



# **Project Darkstar: Next Generation Online Video Game Technology**

**Karl Haberl**

**Director, Project Darkstar**

**Jim Waldo**

**Architect, Project Darkstar**

**Sun Microsystems Labs**

# Online Games Market

- Divided into 3 groups:
  - > Casual/Social – cards, chess, dice, community sites
  - > Mass Market – driving, classic, arcade, simple
  - > Hardcore – MMOG, FPS, RTS
- Online mobile still very small
- Online games are currently the fastest growing segment of the games industry
- Online game subscriptions estimated to hit \$11B by 2011\* \*(Source: DFC intelligence)
  - > not including microtransactions, shared advertising, ...

# The Canonical MMOG: World of Warcraft™

- Approximately 9 million subscribers
  - > Average subscription : \$15/month
  - > Average retention : two years +
  - > \$135 million per month/\$1.62 Billion per year run rate
  - > For one game (they have others)
- Unknown number of servers
- ~2,700 employees world wide
- Company is changing
  - > Was a game company
  - > Now a service company



*World of Warcraft™ is a trademark and Blizzard Entertainment is a trademark or registered trademark of Blizzard Entertainment in the U.S. and/or other countries.*

Copyright 2008, Sun Microsystems, Inc.

# Ganz - Webkinz®

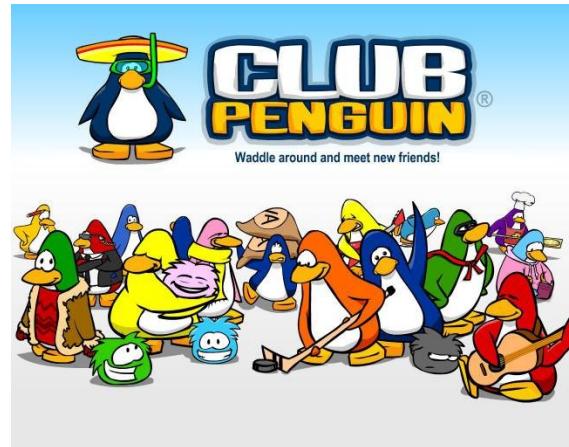
- Approximately 5+ million subscribers
  - > Subscription comes with toy purchase
  - > Subscription lasts one year
  - > Average 100k users at any time
  - > Currently only US and Canada; soon to be world wide
  - > Aimed at the 8-12 demographic
    - And their mothers...
- The company is changing
  - > Was a toy company
  - > Becoming a game/social site company



Webkinz® is a registered Trademark of Ganz®. Photographs and artwork © GANZ. GANZ, WEBKINZ, the WEBKINZ logo and all character names are trademarks of GANZ.

# Club Penguin™

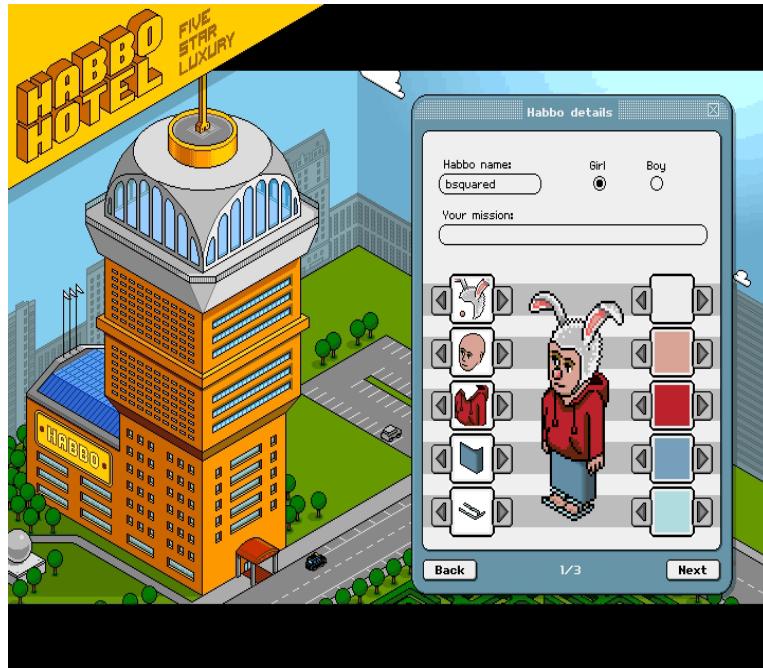
- Virtual world for kids to play and interact with other kids
- Approximately 12 million users
- 2 million active users (connected over a 30 day period)
- 700K paying subscribers @ \$5.95 per month



