

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

UNILOC 2017 LLC,
Patent Owner.

Case IPR2020-00023
Patent 6,467,088 B1

Before SALLY C. MEDLEY, MIRIAM L. QUINN, and
SCOTT RAEVSKY, *Administrative Patent Judges*.

QUINN, *Administrative Patent Judge*.

JUDGMENT
Final Written Decision
Determining No Challenged Claims Unpatentable
35 U.S.C. § 318(a)

We instituted *inter partes* review pursuant to 35 U.S.C. § 314 to review claims 1–4, 6–14, and 16–21 of U.S. Patent No. 6,467,088 B1 (“the ’088 patent”), owned by Uniloc 2017 LLC. We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision is entered pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73. For the reasons discussed below, Petitioner has not shown by a preponderance of the evidence that claims 1–4, 6–14, and 16–21 of the ’088 patent are unpatentable.

I. BACKGROUND

A. Related Matters

The parties identify the following district court proceedings involving the ’088 patent: *Uniloc USA, Inc. and UNILOC Luxembourg, S.A. v. Apple Inc.*, No. 1:18-cv-00296 (W.D. Tex.), filed April 9, 2018; *Uniloc 2017 LLC v. Microsoft Corporation*, No. 8:19-cv-00956 (C.D. Cal.), filed May 20, 2019; and *Uniloc USA, Inc. and Apple Inc.*, No. 6:19-cv-00532 (W.D. Tex.), filed September 10, 2019. Pet. x; Prelim. Resp. 11–12; Paper 4, 2.

In addition to this Petition, the ’088 patent was also challenged by a different party, Apple Inc., in IPR2019-00056 (“the Apple IPR”), which we denied.

B. The ’088 Patent

The ’088 patent is directed to techniques for upgrading or reconfiguring software and/or hardware components in electronic devices. Ex. 1001, 1:6–9. The ’088 patent explains that prior art systems developed for updating components of electronic devices rely on a central computer

system that tracks all software configurations for a number of remote systems. *Id.* at 1:31–36. These prior art systems updated software by the central computer transmitting patches to each of the remote systems. *Id.* at 1:39–42; *see also id.* at 2:4–10 (explaining that a distributed system transmits patches to mobile units). Other known techniques for software update involve assuming that each desktop computer has a set of resources determined in accordance with a set of enterprise policies or a central server maintaining a master list that is used to keep files on a remote device updated to the latest version. *Id.* at 1:49–52, 1:60–65. According to the '088 patent, all of the above techniques fail to avoid potential conflicts and ensure compatibility because they do not account for interdependencies of the resources required by the desktops or the files resident in the remote devices. *Id.* at 1:41–45, 1:52–56, 1:65–2:3, 2:10–14.

The '088 patent solves the problem by providing a list or listing, that indicates “which of a set of software components supported by the manager 10 are known to work well together or are otherwise compatible.” *Id.* at 3:36–42. For instance, Figure 1 of the '088 patent, reproduced below, illustrates reconfiguration manager 10 that “includes a listing 16 of known configurations, and a repository 18 of software components.” *Id.* at 3:27–29.

