

20 Years of Eurobattle.net: A Retrospective on the PvPGN Server and Its Open Source Ecosystem

What is eurobattle.net?

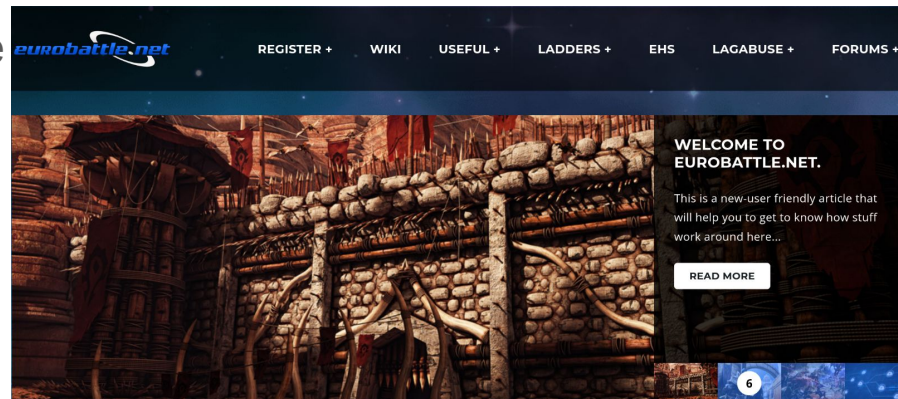
- Unofficial **Warcraft III** multiplayer server.
- Established as a Polish community in 2003, running on FreeBSD 5.
- Early years: Mostly Melee ladder and Custom Map games.
- After **2005** dominated by **DotA Allstars** (**Defense of the Ancients**) games.
- Primarily European player base and some smaller communities around the globe.



2003 - w3.xpam.de



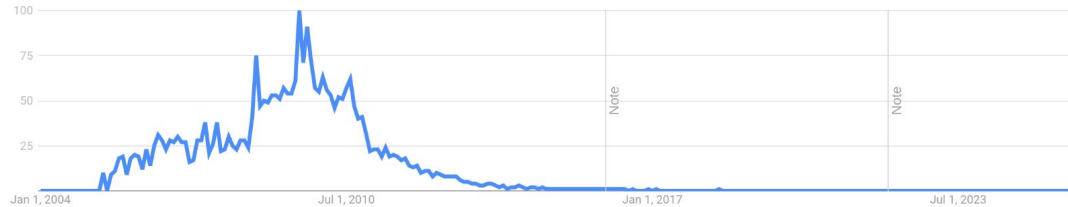
Forum logo
since 2006



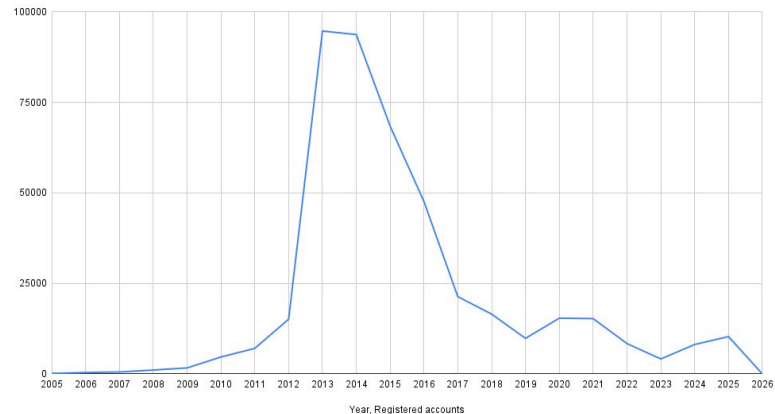
Current website

Google trends

Interest over time ?