**Club Penguin** is a trademark of **Club Penguin Entertainment Inc.**. Club Penguin™ Club Penguin Entertainment Inc. © 2007. All rights reserved.

# Habbo Hotel™

- Virtual hotel for teens
  - > 89M accounts
  - > 8.3M unique users (12/07)
  - > 100K concurrent users peak
  - > will break 1B page hits per month
- Most revenue from content sales
- Grew to 1M users in first year
  - > started w 5 servers and 2 admins
- “Scaling was challenging” - Sulka Haro

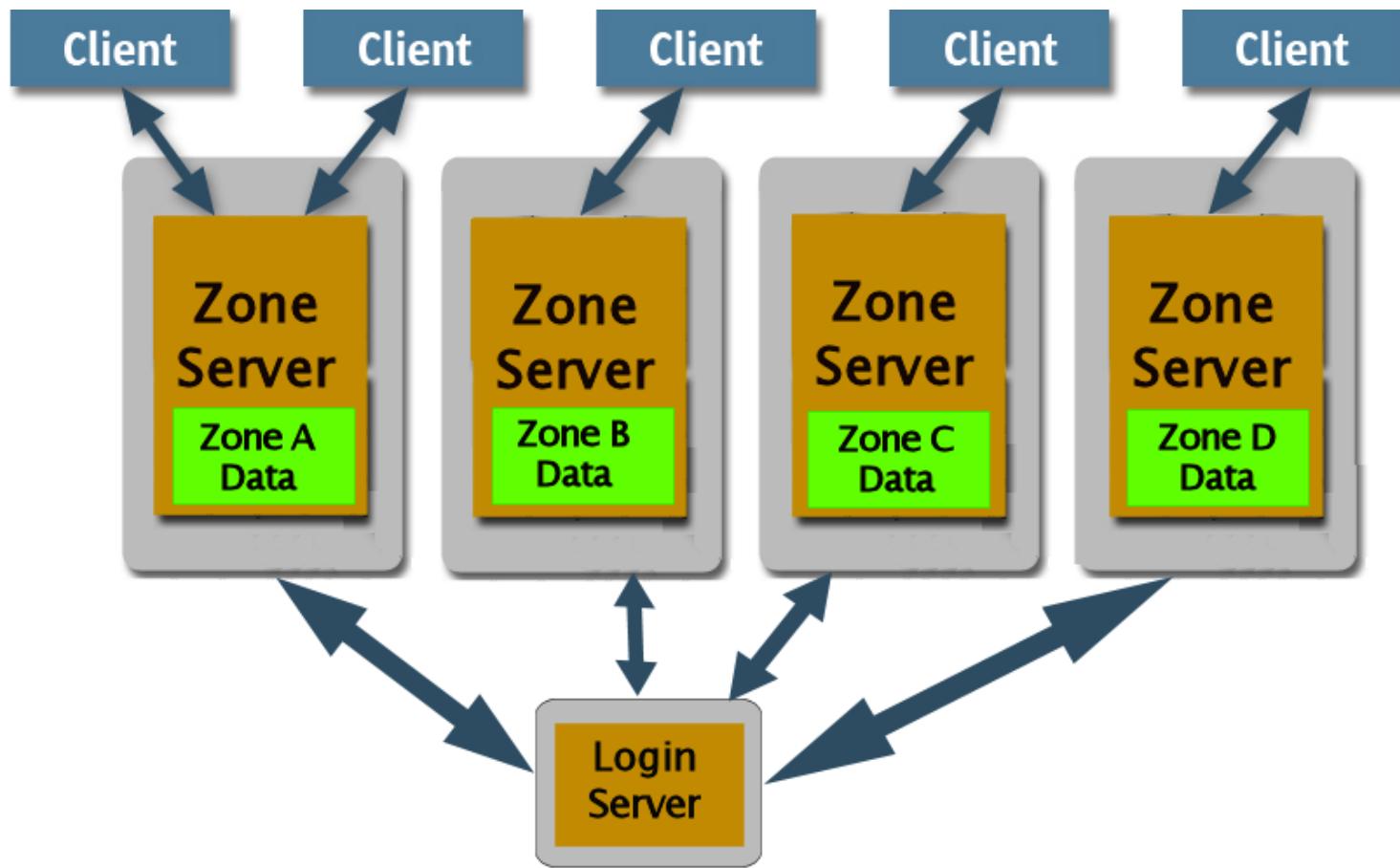


*Habbo Hotel © 2007 - 2008 Sulake Corporation Oy. HABBO is a registered trademark of Sulake Corporation Oy in the European Union, the USA, Japan, the People's Republic of China and various other jurisdictions. All rights reserved.*

# Current Scaling Techniques

- Geographic Decomposition - “Shards”
  - > One server = some geographic area
    - WoW: realm, Second Life: island, Nicktropolis/Webkinz: room
  - > Need to decide scale during production
  - > Get it wrong, game play impacted
  - > When server is full, must connect to a different shard
  - > No communication between shards; bad for guilds
  - > Empty shards = idle servers, poor utilization
  - > For social/casual, can be confusing for kids and adults

