerating XML Query Matching through Custom Stack Generation on FPGAs [Moussalli, HiPEAC’10]

→ No research result about XML processing using GPGPU.
Master project: TwigStackGPU

Outline

Based on Imam Machdi’s research[2] at KDE lab. about *TwigStack algorithm* for parallel query processing on cluster and multicore processors.

Current result

Application possible to Nvidia GPGPU?
6 months work and many technical problems encountered.
→ Project works but slow execution time.
XML document

root

leaves

partition 1 partition 2 partition 3

Partitionning algorithm

XML pattern matching

Intermediate solutions merge

query matches

XML processing using GPGPU
Next challenge

Immediate future tasks
- performance evaluation and profiling.
- solve implementation issues for better performance.

Many more problems to be addressed
- evaluate other architectures than Nvidia.
- enhance pattern matching algorithm to make use of more capabilities of GPU.
- explore other problems that share the same representation of XML documents.
1 Research project
   - Background
   - Master project
   - Next challenge

2 Research plan
   - Schedule
   - Scope of research
Estimated schedule

A
- past
  - first demo in CUDA (nVIDIA)
  - TwigStack implementation on GPGPU
- 2011
  - non-uniform parallelism exploration
  - OpenCL framework for XML processing
- B
  - design a new query processing algorithm that better fits GPGPU
  - algorithm for XML query processing on GPGPU
- C
  - address other fields of XML processing domain
  - XML-OLAP implementation on GPGPU
- D
  - 2014

Master thesis
- TwigStackGPU presentation and benchmark
- Manycore platform comparison of TwigStack algorithm using new OpenCL framework
- Efficient XML query processing on GPGPU
- XML-OLAP on GPU presentation and benchmark
- PhD thesis
XML processing using GPGPU
1. XML query processing is a problem due to the growing amount of content stored into XML documents.

2. Current project shows that XML query processing on GPGPU is possible but well-known algorithm is not efficient.

3. No research results about XML query processing and GPGPU yet, but promising results about relational database query processing.

4. An efficient GPU framework could be the base of other researches related to XML processing. (e.g., XML-OLAP operation[3] using GPGPU)
Nicolas Bruno, Nick Koudas, Divesh Srivastava.  
*Holistic Twig Joins: Optimal XML Pattern Matching.*  
SIGMOD 2002

Imam Machdi, Toshiyuki Amagasa, Hiroyuki Kitagawa.  
*Executing parallel TwigStack algorithm on a multi-core system.*  

Chantola Kit, Toshiyuki Amagasa, Hiroyuki Kitagawa.  
*Algorithms for Efficient Structure-based Grouping in XML-OLAP.*  
iiWAS, 2008.
## backup slide: CPU vs GPU

### thread scheduling

<table>
<thead>
<tr>
<th></th>
<th><strong>GPU</strong></th>
<th><strong>CPU</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>hardware, massive parallelism of non-divergent threads</td>
<td>software, few parallelism of divergent threads</td>
</tr>
</tbody>
</table>

### memory consistency

<table>
<thead>
<tr>
<th></th>
<th><strong>GPU</strong></th>
<th><strong>CPU</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>no hardware consistency, software consistency not recommended (little independent caches, many cores)</td>
<td>hardware and complexe memory consistency management (big unified caches, few cores)</td>
</tr>
</tbody>
</table>

### computing priority

<table>
<thead>
<tr>
<th></th>
<th><strong>GPU</strong></th>
<th><strong>CPU</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Less global memory</td>
<td>More global memory</td>
</tr>
</tbody>
</table>
backup slide: GPU powered webserver

Dynamic XML doc. request
(auth., AES decrypt, ...)

XPath queries on XML doc.

XML document answer
(AES encrypt, ...)

webserver

fastCGI

GPU

CPU

Jordan Vincent

XML processing using GPGPU