OpenTTD@large –OR–
Massivizing Online Games –OR–
Distributed Computing Challenges and
High Quality Time

Alexandru Iosup
Parallel and Distributed Systems Group
Delft University of Technology

Our team: **Undergrad** Adrian Lascateu, Alexandru Dimitriu (UPB, Romania), ..., **Grad** Siqi Shen, Yong Guo, Ruud vd Bovenkamp (TU Delft, the Netherlands), ... **Staff** Otto Visser, Dick Epema, Henk Sips, Fernando Kuipers (TU Delft), Thomas Fahringer, Radu Prodan (U. Innsbruck), Nicolae Tapus, Vlad Posea, Alexandru Olteanu (UPB), ...

Massivizing Social Games: Distributed Computing Challenges and High Quality Time – A. Iosup

Haifux, Technion, Haifa, IL. June 2013
Lectures at the Technion Computer Engineering Center (TCE), Haifa, IL

IaaS Cloud Benchmarking  May 7  10am Taub 337
Massivizing Online Social Games  May 9
Gamification in Higher Education  May 27

OpenTTD@large –or– Massivizing Online Social Games  June 3 (Haifux)

Scheduling in IaaS Clouds  June 5 (HUJI)

A TU Delft perspective on Big Data Processing and Preservation  June 6

Grateful to Orna Agmon Ben-Yehuda, Assaf Schuster, Isaac Keslassy.

June 3, 2013

Also thankful to Bella Rotman and Ruth Boneh.
(TU) Delft – the Netherlands – Europe

founded 13th century
pop: 100,000

founded 1842
pop: 13,000

pop: 16.5 M

(We are here)
The Parallel and Distributed Systems Group at TU Delft

Alexandru Iosup
Grids/Clouds
P2P systems
Big Data
Online gaming

Dick Epema
Grids/Clouds
P2P systems
Video-on-demand
e-Science

Ana Lucia Varbanescu
HPC systems
Multi-cores
Big Data
e-Science

Henk Sips
HPC systems
Multi-cores
P2P systems

Johan Pouwelse
P2P systems
File-sharing
Video-on-demand

Home page
- www.pds.ewi.tudelft.nl

Publications
- see PDS publication database at publications.st.ewi.tudelft.nl

August 31, 2011
Why Social Gaming?
Massivizing Social Gaming = Rich Challenge (of Distributed Computing)

Online Gaming used to be art, may now be computing

Online Gaming used to be multimedia, is now DC

Online Gaming used to be networking, is now all DC

Online Gaming used to be v-worlds, is now many apps

DC = Distributed Computing
What’s in a name? MSG, MMOG, MMO, …

Over 250,000,000 active players

Massively Social Gaming = (online) games with massive numbers of players (100K+), for which social interaction helps the gaming experience

1. Virtual World Sim
   Explore, do, learn, socialize, compete

2. Game Data
   Player stats and relationships, others

3. Game Content
   Graphics, maps, puzzles, quests, culture
Zynga, an Amazon WS User

Selling in-game virtual goods:

“Zynga made est. $270M in 2009 from.”
http://techcrunch.com/2010/05/03/zynga-revenue/

“Zynga made more than $600M in 2010 from selling in-game virtual goods.”
S. Greengard, CACM, Apr 2011
MSGs are a Popular, Growing Market

- 25,000,000+ subscribed players (from 250,000,000+ active)
- Over 10,000 MSGs in operation
- Subscription market size $7.5B+/year, Zynga $600M+/year

World of Warcraft, a Traditional HPC User
(since 2003)

- 10 data centers
- 13,250 server blades, 75,000+ cores
- 1.3PB storage
- 68 sysadmins (1/1,000 cores)

Bungie, Computing then Serving 1.4PB/yr.

- Halo 3 is one of the many successful games
- Halo 3 players get, in 1.4PB
  - Detailed player profiles
  - Detailed usage stats
  - Ranking

- CERN produces ~15PB/year (10x larger)
  - (Not) faster than the speed of light, the Higgs boson (?)
Agenda

1. What’s in a Name?

2. **Our Use-Case: OpenTTD@large**

3. Three Current Challenges
   1. Platform Scalability Challenge
   2. Gaming Analytics Challenge
   3. Content Generation Challenge

4. The Next Five Years

5. Conclusion
# Bringing a Classic to the 21st Century

<table>
<thead>
<tr>
<th>Chris Sawyer’s Transport Tycoon</th>
<th>TTD@Win95</th>
<th>1996</th>
<th>1997</th>
<th>2003</th>
<th>2007</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OpenTTD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OpenTTD@large</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Transport Tycoon Deluxe**: climate, better signals
- Jeff Drexler’s TTDPatch++,
gfx++
- OpenTTD+AIs

---

*Massivizing Social Games: Distributed Computing Challenges and High Quality Time – A. Iosup*

*TU Delft: Delft University of Technology*
**OpenTTD: Open-Source Life to Transport Tycoon Deluxe ~300k players**

<table>
<thead>
<tr>
<th>Replaced</th>
<th>Tech limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>• GFX, SFX, Music</td>
<td>• Max. 15 players (255 if cooperating, rare)</td>
</tr>
<tr>
<td>• Non-cheating AI</td>
<td>• Max. map size $2k^2$</td>
</tr>
<tr>
<td>• AI VM + API (Squirrel~Lua)</td>
<td>• Scalable tech?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Added or improved</th>
<th>Design limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DLC: mods/maps/AIs</td>
<td>• Limited variety</td>
</tr>
<tr>
<td>• Pathfinding, train signal system, vehicle handling</td>
<td>• No social</td>
</tr>
<tr>
<td>• Multiplayer</td>
<td>• Scalable design?</td>
</tr>
<tr>
<td>• Too many to mention</td>
<td></td>
</tr>
</tbody>
</table>

---

Massivizing Social Games: Distributed Computing Challenges and High Quality Time – A. Iosup
OpenTTD: Some Tech Limitations

- Network instability
- CPU overload
- Memory instability
- So ... 15 players in one game

OpenTTD: Some Design Limitations

- # profitable vehicles
- Complex to configure (e.g., AI selection)
- Free-riding AIs so far unbeatable (dominant strategy)
  - Our leading AI Rondje om de Kerk does this
  - Our OtviAI does not, is human level

OpenTTD@large: Massivizing OpenTTD

• **Tech**
  - Automatic scaling of server capacity
  - Single-map scalability enhancements
  - Gaming analytics engine

• **Design**
  - Unlimited map size
  - Unlimited amount of players
  - Support both casual and hardcore gamers
  - Add social aspects (like guilds and achievements)

• **Co-scalability of game platform and design!**
OpenTTD@large: Game Modes (for unlimited map size, # players)

- Quick game
  - Think of a 15 minute lunch break game

- Normal game
  - A few hours; much like current OpenTTD

- Challenge mode
  - Accomplish a certain feat, to unlock technology

- Unlimited
  - Unlimited size, unlimited players, only unlocked technology and your own little square on the map
OpenTTD@large: One Social Aspect
The Neighbor Interaction [1/2]

• A new way to interact

• Scenario: A map can have wood, but no sawmills. Need exchange mechanism to keep economy running.