# Sharded Architecture

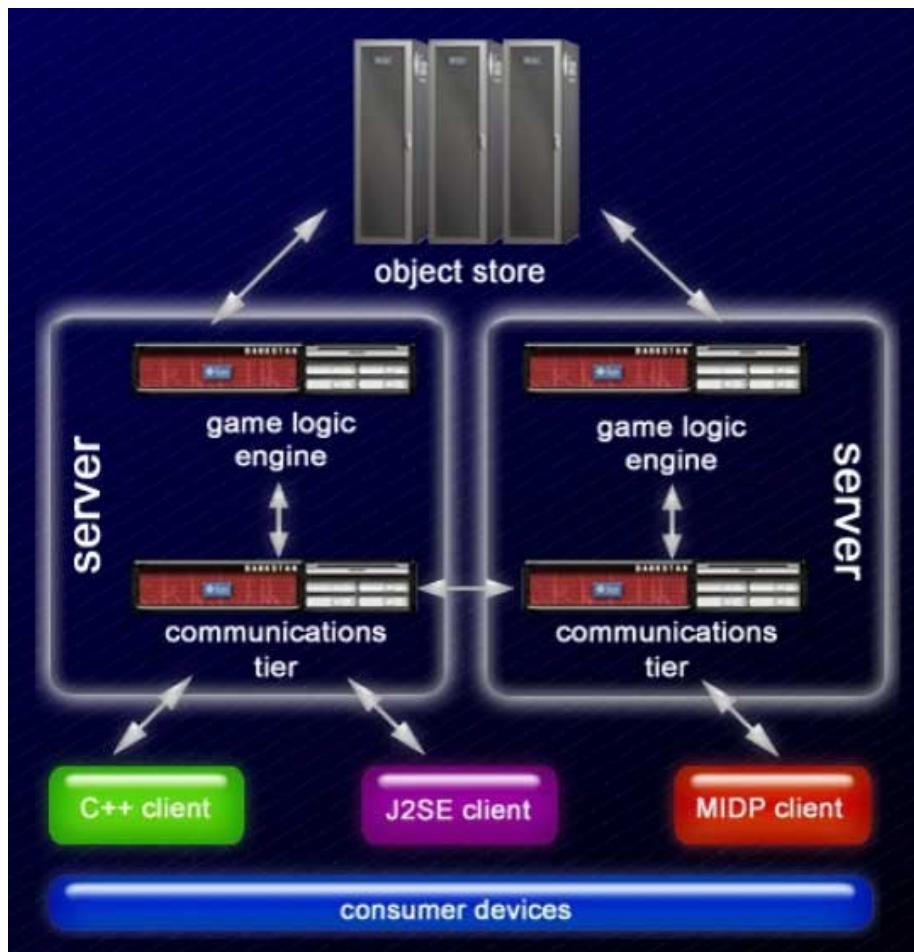


# State-of-play for on-line games

- Difficult and expensive to develop, deploy, and manage
  - > \$30M + multi-year development for big-time MMOGs
  - > Capacity management hard
    - hit games need to scale up dramatically, scale down duds
  - > Very risky – hard to predict success of game in market
- Only the big guys can play
  - > Large game studios like blockbusters, just like movie industry
  - > Lack of innovation in game design
- Scale and reliability are needed
  - > Sharded architectures limit scalability and player interaction
  - > One call to customer service = ~3 month subscription
  - > Chip architectures are changing – threads, not clocks!

# Project Darkstar

- A software server designed to change the develop-and-deploy model for multiplayer online games and virtual worlds
- Written entirely in Java<sup>TM</sup> programming language
- Game agnostic and platform agnostic
- Available as open source under GPLv2 license
  - > Commercial licenses and support can be provided



