Type-Safe Object Exchange between Applications and a DSM-Kernel

- Major goals of Plurix
- Type-safety in OS development
- The Plurix Project
- DHS and transactional consistency

- Where to store kernel and applications?
  - Inter address space pointers
  - A kernel running in the DHS

- Measurements

University of Ulm
Germany

Speaker: Ralph Goeckelmann
Major Goals of the Plurix Project

- Distributed operating system for PC cluster
- Simplified development of distributed applications
  - implicit synchronisation
  - automatic garbage collection
- Using DSM for communication
- Lean system design
  - stand alone OS
- Object orientation
Traditional Operating Systems

- Most distributed OS implemented as Middleware
- Host OS mostly written in C / C++
- OS is not type-safe
  - no protection of code segments
  - no type-safe parameter passing
- explicit protection of the kernel is needed
  - using different address spaces
  - invoking kernel methods by using interrupts
- Challenging parameter passing to the kernel
  - no direct parameter passing possible
- Systems mostly very complex
  - high risk of faults and attacks (viruses / worms)
Type-Safe Operating System

- Using a type-safe language for OS development
- Implicit protection of the kernel
  - Modification of pointers not possible
  - Only published methods can be called
  - Code segments automatically protected
- No different address spaces needed
- No explicit parameter check
- Simplified programming
- Increased security and stability
The Plurix Project

- Type-safe language: Java
- Special Java-Compiler
- Only objects in the DSM
  - DSM organized as type-safe heap
- Transactional consistency
- Persistence and fault tolerance
- Unified name service
Distributed Heap Storage

- Fragmentation and False-Sharing must be handled
  - Relocation of objects necessary
- Relocation of objects in the DHS
  - reallocate object
  - adjust all references
- Bookkeeping of references
  - Backlinks and Backpacks
- Garbage Collection using Backlinks
  - cluster wide and incremental GC
  - objects with empty Backlinks → garbage
Transaction Consistency

- Restartable transactions
  - follow the well known ACID paradigm
  - implicitly defined by the central loop
  - aborted in case of a collision
- Optimistic synchronization
  - writes initially on local copies only, shadow copies are preserved
  - separate Read- and Write-Sets are built during a transaction
  - an ending transaction broadcasts a Commit-Request,
  - reset on collision
- strict consistency at commit points
- sequential consistency always preserved
Where to store the applications?

- Transparency requirements for a distributed OS
- Possible solutions: local memory or DHS
- Applications generated or modified during runtime
- Applications in local memory
  - separate installation on each node
  - explicit transferred to each node
  - new applications must be simultaneously installed
- Applications in the DHS
  - Compiler is running in the DHS
  - new applications are automatically compiled into the DHS
  - Code Segments are automatically shared and distributed

Applications should reside in the DHS
Where to store the kernel?

- Possible solutions: local memory or DHS
  - Kernel in local memory
    - obvious approach
    - same requirements as for applications
    - changes must be simultaneously broadcasted to each node
  - Kernel in the DHS
    - same benefits as for applications
    - but special attention needed
    - protection of kernel objects against invalidation necessary
Inter Address Space Pointers

- Direct pointer from local memory into the DHS or Vice versa
- Indirect pointers / translation tables decreases performance
- Problem 1: Typedescriptor
  - typedescriptor resides in the DHS only => Kernel can not start
  - typedescriptor in local memory only => must reside at the same address
  - two sets of typedescriptors => no Java like type-check possible
Inter Address Space Pointers 2

- **Problem 2: Relocation of DHS-Objects**
  - addresses in local memory not unique
  - relocation of DHS-object requires pointer adjustment
  - difficult adjustment: each node must adjust at the same time
  - automatic adjustment by using the Backlinks not possible
  - relocation is prohibited

**Avoiding different address spaces**
A kernel running in the DHS

- All objects reside in the DHS
- Avoiding inter address space pointers
  - normal type-check possible
  - relocation of objects not prohibited
- Elegant and type-safe parameter passing
- Current kernel version available by the DHS
  - all nodes running the same kernel version
- Ease checkpointing and transaction migration
- But: protection of system-objects needed
Measurements

- Using a distributed raytracer for performance evaluation

- Used scene:
  - 103 spheres
  - 3 spot lights
  - 8 triangles

- Measurements

<table>
<thead>
<tr>
<th>pixels</th>
<th>lfd</th>
<th>node 2</th>
<th>max 4</th>
<th>max 8</th>
<th>max 12</th>
<th>min 2</th>
<th>min 4</th>
<th>min 8</th>
<th>min 12</th>
</tr>
</thead>
<tbody>
<tr>
<td>307200</td>
<td>1</td>
<td>30969</td>
<td>16844</td>
<td>9890</td>
<td>15941</td>
<td>7551</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>602112</td>
<td>2</td>
<td>60265</td>
<td>32344</td>
<td>18062</td>
<td>10878</td>
<td>30762</td>
<td>14789</td>
<td>6926</td>
<td></td>
</tr>
<tr>
<td>3145728</td>
<td>3</td>
<td>306599</td>
<td>157073</td>
<td>81743</td>
<td>45709</td>
<td>32955</td>
<td>155151</td>
<td>80772</td>
<td>26719</td>
</tr>
<tr>
<td>12582912</td>
<td>4</td>
<td>1221654</td>
<td>623468</td>
<td>322903</td>
<td>166561</td>
<td>117484</td>
<td>620206</td>
<td>313608</td>
<td>107382</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>lfd</th>
<th>2 spd</th>
<th>3 spd</th>
<th>4 spd</th>
<th>6 spd</th>
<th>8 spd</th>
<th>12 spd</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.84</td>
<td>2.49</td>
<td>3.13</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.86</td>
<td>2.62</td>
<td>3.34</td>
<td>4.6</td>
<td>5.54</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1.95</td>
<td>3.75</td>
<td>5.36</td>
<td>6.71</td>
<td>9.3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1.96</td>
<td>3.78</td>
<td>7.33</td>
<td>10.4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The End

- http://www.plurix.de