• Mechanism elements:
  • Players can build “trade centers” at the map edges
  • Players can suggest “international” trades (e.g.: oil at 120$) to a specific neighbouring map or to all
  • The neighbouring map player(s) accept (or not)
  • Price and volume are negotiable
  • Play with currency exchange rate if needed
OpenTTD@large: One Social Aspect
The Neighbor Interaction [2/2]

- Players can build “trade centers” at the map edges
OpenTTD@large: One Persistent-Design Aspect – In-Game Time Progress

• Unlimited game → unlimited years?

• Problem: Technological advancement was based on the simulated date → unlimited years would mean unlimited tech (but we have no designers for this)

• A solution: achievements to unlock technology
  • Done 100,000 km on steam? → ready for diesel!
  • Grow a city beyond 10K inhabitants? → bigger stations!
@large: Status Update

- Working:
  - @Large platform with AAA; keeps track of servers, players and their progress
  - OpenTTD@Large With 4 game modes (3 out of 4 working) and cross server trading
  - Achievement system
  - Continuous integration testing with build and test farm

- Work in progress:
  - Automated scaling of server capacity with DAS-4 or EC2
  - Finding and/or making interesting maps for challenge mode
  - Add appropriate achievements and tech unlocks

- Future work:
  - Unlimited map with AOS
OpenTTD@large: We’re Looking for Great Open-Source Developers

• Contact Otto Visser or Alexandru Iosup

• Types of projects
  • Coding, mostly C/C++, Python, * for glue; esp. Social part
  • Game design
  • GFX, SFX, Music
  • Win and Mac ports, other games added to @large

• Other help: advice, comments, help w dissemination
Agenda

1. What’s in a Name?
2. OpenTTD@large

3. Three Current Challenges
   1. Platform Scalability Challenge
   2. Gaming Analytics Challenge
   3. Content Generation Challenge

4. The Next Five Years
5. Conclusion
Cloudification: PaaS for MSGs

(Platform Challenge)
Build MSG platform that uses (mostly) cloud resources

• Close to players
• No upfront costs, no maintenance
• Compute platforms: multi-cores, GPUs, clusters, all-in-one!
• Performance guarantees
• Hybrid deployment model
• Code for various compute platforms—platform profiling
• Load prediction miscalculation costs real money
• What are the services?
• Vendor lock-in?
• My data
Mobile Social Gaming and the SuperServer

(Platform Challenge)
Support MSGs on mobile devices

- Mobiles everywhere (2bn+ users)
- Gaming industry for mobiles is a new Growing Market
- SuperServer to generate content for low-capability devices?
- Battery for 3D/Networked games?
- Where is my server? (Ad-hoc mobile gaming networks?)
- Security, cheat-prevention
Social Everything! So Analytics

- **Social Network** = undirected graph, **relationship** = edge
- **Community** = sub-graph, density of edges between its nodes higher than density of edges outside sub-graph

**(Analytics Challenge)**

**Improve gaming experience**
- Ranking / Rating
- Matchmaking / Recommendations
- Play Style/Tutoring

**Self-Organizing Gaming Communities**
- Player Behavior
(Content Challenge)

Produce and distribute content for 1BN people

- Game Analytics → Game statistic
- Crowdsourcing
- Storification
- Auto-generated game content
- Adaptive game content
- Content distribution/
  Streaming content
Agenda

1. What’s in a Name?
2. OpenTTD@large

3. Three Current Challenges
   1. Platform Scalability Challenge
   2. Gaming Analytics Challenge
   3. Content Generation Challenge

4. The Next Five Years
5. Conclusion
Impact of Platform on Game-Play

Responsive game-play

Unresponsive game-play
@large Research Challenge: V-World Platform for MMOGs

Scaling quickly to millions of players
- 1M in 4 days, 10M in 2 months
- Up-front and operational costs
- Performance, Scalability, & Cost
Rich Research Opportunities: How to Build the Core?

- Application models
- Deployment/Hosting models
- Scheduling
- Performance engineering
- Reliability engineering
- Scalability and Elasticity
- Offloading
- Etc. (including Usability, Security, Utility Models, and Programming Models)
Platform Scalability Challenge

1. **Introduction**
   1. Online Game Types
   2. Game Models

2. **(Scheduling) Hosting Models**
   1. Cluster + Parallelization
   2. Multi-cluster + Sharding
   3. Cloud-based + Sharding
   4. The SuperServer (video)

3. **(Naming) Locating Servers**
   1. Central directory
   2. Peer-to-Peer

4. **Consistency**
   1. (Eventual) Dead-reckoning
   2. (Continuous) Simultaneous Sim
   3. (Continuous) + Pipelining
   4. (Continuous) + Dynamic Tick
   5. (Continuous/Eventual) Area of Simulation vs Interest

5. Other issues

> Gaming Analytics Challenge
> Content Generation Challenge
Online Game Types

• MMO Role Playing Games (MMORPG)
  • Adventure: Runescape, World of Warcraft
  • Thousands of players sharing one persistent game session in a huge game world
  • Latency <1s acceptable

• First Person Shooter (FPS)
  • Action games
  • Counter Strike Source, Call of Duty, ...
  • Max. 64 players in one session (minutes or hours)
  • Latency <100ms needed

• Real-Time Strategy (RTS)
  • Economic and battle strategy games
  • DotA 2, Starcraft
  • Latency <350ms needed
Computational Model for the Server

• Single sequential loop
• 3 steps in each loop:
  1. Game-world state update
  2. Entity interaction computation (dominant for MSGs)
  3. Entity state updates
• Load generated by (2) non-deterministic \(\leftrightarrow\) human factor

