UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ACTIVISION BLIZZARD, INC.,
ELECTRONIC ARTS INC.,
TAKE-TWO INTERACTIVE SOFTWARE, INC.,
2K SPORTS, INC.,
ROCKSTAR GAMES, INC., and
BUNGIE, INC.,
Petitioner,
v.

ACCELERATION BAY, LLC,
Patent Owner.

Case IPR2015-01951[1]
Patent 6,714,966

DECLARATION OF VIRGIL E. BOURASSA IN SUPPORT OF PATENT OWNER'S RESPONSE

[1] Bungie, Inc., who filed a Petition in IPR2016-00935, has been joined as a petitioner in this proceeding.

I, Virgil E. Bourassa, declare as follows:

1. I am over the age of majority and make this declaration of my own personal knowledge.

2. Between October, 1996 and May, 2001, I worked in Boeing's Mathematics and Computing Technology research division as a computing technologist. As a computing technologist, I spent my time augmenting [redacted]

3. [redacted] Boeing used [redacted] in order to [redacted] In November 1996, Robert Abarbanel, the head of [redacted] Research, asked me to start developing a communications library for [redacted]. We referred to the communications library as SWAN, which resulted in inventions covered by U.S. Patent Nos. 6,829,634 ("the '634 Patent"), 6,701,344 ("the '344 Patent"), 6,920,497 ("the '497 Patent"), 6,910,069 ("the '069 Patent"), 6,732,147 ("the '147 Patent"), and 6,714,966 ("the '966 Patent") (collectively, the "SWAN Patents"). I am a co-inventor of the SWAN Patents.

4. At around this time, Boeing was expanding beyond the Puget Sound region as it acquired a series of airplane companies. For these reasons, we wanted to augment [redacted] to allow for peer-to-peer communications for virtual communications so that review sessions could be shared across the world, to allow major portions, such as the database query used for selecting parts, to be broken

out, and to allow new modules to be introduced without a major software overhaul. Boeing had a procedure that allowed for two running [redacted] processes to work in tandem. However, as soon as a third [redacted] process was introduced, it broke down.

5. As Fred and I described in our Invention Disclosure Form, the typical computer network communications techniques included: (1) fully-connected point-to-point network protocols, (2) client/server middleware, (3) multicasting network protocols, and (4) peer-to-peer middleware.

6. Fully-connected point-to-point networking protocols were disadvantageous because fully-connected network graphs do not scale as the number of participating processes grows. Client/server middleware systems were disadvantageous because the server is a performance bottleneck and a single point of failure. Multicast networking protocols were disadvantageous because multicast traffic at the time was limited to single local-area networks. Peer-to-peer middleware at the time was not suitable for the needs of medium to large-scale collaboration.

7. The lack of technology available that could provide peer-to-peer communications among computer processes across the world with high reliability and low latency reaffirmed that a solution was needed in this area to satisfy an existing need in the market. Because I had written a reliable UDP multicast

communications library suitable for local-area networks for another project unrelated to [redacted] Robert Abarbanel, my supervisor and the head of [redacted] research at the time, asked me to come up with a way to create something similar for the wide-area networking required to [redacted] across multiple [redacted] sessions across the country.

8. I originally anticipated this to be a simple task and expected that I would have a working model within three weeks. This estimate was far off from what I was expecting—instead, it took me about three years and about 28 epiphanies to incorporate into [redacted] what would eventually be referred to as SWAN.

9. I knew that I did not want to use a tree per se because that would put the root node in a privileged position. I considered breaking the nodes into two trees that would be connected between the pair of roots and amongst the leaf nodes. In around December of 1996, I began working with Fred Holt, a colleague at Boeing with a Ph.D. in Mathematics, and told him about my proposal. The idea worked well for a pair of ternary trees up to a depth of three. However, Fred determined that beyond this level, there was insufficient connectivity among the leaves to share the information between the trees any more quickly than sending the shared message to the other through the roots. In other words, at this size and

above, some of the leaf nodes would take at least twice as long to share a message as the root nodes.

10. Fred and I first tested a 3-regular (meaning 3 connections per node) and 3-connected (meaning at least 3 nodes would have to be lost to break the graph apart) graph, but it only worked well while the number of nodes in the graph was even. If the number of nodes were odd, it would create a half-edge problem, meaning that it was necessary to keep track of one special node. Fred and I discovered that if we made a four-regular, four-connected graph, the half edge problem in the 3-regular, 3-connected graph was eliminated.

11. We then realized we could solve the half-edge issue by creating a 4-regular, 4-connected graph that had a low diameter. The diameter was our figure of merit for the latency of the graph. We gravitated towards developing an inductive algorithm to build our 4-regular, 4-connected graphs. We would have a known node in the graph to be the point of entry, then using an inductive approach, have new nodes added in the vicinity of this entry point in such a way as to maintain 4-regularity and connectedness. However, when we tested at 20 nodes, we had a diameter of four, which seemed high to me. To confirm this, I ramped up the algorithm to build a graph of 2000 nodes, and got a diameter of 179. Since a balanced binary tree with 2000 nodes would only have a diameter of 20, we clearly had done something wrong.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.