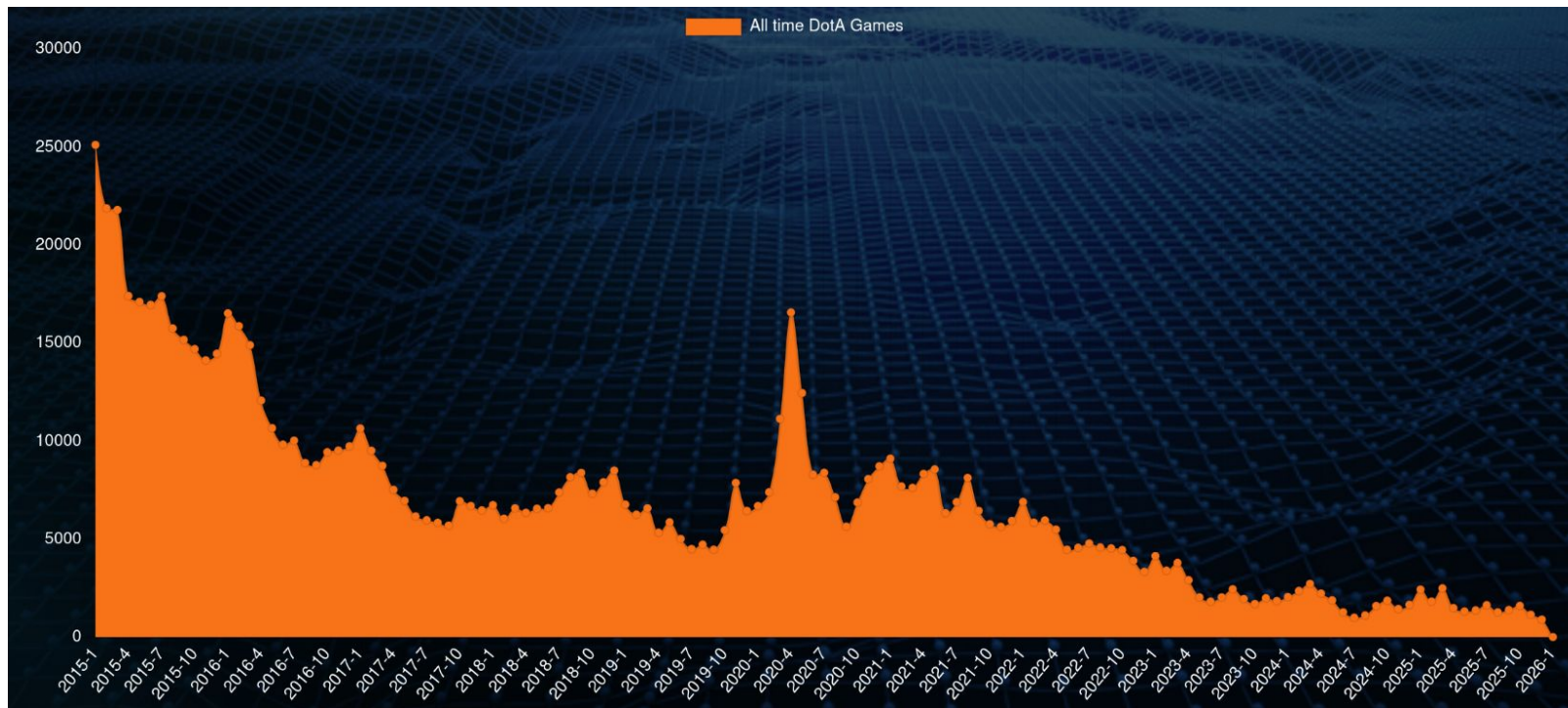
Account registrations over the years



Best years: 4000-5000 online users at peak hours (Europe TZ - evenings).