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Model for Entity Interaction Computation

- Player to Player/Player to Environment interaction
  - **Low** interaction: $O(n)$, e.g. RTS
  - **Medium** interaction: $O(n \cdot \log(n))$ – RPG quest maps
  - **High** interaction: $O(n^2)$ – RPG war maps, FPS
  - **Very High** interaction: $O(n^2 \cdot \log(n))/O(n^3)$, unit-target matching, team path-finding, maxflow alg., ...
Platform Scalability Challenge

1. Introduction
   1. Online Game Types
   2. Game Models

2. (Scheduling) Hosting Models
   1. Cluster + Parallelization
   2. Multi-cluster + Sharding
   3. Cloud-based + Sharding
   4. The SuperServer (video)

3. (Naming) Locating Servers
   1. Central directory
   2. Peer-to-Peer

4. Consistency
   1. (Eventual) Dead-reckoning
   2. (Continuous) Simultaneous Sim
   3. (Continuous) + Pipelining
   4. (Continuous) + Dynamic Tick
   5. (Continuous/Eventual) Area of Simulation vs Interest

5. Other issues

> Gaming Analytics Challenge
> Content Generation Challenge
Online games hosting model

• Generic Online Games (non-MM)
  • **Static**: dedicated isolated single servers

• MMOGs
  • **Static**: dedicated clusters - using parallelization techniques

• Problems with these approaches
  1. Large amount of over-provisioning
  2. Non-efficient coverage of the world for the provided service

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Game parallelization models

- **Models:**
  - **Zoning:** huge game-world division into geographical sub-zones – each zone is handled by different machines
  - **Mirroring:** the same game-world handled by different machines, each one handling a subset of the contained entities (synchronized states)
  - **Instancing/sharding:** multiple instances of the same zone with independent states. (World of Warcraft, Runescape,..)
Proposed hosting model: dynamic

- Using data centers for dynamic resource allocation

- Main advantages:
  1. Significantly lower over-provisioning
  2. Efficient coverage of the world is possible

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Cloud Computing

VENI – @larGe: Massivizing Online Games using Cloud Computing
Dynamic MSG Ecosystem: Model

• **Game operators**
  - Past player activity/business model → **Predicted load** → **requests**

• **Data centers**
  - **Local** time-space **renting policy** → **offers**
  - Time-Space renting policy, e.g., 1 node-hour

• Resource allocation: central request-offer matching

• Rules for ranking request-offer match:
  - The offer size and type vs. the request, The geographical proximity offer-request, etc.

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Experimental Setup [1/3] Discrete-Event Simulator

• **Input**
  - Traces from RuneScape, a real top-5 MMOG
    - 7 countries, 3 continents
    - More than 130 game worlds
  - Consisting of
    - Geographical location
    - Number of clients
    - Over 10,000 samples at 2 min. interval, 2 weeks

• **Output** (for every time-step)
  - Resource allocation decisions
  - Resource allocation
  - Performance metrics

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Experimental Setup [3/3]  
Performance Metrics

• Resource over-provisioning [%]  
  • The wasted resources vs. optimal provisioning at each simulation time step for all utilized machines (cumulative)

• Resource under-provisioning [%]  
  • The amount of resources needed but not allocated, for all machines (computed individually)

• Significant under-provisioning events (count)  
  • Count of events: resource under-provisioning is >1%, for a period of 2 minutes → people leave

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Resource Provisioning and Allocation

**Static vs. Dynamic Provisioning**

![Graph showing over-provisioning over time with 250% and 25% highlights.]

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Also Studied

- Via real game measurements
  - Interactivity model (short-term msmt.)
  - Effects of underperforming platform (long-term msmt.)

- Via prototype implementation
  - Match model-reality [TPDS’11]

- Via simulation
  - Impact of virtualization [NetGames’11][IJAMC’11]
  - Economic models [CAC’13]
Remaining Challenge in Perf. Eng.: To the Real IaaS Cloud

VS

• “The path to abundance”
• On-demand capacity
• Cheap for short-term tasks
• Great for web apps (EIP, web crawl, DB ops, I/O)

• “The killer cyclone”
• (1) Not so great performance for scientific applications (compute- or data-intensive)
• (2) Unstable performance

2- Iosup et al., On the Performance Variability of Production Cloud Services, CCGrid 2011, pds.twi.tudelft.nl/reports/2010/PDS-2010-002.pdf
Remaining Challenge in Scheduling: Provisioning and Allocation Policies

Provisioning

When?
From where?
How many?
Which type? etc.

Allocation

When?
Where?
etc.

(Source: A. Antoniou, MSc Defense, TU Delft, 2012.)
An Example: Portfolio Scheduling for Online Gaming (also for Scientific Workloads)

- **CoH** = Cloud-based, online, Hybrid scheduling
  - Intuition: keep rental cost low by finding good mix of machine configurations and billing options
  - Main idea: *portfolio scheduler* = run *both* solver of an Integer Programming Problem and various heuristics, then pick best schedule at deadline
  - Additional feature: Can use *reserved cloud instances*

- Promising early results, for *Gaming* (and scientific) workloads

<table>
<thead>
<tr>
<th>Trace</th>
<th>#jobs</th>
<th>average runtime [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grid5000</td>
<td>200,450</td>
<td>2728</td>
</tr>
<tr>
<td>LCG</td>
<td>188,041</td>
<td>8971</td>
</tr>
<tr>
<td>DotaLicious</td>
<td>109,251</td>
<td>2231</td>
</tr>
</tbody>
</table>

Shen, Deng, Iosup, Epema, Scheduling Jobs in the Cloud Using On-demand and Reserved Instances, EuroPar’13
Remaining Challenge in Deployment: Zynga zCloud: Hybrid Self-Hosted/EC2

- After Zynga had large scale

- More efficient self-hosted servers
  - Run at high utilization

- Use EC2 for unexpected demand

“Cloud Gaming” = Interactive Gaming Videos from the Cloud

- Gaikai, OnLive—servers in own data centers, HDTV 720p

This is only one type of cloud offloading

\[ \text{End-to-End Service Delay} = \text{Transmission} + \text{Processing} + \text{Probing range} + \text{Playout} = \text{Decoding} + \text{Display} \]

Kuan-Ta Chen et al., Measuring the latency of cloud gaming systems. ACM Multimedia 2011: 1269-1272.
Remaining Challenges: Many Cloud Offloading Alternatives