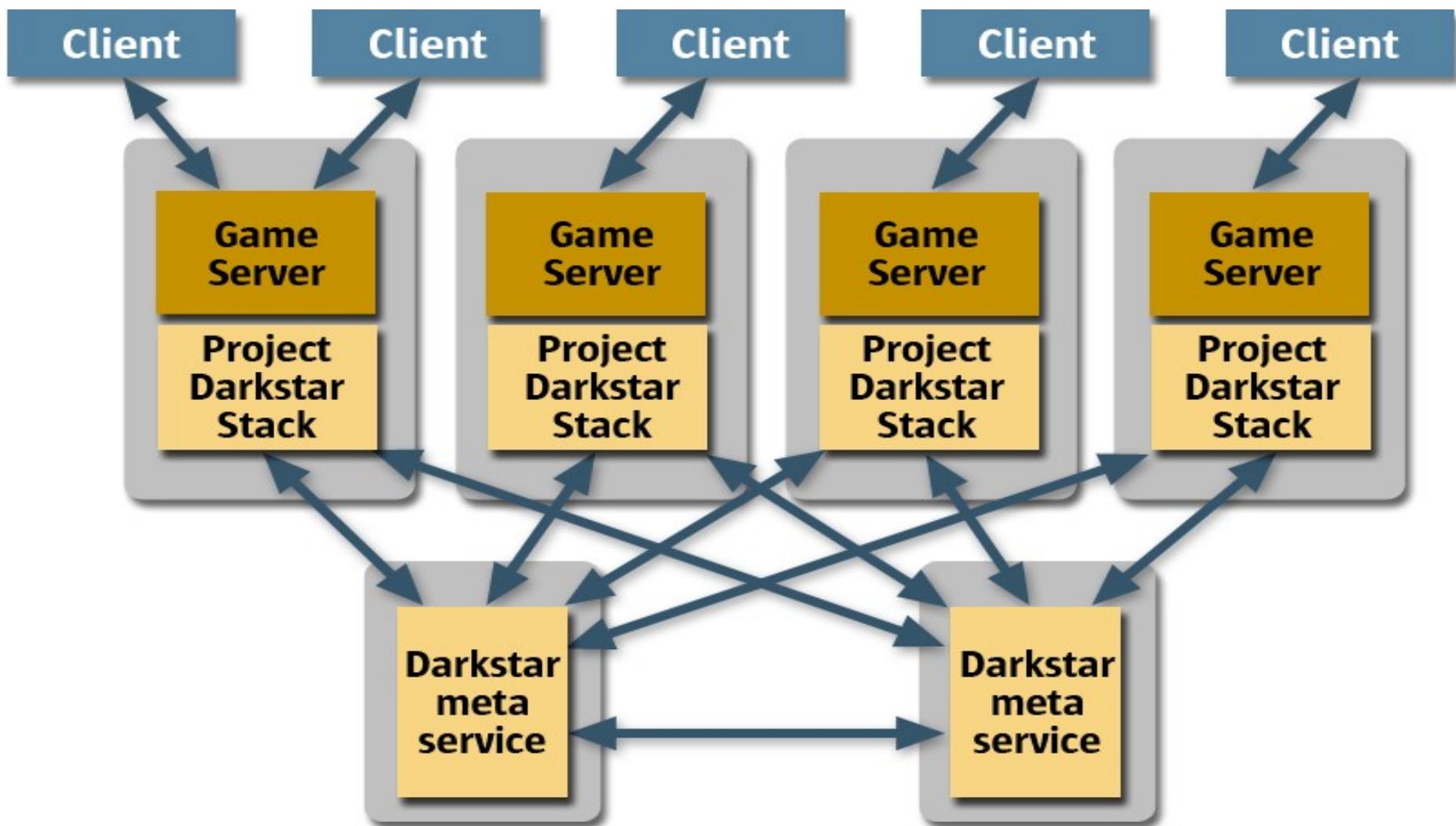
# Project Darkstar Goals

- Support Server Scale
  - > Games are embarrassingly parallel
  - > Multiple threads
  - > Multiple machines
- Simple Programming Model
  - > Multi-threaded, distributed programming is hard
  - > Single thread
  - > Single machine
- In the general case, this is impossible
  - > “A Note on Distributed Computing” - Waldo et al Nov1994