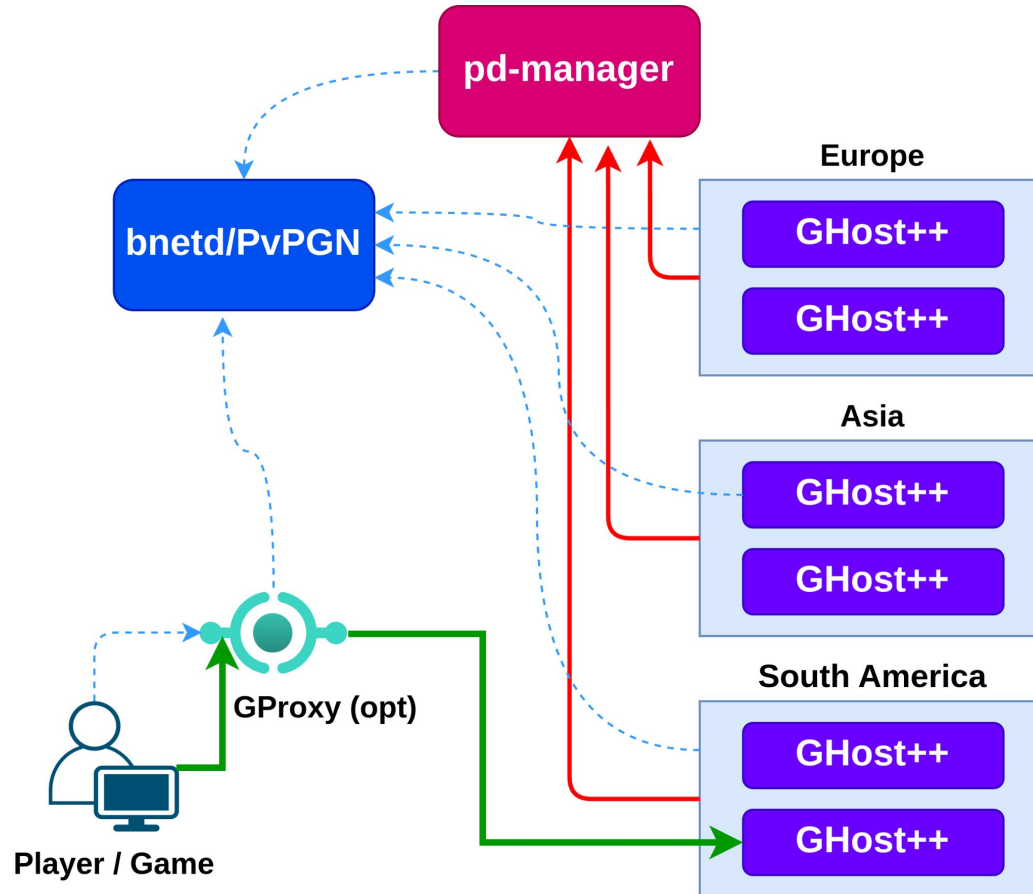
Today: 200-300

DotA League - dota.eurobattle.net, games per month

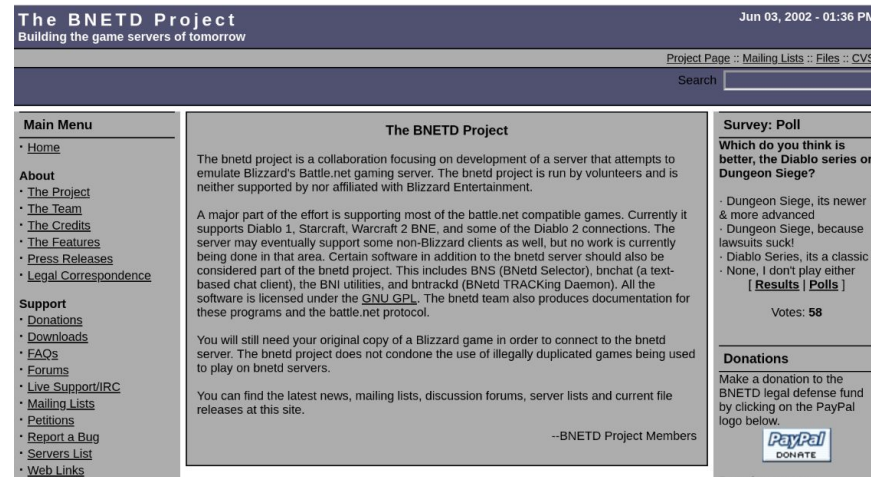


We lost data pre-2015 because someone accidentally did a thing

Tech stack



- Reverse engineered implementation of battle.net v1 server protocol.
- Released in 1998 with StarCraft support.
- Later Diablo I, II, Starcraft, Warcraft II, III.
- **The BNETD Project** sued and taken down by Blizzard (case: Blizzard v. Bnetd*)
- Forked as **PvPGN**: **P**layer **vs** **P**layer **G**aming **N**etwork.
- Added Westwood Online support (C&C games).
- GPLv2, C/C++ (lot's of C style C++).



BNETD Project website in 2002

* <https://slashdot.org/index2.pl?fhfilter=bnetd> ; Lawmeme archived article: <https://tinyurl.com/lawmeme-bnetd>

battle.net v1 protocol

- Unencrypted.
- Today: WireShark + bnetp filter
<https://github.com/diegonc/packet-bnetp>
- Community documentation:
<https://bnetdocs.org>
- **SID** sub-protocol: account, auth, chat, clans, game discovery.
- **W3GS** sub-protocol: lobby and gameplay traffic.