Cases to investigate:
1. server in cloud
2. server behind cloud
3. clients in same LAN
4. hybrid

June 3, 2013

Work in Progress. With Alexandru Olteanu.
An Offloading Use Case: the OpenTTD Client (Early Work)

Game Parameters:
- map size
- number of players
- number of cities
- number of resources
- animations on/off

Comm. Mgmt

squ-VM

GUI

Input capture

Simulator

Renderer

snd
rcv
doCmd
world_state
img
doCmd

June 3, 2013

Work in Progress. With Alexandru Olteanu.
“Cloud Gaming” = Interactive Gaming Videos from the Cloud [2/3]

- Measurement approach
  - Reverse engineer calls
  - Inject measurement probes into library calls
  - Smart detection of $t_3$, up to and including Decoding

Kuan-Ta Chen et al., Measuring the latency of cloud gaming systems. ACM Multimedia 2011: 1269-1272.
“Cloud Gaming” = Interactive Gaming Videos from the Cloud [3/3]

- Games used in experiments
  - Batman (action-adventure)
  - Warhammer 40k: DOW (RTS)
  - FEAR (FPS)

- Measurement results
  - OnLive better processing delay
  - FPS FEAR supported very well by OnLive
  - OnLive supports well Batman&DOW
  - StreamMyGame: 400-500ms delay
  - Similar playout delays

Kuan-Ta Chen et al., Measuring the latency of cloud gaming systems. ACM Multimedia 2011: 1269-1272.
1. Introduction
   1. Online Game Types
   2. Game Models
2. (Scheduling) Hosting Models
   1. Cluster + Parallelization
   2. Multi-cluster + Sharding
   3. Cloud-based + Sharding
   4. The SuperServer (video)
3. (Naming) Locating Servers
   1. Central directory
   2. Peer-to-Peer
4. Consistency
   1. (Eventual) Dead-reckoning
   2. (Continuous) Simultaneous Sim
   3. (Continuous) + Pipelining
   4. (Continuous) + Dynamic Tick
   5. (Continuous/Eventual) Area of Simulation vs Interest
5. Other issues

> Gaming Analytics Challenge
> Content Generation Challenge
Other Distributed Systems Issues
Consistency

- Eventual Consistency or Inconsistency
  - Dead-reckoning: eventual consistency with low message overhead
  - Smarter extensions

\[ t = \text{time} \]
\[ p = \text{position} \]
\[ v = \text{velocity} \]

Remember RTS Games?

- Players control tens up to hundreds of units.
- Players need to take decisions in real-time.

> To our approach for consistency: Area of Simulation
Other Distributed Systems Issues **Consistency: 1.5k Archers on 28.8k Line [1/3]**

- Age of Empires [Bettner & Terrano GDC01]
- Problem: Too many players/units to update at each click

⇒ **New Approach: Simultaneous simulations**

```
next render
u1: (x1,y1)
  u2: (x2,y2)
    ...
  un: (xn,yn)
```

```
Player1

left button clicked on (xd,yd)
```

```
Player2

left button clicked on (xd,yd)
```

```
next render
u1: (x1,y1)
  u2: (x2,y2)
    ...
  un: (xn,yn)
```

"Turn-based": in each turn, receive messages from others, process/simulate, and render

Other Distributed Systems Issues **Consistency:**

1.5k Archers on 28.8k Line [2/3]

- **Problem:** need very long turn to finish everything!
  - **Approach:** Pipelining of operations, have multi-turn tick

![Diagram showing the process of turn-based rendering and interaction between players](image)

**Player1**
- Left button clicked on \((x_1, y_1)\)
- Next render

**Player2**
- Left button clicked on \((x_d, y_d)\)
- Message received

*Turn 1*
- \((x_1, y_1)\)
- \((x_2, y_2)\)
- ...\((x_n, y_n)\)

*Turn 2*
- Message received
- \((x_1, y_1)\)
- \((x_2, y_2)\)
- ...\((x_n, y_n)\)

*Turn 3*
- Next render

**Problem:** latency/processing time vary with entity interaction (remember the \(O(n^3)\) interaction model?)


Other Distributed Systems Issues

Consistency: 1.5k Archers on 28.8k Line [3/3]

• Approach: dynamic turn length
  • Adjusts to real delays experienced by real players

Regular Net/CPU
200 ms latency
50 ms proc/render

<table>
<thead>
<tr>
<th>Communications turn (200 msec) - scaled to 'round-trip ping' time estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process all messages</td>
</tr>
<tr>
<td>50 msec</td>
</tr>
</tbody>
</table>

Slow Net/Reg. CPU
1000 ms latency
50 ms proc/render

<table>
<thead>
<tr>
<th>Communications turn (1000 msec) - scaled to 'round-trip ping' time estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process all messages</td>
</tr>
<tr>
<td>50 msec</td>
</tr>
</tbody>
</table>

Reg. Net/Slow CPU
200 ms latency
100 ms proc/render

<table>
<thead>
<tr>
<th>Communications turn (200 msec) - scaled to 'round-trip ping' time estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process all messages</td>
</tr>
<tr>
<td>100 msec</td>
</tr>
</tbody>
</table>

• Problem: slow turns. Could we use only Area of Interest?

Slides source: http://www.cs.duke.edu/courses/spring08/cps214/lectures/lecture18.ppt
Traditional AoI does not work

- Area of Interest (AoI) = traditional mechanism for RPG: only receive information around avatar, but...
- ...In RTS, each player has tens of units under control, so much more data to be transferred
- ... In RTS, players change focus (interest) more often than in RPG and FPS, so higher management overhead
Core Idea

- Partition the game into multiple areas (rectangular)

- Each player pays attention to different areas + attention level

- Depending on attention level and machine performance, the player will receive different types of information (commands or state) about the game world
  - AoS: Area of Simulation = high-attention area, full simulation based on input commands (CPU-intensive)
  - AoU: Area of Update = low-attention area, receives state (Net)
  - NUA: No update area

- Each player can have multiple AoS and AoU
Experimental results

- Simulator and prototype RTS game
- Evaluate in two Cloud platforms: EC2 and Azure
- Prototype about 20k lines of C++ code
  - Based on an open source game (~6k lines)
- Up to 200 players and **10,000 battle units**
  - State-of-the-art unplayable at 1-2,000
  - Crashes not uncommon due to CPU and Network bottlenecks