  - > <http://research.sun.com/techrep/1994/abstract-29.html>

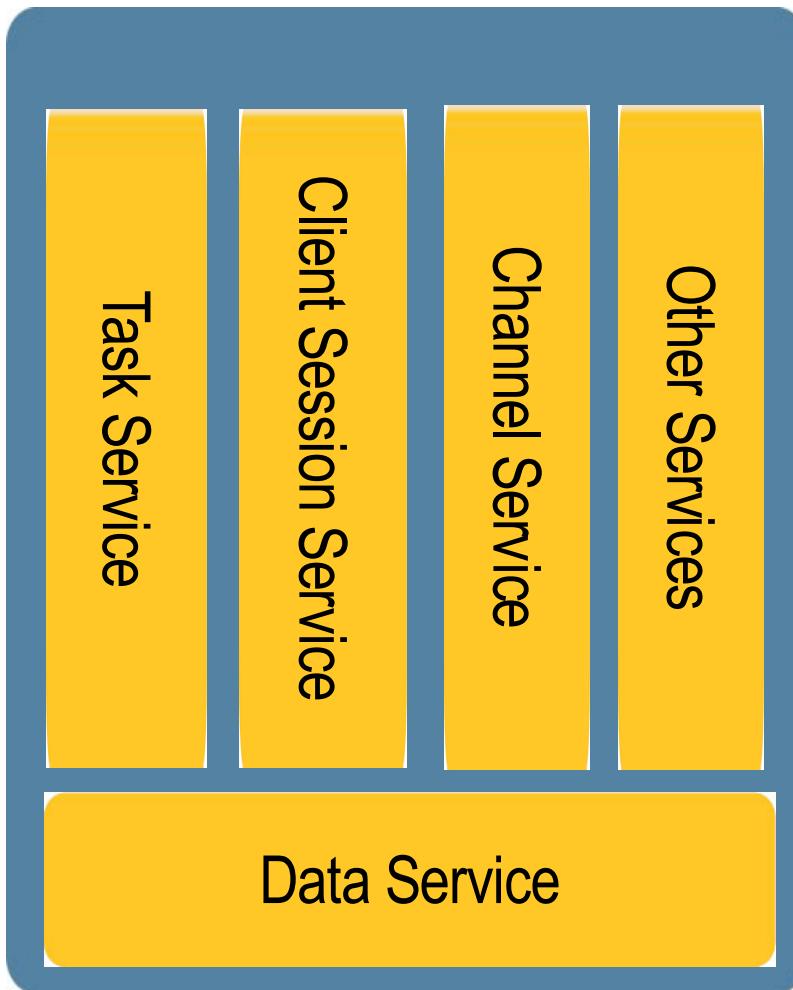
# The Special Case

- Event-driven Programs
  - > Client communication generates a task
  - > Tasks are independent
- Tasks must
  - > Be short-lived
  - > Access data through Darkstar
- Communication is through
  - > Client sessions (client to server)
  - > Channels (publish/subscribe client/server-to-client)