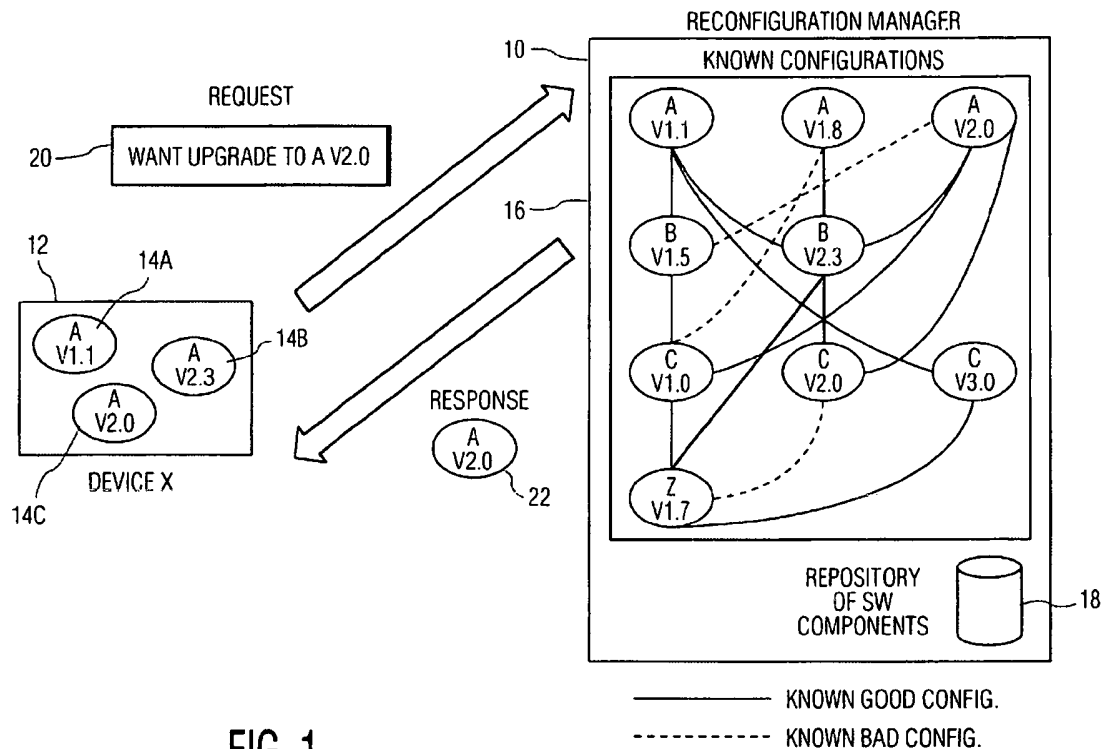


FIG. 1

Figure 1, above, illustrates a reconfiguration manager 10 interacting with an electronic device 12, also referred to as "Device X." *Id.* at 3:14–16. When reconfiguration manager 10 receives a request for an upgrade from Device X, the request indicates that the device wants to upgrade to version 2.0 of software component A and includes a list of the components currently on the device, i.e., version 1.1 of component A, version 2.0 of component C, and version 2.3 of component B. *Id.* at 4:12–19. Reconfiguration manager 10 processes the request, and if appropriate, delivers the requested version 2.0 of software component A. *Id.* at 4:22–26. Processing the request involves generating a potential upgrade configuration that will satisfy the received request, and searching through a set of known "bad" configurations. *Id.* at 4:62–66. A known "bad" configuration is indicated in

Figure 1 as a dashed line between components that are not compatible. *Id.* at 3:58–61. For example, “[t]he pair including version 1.8 of component A and version 1.0 of component C is an example of a known bad configuration.” *Id.* at 3:61–63.

If the upgrade configuration corresponds to a bad configuration, the reconfiguration manager “attempts to find a set or sets of potential upgrade configurations from a set of known good configurations.” *Id.* at 4:67–5:3. A known “good” configuration is indicated in Figure 1 by a solid line between a given pair of components indicating that the components work well together or are otherwise compatible. *Id.* at 3:52–55.

C. Illustrative Claim

Petitioner challenges claims 1–4, 6–14, and 16–21 of the ’088 patent. Pet. 2. The ’088 patent recites three independent claims: 1, 11, and 21. Challenged claim 1, reproduced below, is illustrative of the recited subject matter:

1. A processor-implemented method for controlling the reconfiguration of an electronic device, the method comprising the steps of:
 - receiving information representative of a reconfiguration request relating to the electronic device;
 - determining at least one device component required to implement the reconfiguration request;
 - comparing the determined component and information specifying at least one additional component currently implemented in the electronic device with at least one of a list of known acceptable configurations for the electronic device and a list of known unacceptable configurations for the electronic device; and

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.