- ➔ Using our AoS-based method can lead to **lower CPU** consumption than pure event-based method (RTS) and **lower network consumption** than pure update-based method (RPG)

---

Work in Progress
AoS << RTS  
AoS ~ RPG

Server CPU

Server Network

Client CPU

Client Network

Work in Progress
Agenda

1. What’s in a Name?
2. OpenTTD@large

3. Three Current Challenges
   1. Platform Scalability Challenge
   2. Gaming Analytics Challenge
   3. Content Generation Challenge

4. The Next Five Years
5. Conclusion
“When you play a number of games, not as ends unto themselves but as parts of a larger game, you are participating in a metagame.” (Dr. Richard Garfield, 2000)

XFire: since 2008 (3+ years), 500K of 20M players

S. Shen, and A. Iosup, The XFire Online Meta-Gaming Network: Observation and High-Level Analysis, MMVE 2011
@large Research Challenge: Continuous Analytics for MMOGs

Analyzing the behavior of millions of players, on-time
- **Data mining**, data access rights, cost v. accuracy, ...
- Reduce upfront costs
- Low response time & Scalable
- Large-scale Graph Processing
How to Analyze? Ecosystems of Big-Data Programming Models

Adapted from: Dagstuhl Seminar on Information Management in the Cloud, http://www.dagstuhl.de/program/calendar/partlist/?semmr=11321&SUOG
A Use-Case in Gaming Analytics, based on Defence of the Ancients (DotA)

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
DotA communities

- Players are loosely organised in communities
  - Operate game servers
  - Maintain lists of tournaments and results
  - Publish statistics and rankings on websites

- Dota-League: players join a queue and matchmaking forms teams
- DotAlicious: players can choose which match/team to join

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Our Datasets

• We have crawled all matches played and per match have:
  • Names of the players for each team
  • Active, start and end time
  • Game-play statistics per team
  • The team that has won

• Dota-League:
  • ~1.5M matches played between Nov’08 and Jul’11, 61K players

• DotAlicious:
  • ~0.6M matches played between Apr’10 to Feb’12, 62K players

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
From game instances to social ties

- We need to define how to map the relationships found in real-world matches to a **gaming graph** (nodes and links).

- We use six different mappings and various thresholds:
  - **SM**: two players occur more than $n$ times in the **same match**
  - **SS**: two players occur more than $n$ times on the **same side**
  - **OS**: two players occur more than $n$ times on **opposing sides**
  - **ML**: two players have **lost** more than $n$ **matches together**
  - **MW**: two players have **won** more than $n$ **matches together**
  - **PP**: a directed version of the mappings above. A link exists if a player has played more than $n$ percent of his matches together.
Network sizes (w/o isolated nodes) in the Gaming Graph

Number of nodes in the network as a function of the threshold

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Small clusters show strong ties in the gaming graph

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Relationships in the gaming graph

• Players who regularly play together in DotAlicious do so in more diverse combinations than in Dota-League

• Contrary to Dota-League, DotAlicious players tend to play on the same side: playing together intensifies the social bond

• Winning together increases friendship relationships, while loosing together weakens friendship relationships

• Small clusters of friends with very strong social ties exist

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Matchmaking application

- Replay match list, but also consider clusters in gaming graph
- Scoring methodology:
  - Points per cluster: Number of players in the match that are part of the same cluster
  - Excluding largest cluster of the network and clusters of size 1

<table>
<thead>
<tr>
<th>Player</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1</td>
</tr>
<tr>
<td>b</td>
<td>2</td>
</tr>
<tr>
<td>c</td>
<td>1</td>
</tr>
<tr>
<td>d</td>
<td>3</td>
</tr>
<tr>
<td>e</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Player</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>f</td>
<td>2</td>
</tr>
<tr>
<td>g</td>
<td>5</td>
</tr>
<tr>
<td>h</td>
<td>3</td>
</tr>
<tr>
<td>i</td>
<td>6</td>
</tr>
<tr>
<td>j</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Team 1: Total of 7 points for this match

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Results matchmaking

Dota-League

Can do much better than random matchmaking

Can already improve original matchmaking algorithm for all gaming graphs!

DotAlicious

R. van de Bovenkamp, S. Shen, A. Iosup, F. A. Kuipers: Understanding and recommending play relationships in online social gaming. COMSNETS 2013: 1-10
Agenda

1. What’s in a Name?
2. OpenTTD@large
3. Three Current Challenges
   1. Platform Scalability Challenge
   2. Gaming Analytics Challenge
   3. Content Generation Challenge
4. The Next Five Years
5. Conclusion
**@large: Sample Analytics Results**

**Skill Level Distribution in RuneScape**

- Runescape: 135M active accounts, 7M active (2008)
- *(largest MMOG msmt.)*

- **Player skill: distribution changes over time**

---

@large Research Challenge: Content Generation for MMOGs

Generating content on time for millions of players
- Player-customized: Balanced, Diverse, Fresh
- Up-front and operational costs
- Response time, Scalability, & Cost
(Procedural) Game Content (Generation)

Derived Content
NewsGen, Storification

Game Design
Rules, Mechanics, ...

Game Scenarios
Puzzle, Quest/Story, ...

Game Systems
Eco, Road Nets, Urban Envs, ...

Game Space
Height Maps, Bodies of Water, Placement Maps, ...

Game Bits
Texture, Sound, Vegetation, Buildings, Behavior,
Fire/Water/Stone/Clouds

Hendricks, Meijer, vd Velden, Iosup,
Procedural Content Generation for Games: A Survey,
ACM TOMCCAP, 2012
The POGGI Content Generation Framework

Only the puzzle concept, and the instance generation and solving algorithms, are produced at development time.

Distributed system to generate instances on-demand, reliably, efficiently, and with performance guarantees.