# Project Darkstar Architecture



# Stack Architecture



# Dealing with Concurrency

- All tasks are transactional
  - > Either everything is done, or nothing is
  - > Commit or abort determined by data access and contention
- Data access
  - > Data store detects conflicts, changes
  - > If two tasks conflict
    - One will abort and be re-scheduled
    - One will complete
- Transactional communication
  - > Actual communication only happens on commit

# Project Darkstar Data Store

- Not a full (relational) database
  - > No SQL
  - > Optimized for 50% read/50% write
- Keeps all game state
  - > Stores everything persisting longer than a single task
  - > Shared by all copies of the stack
- No explicit locking protocols
  - > Detects changes automatically
  - > Programmer can provide hints for optimizations

# Project Darkstar Communication

- Listeners hear client communication
  - > Simple client protocol
  - > Listeners established on connection
- Client-to-client through the server
  - > Allows server to listen if needed
  - > Very fast data path
- Mediation virtualizes end points
  - > Indirection abstracts actual channels

# Dealing with Distribution

- Tasks can run anywhere
  - > Data comes from the data store
  - > Communications is mediated
  - > Where a task runs doesn't matter
- Tasks can be allocated on different machines
  - > Players on different machines can interact
  - > The programmer doesn't need to choose
- Tasks can be moved
  - > Meta-services can track loads and move tasks
  - > New stacks can be added at runtime

# The End Result

- Simple and familiar programming model
  - > A single thread
  - > A single machine
- Multiple threads
  - > Task scheduling part of the infrastructure
  - > Concurrency control through the data store, transactions
- Multiple machines
  - > Project Darkstar manages data and communication references
  - > Computation can occur on any machine
  - > Machines can be added (or subtracted) at any time

# Project Darkstar Differentiators

- Comes from a tradition of enterprise class performance
- Simple programming model
- Shardless architecture
- Not a game engine
- Dynamic load balancing
- Server utilization
  - > Higher efficiencies
  - > Infrastructure flexibility and reuse
- Open and extensible
  - > 100% Java technology
  - > Open Source – GPL v2

# Project Darkstar Technology Status

- Latest release 0.9.5 - 07Feb08
  - > single node version w updated API set
  - > open source under a GPL v2 license
  - > Client APIs for Java and C, wire protocol spec published
- First multinode version currently running in lab
- Limited Darkstar Playground operational
- Numerous PoC and evaluation efforts underway
  - > some we know about, many we don't
- We are looking for developers

# Project Darkstar Roadmap

- Multinode (v1.0)
- Performance and Reliability Enhancements
- Expanded Developer and Management Tools
- Extended API set
- Automatic Load Balancing
- Expanded Darkstar Playground

# Project Darkstar Community

- <http://projectdarkstar.com>
  - > growing open source community
  - > downloads, source, forums, wikis, doc
  - > feedback, requirements, lessons learned
  - > subprojects: extensions, 3<sup>rd</sup> party integrations, tools
- Related projects and initiatives
  - > Project Wonderland: an open source toolkit for building 3D virtual worlds (built on Project Darkstar)
  - > Open Virtual Worlds project: collaboration of Sun Microsystems and New Media Consortium
- Come and participate!



# Resources

## **Project Darkstar**

<http://projectdarkstar.com/>

## **Project Wonderland**

<http://wonderland.dev.java.net/>

## **NMC Virtual Worlds**

<http://virtualworlds.nmc.org/>

## **Karl Haberl**

[karl.haberl@sun.com](mailto:karl.haberl@sun.com)