UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

MICROSOFT CORPORATION,
Petitioner,

v.

UNILOC 2017 LLC,
Patent Owner.

_____

IPR2020-00023
Patent 6,467,088 B1

_____

Before MIRIAM L. QUINN, AMANDA F. WIEKER and
SCOTT RAEVSKY, *Administrative Patent Judges.*

RAEVSKY, *Administrative Patent Judge*.

JUDGMENT
Final Written Decision on Remand
Determining All Challenged Claims Unpatentable
35 U.S.C. §§ 114, 318

I. INTRODUCTION

This case is on remand from the United States Court of Appeals for the Federal Circuit to address the patentability of 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 ("Patent Owner"). *Microsoft Corp. v. Uniloc 2017 LLC*, No. 2021-2039 (Fed. Cir. Oct. 20, 2022) (nonprecedential).

For the reasons discussed below, we determine that Petitioner has proven by a preponderance of the evidence that claims 1–4, 6–14, and 16–21 of the '088 patent are unpatentable.

II. BACKGROUND

A. *The '088 Patent and Illustrative Claim*

The '088 patent is directed to techniques for upgrading or reconfiguring software and/or hardware components in electronic devices. Ex. 1001, 1:6–9. According to the '088 patent, prior art software update techniques fail to avoid potential conflicts and thus 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 this problem by providing a list or listing that indicates "which of a set of software components supported by [a reconfiguration] 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.
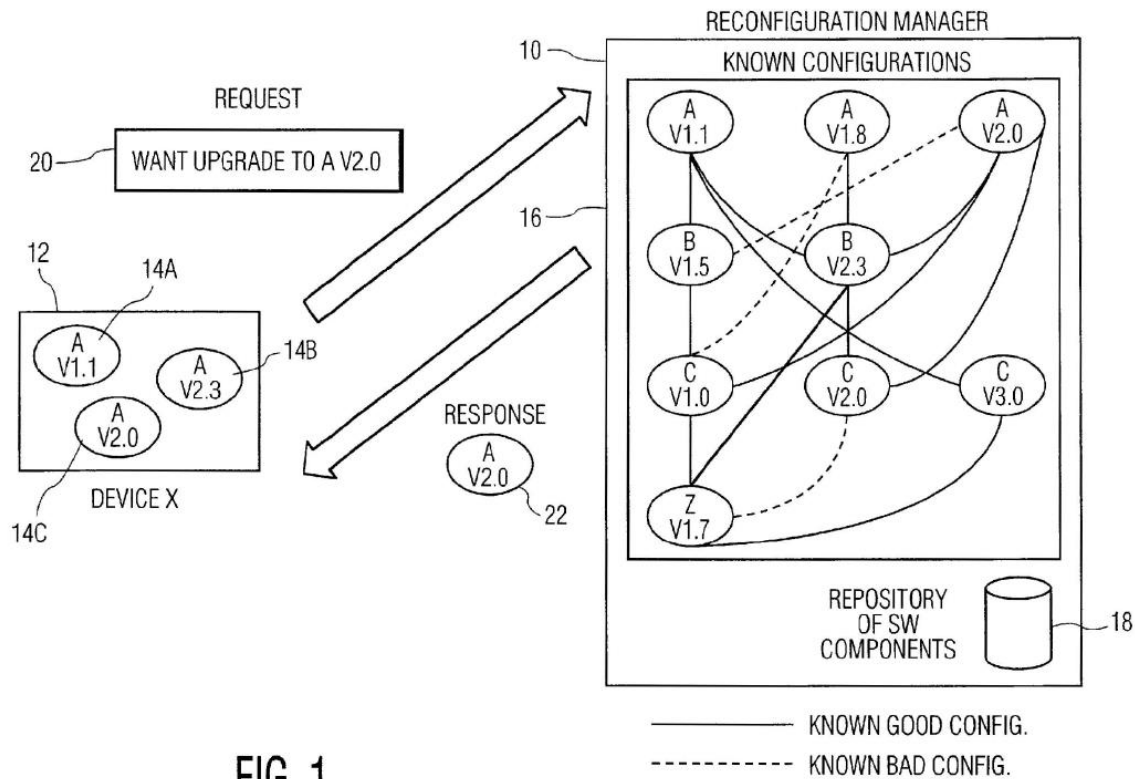

FIG. 1

Figure 1 illustrates reconfiguration manager 10 interacting with 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.[1]
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.

Challenged claim 1, reproduced below, is illustrative of the
challenged claims:

---

[1] Although Fig. 1 depicts device 12 having three versions of software
component A (labels 14A–C), the description of Fig. 1 indicates these are
different versions of components A, B, and C, respectively. Ex. 1001, 3:20–
24.

> 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
>
>   generating information indicative of an approval or a denial of the reconfiguration request based at least in part on the result of the comparing step.

Ex. 1001, 6:43–59. We refer to the steps of claim 1 as the receiving step, the determining step, the comparing step, and the generating step, respectively.

B. Trial Background

Microsoft Corporation ("Petitioner") filed a Petition (Paper 2, "Pet.") requesting *inter partes* review of claims 1–4, 6–14, and 16–21 of the '088 patent. The Petition asserts that the claims are unpatentable on the following grounds (Pet. 3–4), which are supported by a declaration by John Villasenor (Ex. 1003):

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.