* A. Iosup, POGGI: Puzzle-Based Online Games on Grid Infrastructures, EuroPar 2009 (Best Paper Award)
Puzzle-Specific Considerations Generating Player-Customized Content

Puzzle difficulty
- Solution size
- Solution alternatives
- Variation of moves
- Skill moves

Player ability
- Keep population statistics and generate enough content for most likely cases
- Match player ability with puzzle difficulty
- Take into account puzzle freshness
Agenda

1. What’s in a Name?
2. OpenTTD@large
3. Three Current Challenges
4. The Next Five Years
   1. Cloudification
   2. Mobile Social Gaming
   3. Social Everything!
   4. Content, Content, Content
5. Conclusion
Cloudification: PaaS for MSGs

(Platform Challenge)
Build MSG platform that uses (mostly) cloud resources

- Close to players
- No upfront costs, no maintenance
- Compute platforms: multi-cores, GPUs, clusters, all-in-one!
- Performance guarantees
- Hybrid deployment model
- Code for various compute platforms—platform profiling
- Load prediction miscalculation costs real money
- What are the services?
- Vendor lock-in?
- My data
Mobile Social Gaming and the SuperServer

(Platform Challenge)
Support MSGs on mobile devices

- Mobiles everywhere (2bn+ users)
- Gaming industry for mobiles is new Growing Market
- SuperServer to generate content for low-capability devices?
- Battery for 3D/Networked games?
- Where is my server? (Ad-hoc mobile gaming networks?)
- Security, cheat-prevention
Social Everything!

- **Social Network** = undirected graph, **relationship** = edge
- **Community** = sub-graph, density of edges between its nodes higher than density of edges outside sub-graph

(Analytics Challenge) Improve gaming experience

- Ranking / Rating
- Matchmaking / Recommendations
- Play Style/Tutoring

Self-Organizing Gaming Communities

- Player Behavior
(Content Challenge)

Produce and distribute content for 1BN people

- Game Analytics → Game statistic
- Crowdsourcing
- Storification
- Auto-generated game content
- Adaptive game content
- Content distribution/Streaming content
Massivizing Online Gaming

- Million-user, multi-bn market
- V-World, Content, Analytics

Current Technology

- Upfront payment
- Cost and scalability problems
- Makes players unhappy

@large: Our Vision

- DC/Cloud has to help
- Economy of scale with clouds

@large: Ongoing Work

- Content: POGGI Framework
- Platform: edutain@grid
- Analytics: CAMEO Framework

@large: The Future

- Happy players
- Happy cloud operators

Publications

<table>
<thead>
<tr>
<th>Year</th>
<th>Gaming and Clouds</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>ACM SC</td>
</tr>
<tr>
<td>2009</td>
<td>ROIA, CCGrid, NetGames, EuroPar (Best Paper Award), ...</td>
</tr>
<tr>
<td>2010</td>
<td>IEEE TPDS, Elsevier CCPE</td>
</tr>
<tr>
<td>2011</td>
<td>Book Chapter CAMEO, IEEE TPDS, IJAMC</td>
</tr>
<tr>
<td>2012</td>
<td>IPDPS, CCGrid, ...</td>
</tr>
</tbody>
</table>

Graduation (Forecast)

2012–14: 3PhD, 6Msc, 6BSc

Massivizing Social Games: High Performance Computing and High Quality Time – A. Iosup

TU Delft
Delft University of Technology
Thank you for your attention! Questions? Suggestions? Observations?

More Info:
- http://www.st.ewi.tudelft.nl/~iosup/research.html
- http://www.st.ewi.tudelft.nl/~iosup/research_gaming.html
- http://www.st.ewi.tudelft.nl/~iosup/research_cloud.html

Alexandru Iosup

A.Iosup@tudelft.nl
http://www.pds.ewi.tudelft.nl/~iosup/ (or google “iosup”)
Parallel and Distributed Systems Group
Delft University of Technology
Platform Scalability Challenge

1. Introduction
   1. Online Game Types
   2. Game Models
2. (Scheduling) Hosting Models
   1. Cluster + Parallelization
   2. Multi-cluster + Sharding
   3. Cloud-based + Sharding
   4. The SuperServer (video)
3. (Naming) Locating Servers
   1. Central directory
   2. Peer-to-Peer
4. Consistency
   1. (Eventual) Dead-reckoning
   2. (Continuous) Simultaneous Sim
   3. (Continuous) + Pipelining
   4. (Continuous) + Dynamic Tick
   5. (Continuous/Eventual) Area of Simulation vs Interest
5. Other issues
Other Distributed Systems Issues
Naming (Locating Servers)

- Master directory
  - Web page with hundreds of servers (Minecraft)
  - TCP/IP-based protocol to get information from master server

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of advertised servers</td>
<td>7984</td>
<td>20930</td>
<td>18891.10</td>
<td>18789</td>
</tr>
<tr>
<td>Number of active servers</td>
<td>2458</td>
<td>7588</td>
<td>5659.08</td>
<td>5804</td>
</tr>
<tr>
<td>Active servers (as % of advertised)</td>
<td>16.32</td>
<td>72.19</td>
<td>30.21</td>
<td>30.71</td>
</tr>
<tr>
<td>Time taken for query (seconds)</td>
<td>310</td>
<td>676</td>
<td>656.45</td>
<td>670</td>
</tr>
</tbody>
</table>

- Peer-to-peer architecture

Platform Scalability Challenge

1. Introduction
   1. Online Game Types
   2. Game Models
2. (Scheduling) Hosting Models
   1. Cluster + Parallelization
   2. Multi-cluster + Sharding
   3. Cloud-based + Sharding
   4. The SuperServer (video)
3. (Naming) Locating Servers
   1. Central directory
   2. Peer-to-Peer
4. Consistency
   1. (Eventual) Dead-reckoning
   2. (Continuous) Simultaneous Sim
   3. (Continuous) + Pipelining
   4. (Continuous) + Dynamic Tick
5. Other issues
Other Distributed Systems Issues

- Operation of replicated servers: performance guarantees
- Operation with slow user clients/networks
- Persistent worlds
- Content distribution
- The whole CAP spectrum
  - Consistency
  - Availability
  - Partition-tolerance

http://www.popscreen.com/v/6wEHS/Minecraft-Epic-Fail-Creeper
Impact on Game Experience

Responsive game

Unresponsive game

June 3, 2013

[Source: Nae, Iosup, and Prodan, ACM SC 2008 and IEEE TPDS 2011]
Experimental Setup [2/3]  
Environment

- 1 game operator
- 17 data centers
- 11 data center time-space renting policies

<table>
<thead>
<tr>
<th>Location</th>
<th>Continent</th>
<th>Country</th>
<th>Data Centers</th>
<th>Machines (total)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Europe</td>
<td>Finland</td>
<td>2</td>
<td>8 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sweden</td>
<td>2</td>
<td>8 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>U.K.</td>
<td>2</td>
<td>20 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Netherlands</td>
<td>2</td>
<td>15 machines</td>
<td></td>
</tr>
<tr>
<td>North America</td>
<td>U.S. (West)</td>
<td>2</td>
<td>35 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Canada (West)</td>
<td>1</td>
<td>15 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>U.S. (Central)</td>
<td>1</td>
<td>15 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>U.S. (East)</td>
<td>2</td>
<td>32 machines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Canada (East)</td>
<td>1</td>
<td>10 machines</td>
<td></td>
</tr>
<tr>
<td>Australia</td>
<td>Australia</td>
<td>2</td>
<td>8 machines</td>
<td></td>
</tr>
</tbody>
</table>

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Impact of Load Prediction Accuracy

Q: How does the prediction accuracy impact resource provisioning? A: Good prediction matters.

