UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

MASTEROBJECTS, INC.,

Plaintiff,

v.

AMAZON.COM, INC,

Defendant.

No. C 20-08103 WHA

ORDER RE SUMMARY JUDGMENT

INTRODUCTION

In this patent infringement action, defendant moves for summary judgment of noninfringement. To the following extent, the motion is **GRANTED**.

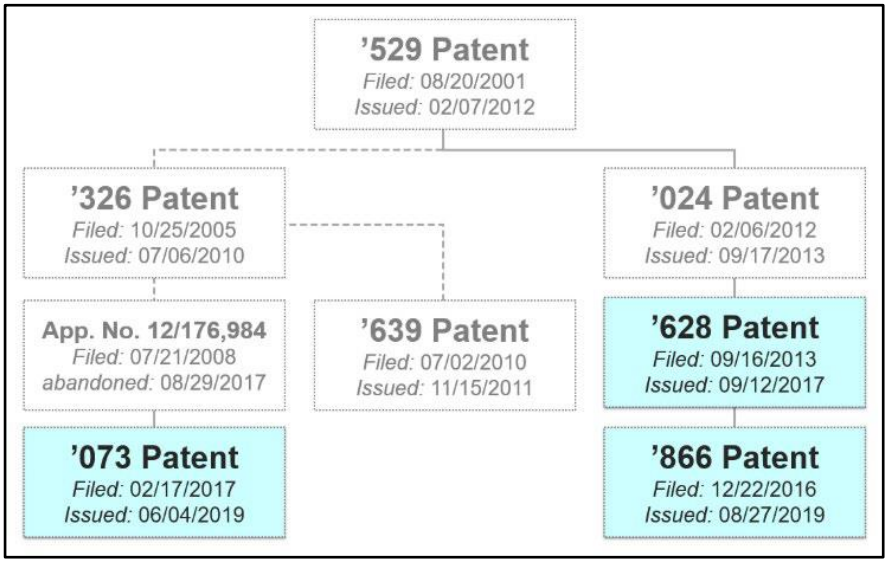
STATEMENT

This litigation concerns autocomplete technology for digital searches. Autocompletion suggests ways for the user to complete her search as she actively types it into a search bar. Patent owner MasterObjects, Inc. accuses alleged infringer Amazon.com, Inc. of infringing three of its patents in this area, U.S. Patent Nos. 9,760,628; 10,311,073; and 10,394,866.

Our parties have some history. MasterObjects first filed a patent-infringement lawsuit against Amazon in 2011, but the parties stipulated to dismissal without prejudice less than six months later. *MasterObjects, Inc. v. Amazon.com, Inc.*, No. C 11-01055 CRB (N.D. Cal. filed Mar. 7, 2011) (Judge Charles R. Breyer). MasterObjects targeted Amazon once again in May

2020, when it initiated the instant lawsuit in the United States District Court for the Southern District of New York. In October 2020, Judge P. Kevin Castel transferred the suit to our district (Dkt. No. 82). MasterObjects originally alleged infringement of four patents, but one was later withdrawn.

The three patents still in suit all descend from U.S. Patent No. 8,112,529 (filed in 2001). The following diagram lays out the patent genealogy. The asserted patents are highlighted in blue, full lines demarcate continuations of the earlier patent, and dashed lines represents continuations-in-part:



The '628 and '866 patents are continuations of the original '529 patent and share its specification. The '073 patent is a continuation-in-part of the '529 patent and incorporates its specification by reference.

Of the fifteen claims asserted, there are three independent claims from which the rest of the claims-in-suit depend: claim 13 of the '628 patent; claim 1 of the '073 patent; and claim 1 of the '866 patent. To frame the analysis that follows, here is claim 13 of the '628 patent in its entirety (emphasis added):

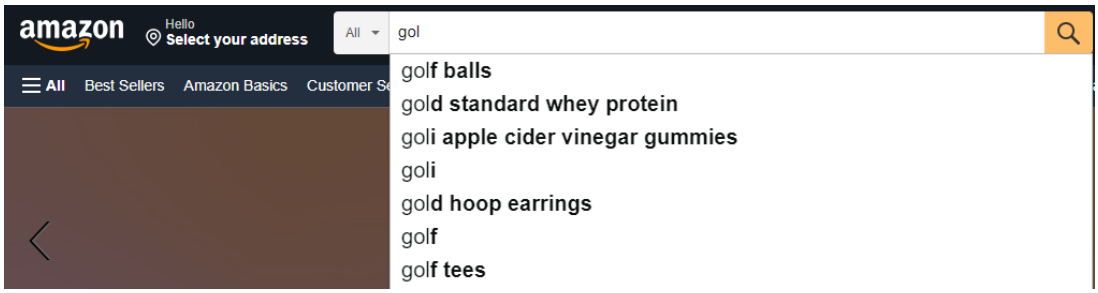
> **13[a]** A system comprising:
>
> **13[b]** a server system, including one or more computers, which is configured to receive query messages from a client object, the server system receiving and asynchronously responding to the query messages from the client object over a network;

> **13[c]** the client object that, while a user is providing input comprising a lengthening string of characters, sends query messages to the server system;
>
> **13[d]** Whereby the query messages represent the lengthening string as additional characters are being input by the user; and
>
> **13[e]** wherein the server system, while receiving said query messages, uses the input to query data available to the server system and send return messages to the client object containing results in response to the input;
>
> **13[f]** *wherein the server system caches query results and subsequently determines results by looking up the query in said cache so that it can avoid performing a query for the same input on a data source or looking up said query in a second cache.*

Per the common specification, "[a]s a user inputs data into a field on a form, the auto-complete function analyzes the developing character string and makes intelligent suggestions about the intended data being provided. These suggestions change dynamically as the user types additional characters in the string" ('628 patent 6:44–48). All agree that the claimed system generates autocomplete results from a specific server system as opposed to a broader search of the internet, generally.