S>C 0x04 W3GS_SLOTINFOJOIN

Transport Layer: Transmission Control Protocol (TCP) Application Layer: Warcraft III In-Game Messages (W3GS)

Message Id: 0x04 Message Name: W3GS_SLOTINFOJOIN

Direction: Server to Client

Used By:
Warcraft III Reign of Chaos Warcraft III The Frozen Throne

Message Format:
(does not include protocol header)

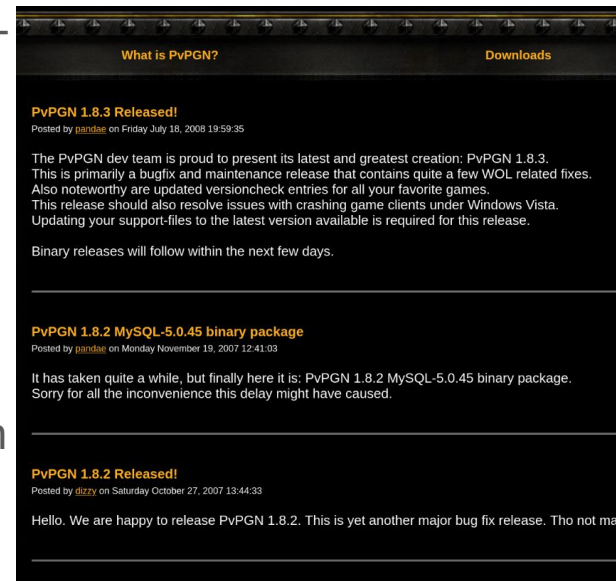
```
(UINT16) Length of Slot data
(UINT8) Number of slots
(UINT8) [] Slot data
(UINT32) Random seed
(UINT8) Game type
(UINT8) Number of player slots without observers
(UINT8) Player number
(UINT16) AF_INET (2)
(UINT16) Port
(UINT32) External IP
(UINT32) Unknown (0)
(UINT32) Unknown (0)

For each slot:
(UINT8) Player number
(UINT8) Download status
(UINT8) Slot status
(UINT8) Computer status
(UINT8) Team
(UINT8) Color
(UINT8) Race
(UINT8) Computer type
(UINT8) Handicap
```

Remarks
This is sent to tell the client about the game slots, upon entry of the lobby.

Example of a packet documentation

- PvPGN project hosting moved to BerliOS code platform - pvpkn.berlios.de
- Development stopped around 2011, final version 1.99-SVN available on GH: <https://github.com/d1zzy/pvpkn>
- In 2012 Forked as “**PvPGN Pro**”
<https://github.com/pvpkn/pvpkn-server>
- Code, CMake, JSON config, LUA plugin support.
- Development stopped in 2021. Probably the best version to host today but perhaps a bit less stable (some new bugs introduced, lack of unit tests).



Landing page on BerliOS

<https://github.com/eurobattle/pvpgn>

- Eurobattle uses 1.99-SVN with some additional patches.
- Command flooding prevention/rate limiting.
- Minor code fixes for latest compilers and C/C++ standards.
- Muted user allowlist: if a player is punished with a mute they are still allowed to whisper to certain system bot accounts.

Why was GHost++ needed?

Problems with traditional hosting on battle.net

- Players are game hosts - latency varies.
- Demand for competitive DotA.
- Game lists, player stats, ladders..
- “Username spoofing” problem - players using fake names in game lobbies.

What is GHost++?

- GHost (Game Host) ++ (C++)
 - Bot simulating a player that can create a game lobby and act as the game host.
 - Typically deployed on dedicated servers or a VPS.
 - Players interact through chat (!map <map name>, !pub <game name>, !ping...).
 - Can capture game events (Hero kills, winners..) and store them in a database.
 - Native support for DotA stats and a generic W3MMD standard.
-
- Solution to username spoofing: demand player to send /w <botname> sc, GHost can then verify with PvPGN in which game the “whisperer” is currently located (/where <whisperer>). Location mismatch == kick.

GHost++ development history

- Ghost Pwner project written in VB.NET by MrJag.
- Ported to C++ in 2008 by Trevor Hogan
- Apache 2.0 license.
- GhostOne fork with Windows GUI wrapper by Psionic One.
- Development via forum.codelain.com, later ghostpp.com (both now defunct).
- Google code SVN code repo, precompiled dependency dll downloads available via forum.