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Q: How are different MMOG types handled under dynamic resource provisioning?

(Interaction models Low \( \sim O(n) \), Medium, High, Very High \( \sim O(n^3) \))

A: Over-provisioning, Under-provisioning worse with increase in interaction compute-intensiveness

[Source: Nae, Iosup, and Prodan, ACM SC 2008]
Latency Tolerance: From None to High

Q: What is the impact of latency tolerance on hosting?

A: (left) very sensitive very costly
    (mid) sensitive costly
    (right) non-sensitive cheap
Bridge Base Online (BBO): 1M+ players, top free site

Dataset: 100K players
- 9K group
- Social relationships from bridge pairing

Large (~10K) online social groups can coordinate

Identified player behavior
- Community builder
- Community member
- Random player
- Faithful player

M. Balint, V. Posea, A. Dimitriu, and A. Iosup, An Analysis of Social Gaming Networks in Online and Face to Face Bridge Communities, LSAP 2011.
Resource Provisioning and Allocation
Compound Metrics

- Trade-off Utility-Cost still needs investigation
- Performance and Cost are not both improved by the policies we have studied

Massivizing Social Games: High Performance Computing and High Quality Time – A. Iosup

(Variable) Blackbox Performance Engineering

- Performance Evaluation of Four Commercial Clouds
  - Amazon EC2, GoGrid, Elastic Hosts, Mosso
  - Resource acquisition
  - Single- and Multi-Instance benchmarking

- Low compute and networking performance\(^1\)

- Performance variability over time\(^2\)

---

2- Iosup et al., On the Performance Variability of Production Cloud Services, CCGrid 2011, pds.twi.tudelft.nl/reports/2010/PDS-2010-002.pdf
• Time for *multi*-resource increases with number of resources
**GAE Dataset: Run Service**

- **Fibonacci [ms]**: Time it takes to calculate the 27th Fibonacci number
- Highly variable performance until September
- Last three months have stable performance (low IQR and range)
Online Scheduling + Optimization

ExPERT

ExPERT recommended:

\[(N = 3, T = T_{ur}, D = 2T_{ur}, M_r = 0.02), \text{ in words:} \]

Send \( N = 3 \) instances to the unreliable pool during the tail phase, each timed out after twice the average task time \( (D = 2T_{ur}) \). Send the next instance after the average task time passes \( (T = T_{ur}) \). Use only one \( (\#ur = 50, 50 \times M_r = 1) \) reliable machine at a time.

Performance Metrics

- Makespan very similar
- Very different job slowdown
Cost Metrics

Actual Cost

- Very different results between actual and charged
- Cloud charging function an important selection criterion
- All policies better than Startup in actual cost
- Policies much better/worse than Startup in charged cost

Charged Cost

Single Resource Provisioning/Release

- Time depends on instance type
- Boot time non-negligible

Massivizing Social Games: High Performance Computing and High Quality Time – A. Iosup

CPU Performance of Single Resource

- ECU definition: “a 1.1 GHz 2007 Opteron” ~ 4 flops per cycle at full pipeline, which means at peak performance one ECU equals 4.4 gigaflops per second (GFLOPS).
- Real performance 0.6..0.1 GFLOPS = ~1/4..1/7 theoretical peak.
HPLinpack Performance (Parallel)

- Low efficiency for parallel compute-intensive applications
- Low performance vs cluster computing and supercomputing

Massivizing Social Games: High Performance Computing and High Quality Time – A. Iosup

Performance Stability (Variability)

- High performance stability for the best-performing instances

---

• **All services exhibit time patterns in performance**
  - **EC2**: periods of special behavior
  - **SDB and S3**: daily, monthly and yearly patterns
  - **SQS and FPS**: periods of special behavior
- **Read Latency [s]**: Time it takes to read a “User Group”
- Yearly pattern from January to August
- The last four months of the year exhibit much lower IQR and range
  - More stable performance for the last five months
  - Probably due to software/infrastructure upgrades
GAE Dataset (3/4): Memcache

- **PUT [ms]**: Time it takes to put 1 MB of data in memcache.
- Median performance per month has an increasing trend over the first 10 months.
- The last three months of the year exhibit stable performance.

---

GAE Dataset (4/4): Summary

- All services exhibit time patterns
  - **Run Service**: daily patterns and periods of special behavior
  - **Datastore**: yearly patterns and periods of special behavior
  - **Memcache**: monthly patterns and periods of special behavior
  - **URL Fetch**: daily and weekly patterns, and periods of special behavior
@large: Social Everything!

- **Social Network** = undirected graph, **relationship** = edge
- **Community** = sub-graph, density of edges between its nodes higher than density of edges outside sub-graph

(Analytics Challenge)

**Improve gaming experience**
- Ranking / Rating
- Matchmaking / Recommendations
- Play Style/Tutoring

**Self-Organizing Gaming Communities**
- Player Behavior
The Game Trace Archive (upcoming)

- Share gaming traces and best-practices on using them
- Support simulations and real-world experiments