Of principal concern here, the key limitation in the claim recited above is directed to the system's use of a "cache" to provide autocomplete results. "Cache" is used as both noun and verb. All the other claims-in-suit similarly reference a cache as well as a "data source" (or "content sources"). There is no dispute that "content sources," like a "data source," generally provide data to the claimed system. The lay reader will likely have a general awareness of a cache in the computing sense, such as a web browser (*e.g.*, Mozilla Firefox) storing an iteration of a previously visited webpage (*e.g.*, cand.uscourts.gov). This order addresses the use of the term "cache" by the patents-in-suit.

Before going further into the claims, this order will get into the nuts and bolts of the accused system. Let's start with a practical example of Amazon's autocomplete as a user would find it on the Amazon.com homepage:



A user here has typed "gol" into Amazon's search bar (an autocomplete query), and Amazon's system has suggested several autocomplete query results, *i.e.*, suggestions for a complete query for products such as "golf balls," "gold standard whey protein," etc. The autocomplete query results adjust as the user continues typing and revises her autocomplete query.

Amazon's autocomplete system uses a freestanding, constructed set of databases built by a process Amazon calls the "daily build." The "daily build" process occurs every one or two days where it affirmatively builds two paired databases called read-only databases (RODBs). The RODB databases are the *only* part of Amazon's system that can process an autocomplete query, an important point for noninfringement. These RODBs are, by definition, "read only" — data can neither be added nor removed from the databases during their one-to-two-day lifespan. The paired RODBs work together and store data in key-value pairs, as follows. The *first* RODB, the "Prefix RODB," includes queries (*i.e.*, prefixes) a user might type (*e.g.*, "gold," "golde," "golden"), paired with information reflecting rows in the *second* RODB. The *second* RODB, the "Keyword RODB," stores the autocomplete suggestions (*i.e.*, keywords) that correspond to the user's query/prefix:

Prefix RODB		Keyword RODB	
Key	Value	Key	Value
gold	[Index 1, Index 3]	Index 1	gold earrings
golde	[Index 2, Index 3]	Index 2	golden girls
golden	[Index 2, Index 3]	Index 3	golden gate bridge

While a user types, the autocomplete system finds a matching prefix in the Prefix RODB, if any, then follows the index numbers to the Keyword RODB, which the system then pushes to

the user (Turnbull Noninfringement Rep. ¶¶ 103, 106–12, Dkt. Nos. 380-12, 384-5; Miller Decl. Exh. F, Dkt. No. 380-14).

The key point, however, is how the columns are created in the first place. The RODBs are created from the "Tommy Query Groups." Tommy is the name for the Amazon search dataset, which includes information on searches, clicks, purchases, etc. The Tommy Query Groups are aggregated Tommy search-analytics data related to completed product searches. Tommy Query Groups contain completed search strings from product search requests and metadata associated with each search request. This bears emphasizing because the Tommy dataset does not store autocomplete queries and at most stores autocomplete results (and arguably not even that). Rather, the Tommy dataset stores completed *product search queries*, *i.e.*, not a query where the results are suggestions for a complete query (*e.g.*, "Rol" = "Rolex"), but a query where the corresponding results are listings for actual products Amazon has for sale (*e.g.*, the range of Rolex watches a customer could buy).

The "daily build" process uses data from the Tommy Query Groups to populate the Keyword RODB (the results column). During this process, among other things, spelling-corrected suggestions are generated and a blocklist is applied to remove certain entries. Once the "daily build" collects all the keywords in the Keyword RODB, it generates prefixes for each using a rote, progressive letter construction: for a keyword such as, *e.g.*, "Rolex," it would generate the prefixes of "R," "Ro," "Rol," "Role," and "Rolex." Amazon also uses a "Personalization Platform Data Service" (PPDS), which includes the individual user's recent completed *product search queries*, which are used to re-rank suggestions to more closely match those prior queries (*see* Turnbull Noninfringement Rep. ¶¶ 109–12, 171, 181 n.21, 244; Peck Rep. ¶¶ 140–41, Dkt. Nos. 380-6, 384-7, 389-2).

Having completed this high-level overview of the patents-in-suit and the accused system, this order proceeds to analyze the specifics. This order follows full briefing and oral argument.

ANALYSIS

A patent infringement analysis involves two steps. The claim must be properly construed to determine its scope and meaning. Claim terms generally take "their ordinary and customary

DOCKET ALARM

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.