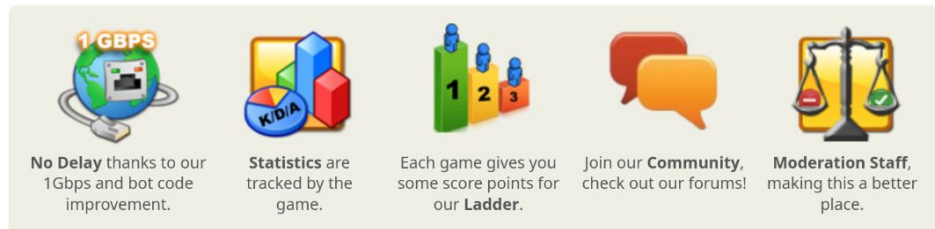
GHost [Owners:MrJag,Netrunner] - Dota Game Host BOT	
	GHost Development & Planning Direction and progression of GHost, please Do Not Post User Related Issue Here <i>Moderators: HearthCore, Varlock</i>
	GHost C++ Development & Planning <i>Moderators: HearthCore, Varlock</i>
	GHost Usage Problems In Battle.NET Post the problems you are having when running GHost on Battle.net <i>Moderators: HearthCore, Varlock</i>
	GHost Usage Problems In PVPGN Post the problems you are having when running GHost on PVPGN <i>Moderators: HearthCore, Varlock</i>
	GHost Resolved Problems [Read Only] Archive of previously resovled problems <i>Moderators: HearthCore, Varlock</i>

codelain forums ~2008

- **BNCSUtil** - <https://github.com/BNETDocs/bncsutil> C/C++ library implementing the logon sequence, co-maintained with BnetDocs.org
- **StormLib** - <https://github.com/ladislav-zezula/StormLib> C/C++ library to work with MPQ archives. Used to parse map metadata and create game lobby with correct number of slots, checksum etc.
- **Boost** (fs, asio, thread, date-time, regex), possible future work to replace with modern C++ alternatives.

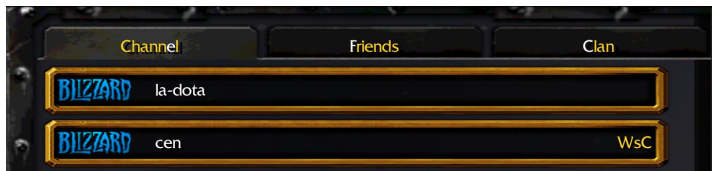
<https://github.com/cen1/pd-manager>

- Two GHost forks independently emerged within eurobattle community.
- pd-manager by playdota.eu DotA league (independent platform, later merged).



- **Problem:** a single GHost instance can only be in **one active lobby** at the same time.
- Once the game starts, a new lobby can be created but this is a bottleneck.

- **manager** - sits in a public channel, accepting player commands (!pub <gamename>)



- **ghost** - multiple instances connect to the manager via UDP protocol.
- manager delegates game creation to a free GHost instance.
- manages **state** (open lobbies, active games, stats collection, db storage).
- Multi-region support: player can request a GHost instance from certain location (Asia, South America, EU)
- PSR score, !balance(d) lobbies.
- Modern CMake, Conan v2, vcpkg support. Before: Makefile+Visual Studio 2008.sln
- Dockerfile/Compose support.

- The 2000s - age of dial up and early DSL.



- When a player loses connection mid-game, Warcraft III game client has a hard reconnection limit of **70 seconds**.

- “Game Proxy”, disconnection protection tool by Trevor Hogan, 2010.
 - Client side binary, sits between Warcraft III client and GHost++.
 - Additional GPS protocol layer between GProxy and GHost++.
 - For every real action, we send predetermined **N** empty actions.
-
- When player loses connection, if GProxy doesn't receive a W3GS action for 60 seconds it will feed one of the available empty actions to the game client and keep it interested.

```
cen: This is a reliable game. Requesting GProxy++ disconnect protection from server...
LA02 has joined the game.
LA02: =====
LA02:  ::dota.eurobattle.net :: region: eu ::
LA02: =====
```

<https://github.com/cen1/gproxy>

- Automatically spoof-checks on lobby join (/w <hostbot> sc)

```
LA02: Spoof check by whispering bot with "sc" [ /w lagabuse.com.02 sc ]  
You whisper to lagabuse.com.02: sc  
cen: GProxy++ disconnect protection is ready (360 second buffer).  
LA02: Spoof check accepted for [cen : server.eurobattle.net].
```

- Report W3 client version to GHost, enforce single version in lobby.

Some random stuff..

- Auto-login feature.
- Follow a friend into a lobby feature.
- Chat history (press UP/DOWN arrow keys to fill your chat box with your command history, terminal-like)

DEMO