<table>
<thead>
<tr>
<th>Name</th>
<th>Period</th>
<th>Size (GB)</th>
<th>Node (M)</th>
<th>Edge (M)</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>KGS</td>
<td>2002/02-2009/03</td>
<td>2</td>
<td>0.8</td>
<td>27.4</td>
<td>Chess Game</td>
</tr>
<tr>
<td>FICS</td>
<td>1997/11-2011/09</td>
<td>168</td>
<td>0.4</td>
<td>144.2</td>
<td>Chess Game</td>
</tr>
<tr>
<td>BBO</td>
<td>2009/11-2009/12</td>
<td>10</td>
<td>3.9</td>
<td>12.9</td>
<td>Card Game</td>
</tr>
<tr>
<td>XFire</td>
<td>2008/05-2011/12</td>
<td>58</td>
<td>7.7</td>
<td>34.7</td>
<td>OMGN</td>
</tr>
<tr>
<td>Dota League</td>
<td>2006/07-2011/03</td>
<td>23</td>
<td>0.1</td>
<td>3.0</td>
<td>RTS</td>
</tr>
<tr>
<td>DotaLicious</td>
<td>2010/04-2012/02</td>
<td>1</td>
<td>0.1</td>
<td>0.6</td>
<td>RTS</td>
</tr>
<tr>
<td>Dota Garena</td>
<td>2009/09-2010/05</td>
<td>1</td>
<td>0.3</td>
<td>0.1</td>
<td>RTS</td>
</tr>
<tr>
<td>WoWAH</td>
<td>2006/01-2009/10</td>
<td>3</td>
<td>0.1</td>
<td>N/A</td>
<td>MMORPG</td>
</tr>
</tbody>
</table>
The CAMEO Framework

1. Address community needs
   - Can analyze skill level, experience points, rank
   - Can assess community size dynamically

2. Using on-demand technology: Cloud Comp.
   - Dynamic cloud resource allocation, Elastic IP

3. Data management and storage: Cloud Comp.
   - Crawl + Store data in the cloud (best performance)

4. Performance, scalability, robustness: Cloud Comp.

A. Iosup, CAMEO: Continuous Analytics for Massively Multiplayer Online Games on Cloud Resources. ROIA, Euro-Par 2009 Workshops, LNCS 6043, (2010)
Focus on game content generation on grids

- Use existing middleware
- Control MMOG-specific workload demands and variability (soft guarantees for low response time by pre-generating content)

... but do not forget lessons on system design

- Add components for capacity planning and process monitoring
Continuous Analytics for MMOGs

MMOG Data = raw and derivative information from the virtual world (millions of users)

Continuous Analytics for MMOGs = Analysis of MMOG data s.t. important events are not lost

- Data collection
- Data storage
- Data analysis
- Data presentation
- ... at MMOG rate and scale
Continuous Analysis for MMOGs
Main Uses By and For Gamers

1. Support player communities
2. Understand play patterns (decide future investments)
3. Prevent and detect cheating or disastrous game exploits (think MMOG economy reset)
4. Broadcasting of gaming events
5. Data for advertisement companies (new revenue stream for MMOGs)
Other Uses for MMOG Data

Social Sciences
- The emergence and performance of ad hoc groups in contemporary society
- Emergent behavior in complex systems

Economy
- Contemporary economic behavior

Psychology
- Games as coping mechanism (minorities)
- Games as cure (agoraphobia)

Biology
- Disease spread models
The CAMEO Framework [ROIA09]
Continuous MSG Analytics on the Cloud

- Use own resources for continuous or predicted load
- Use cloud (on-demand, paid-for, guaranteed) resources for sparse or excess load
- Users (peers) may also provide service (future)

CAMEO: Analytics Capabilities

1. Various pieces of information
   - Skill level, experience points, rank

2. Single and Multi-snapshot analysis

3. Analysis functions already implemented
   - Ranking by one or more pieces of information
   - Community statistical properties for a piece of information
   - Identification of Top-K players in single/multi-snapshot
   - Evolution of (Top-)K players
   - Evolution of average community skill
   - Identification of players with special skill combos
CAMEO: Cloud Resource Management

- Snapshot = dataset for a set of players
- More machines = more snapshots per time unit
CAMEO: Exploiting Cloud Features

- Machines close(r) to server
  - Traffic dominated by small packets (latency)

- Elastic IP to avoid traffic bans
  (legalese: acting on behalf of real people)

Cost of Continuous RuneScape Analytics

### Billing Statement: April 1, 2009

**Billing Cycle for this Report:** March 1 - March 31, 2009

<table>
<thead>
<tr>
<th>Rate</th>
<th>Usage</th>
<th>Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Amazon Elastic Compute Cloud</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>View/Edit Service</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Amazon EC2 running Linux/UNIX</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$0.10 per Small Instance (m1.small) instance-hour (or partial hour)</td>
<td>2,097 Hrs</td>
<td>209.70</td>
</tr>
<tr>
<td><strong>Amazon EC2 Bandwidth</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$0.100 per GB Internet Data Transfer - all data transfer into Amazon EC2</td>
<td>611.005 GB</td>
<td>61.10</td>
</tr>
<tr>
<td><strong>Taxes</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$0.170 per GB Internet Data Transfer - first 10 TB / month data transfer out of Amazon EC2</td>
<td>507.121 GB</td>
<td>86.21</td>
</tr>
<tr>
<td><strong>Charges due on April 1, 2009†</strong></td>
<td></td>
<td>424.85</td>
</tr>
</tbody>
</table>

- Put a price on MMOG analytics (here, **$425/month**, or less than **$0.00015/user/month**)
- Trade-off accuracy vs. cost, runtime is constant
Performance Results: Why Choosing the Cloud Matters

- Location of machines influences MMOG analytics performance (data acquisition)
Player-Customized Content
Skill Level Distribution in RuneScape

- **RuneScape**: 135M+ open accounts (world record)
- Dataset: **3M players (largest measurement, to date)**
  - 1,817,211 over level 100
  - Max skill 2,280
- Number of mid- and high-level players is significant

New Content Generation Challenge

---

Only the puzzle concept, and the instance generation and solving algorithms, are produced at development time.

*A. Iosup, POGGI: Puzzle-Based Online Games on Grid Infrastructures, EuroPar 2009 (Best Paper Award)*
Bags-of-Tasks (BoTs)

- BoT = set of jobs...
  \[ W_u = \{ J_i | user(J_i) = u \} \]
  ...that start at most \( \Delta \)s after the first job
  \[ ST(J') \leq ST(J) + \Delta \]

- Parameter Sweep App. = BoT with same binary

Workflows (WFs)

- WF = set of jobs with precedence (think Direct Acyclic Graph)
Workflow Engines: Performance vs. Resource Consumption

<table>
<thead>
<tr>
<th>Middleware</th>
<th>MS [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAGMan</td>
<td>1,327 ± 138</td>
</tr>
<tr>
<td>Karajan</td>
<td>1,111 ± 154</td>
</tr>
</tbody>
</table>

Karajan performs better than DAGMan, but runs quickly out of resources.