UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
13/285,101	10/31/2011	Joerg Schulze	DE920110056US1	4184

40412 7590 02/27/2018
IBM CORPORATION- AUSTIN (JVL)
C/O LESLIE A. VAN LEEUWEN
6123 PEBBLE GARDEN CT.
AUSTIN, TX 78739

EXAMINER	
VU, TUAN A	
ART UNIT	**PAPER NUMBER**
2193	
NOTIFICATION DATE	**DELIVERY MODE**
02/27/2018	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

leslie@vl-patents.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte JOERG SCHULZE

Appeal 2017-001316
Application 13/285,101
Technology Center 2100

Before JEAN R. HOMERE, BRADLEY W. BAUMEISTER,
and SHARON FENICK, *Administrative Patent Judges*.

HOMERE, *Administrative Patent Judge*.

DECISION ON APPEAL

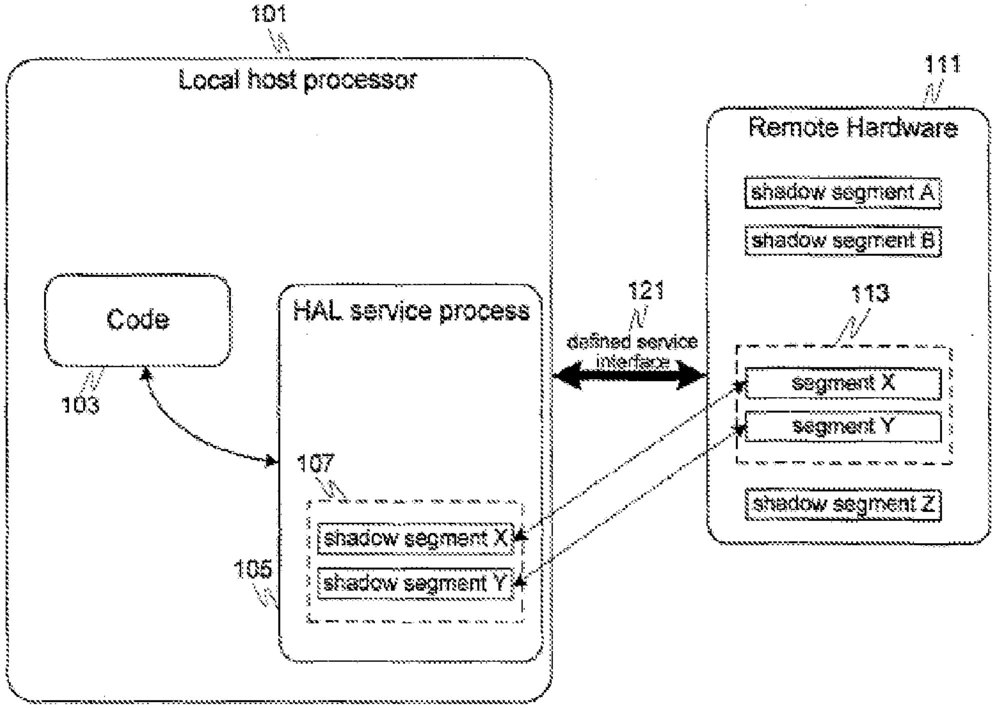
Appellant appeals under 35 U.S.C. § 134(a) from a Final Rejection of claims 1–22, which constitute all claims pending in this application.[1] Claims App'x. We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

[1] Appellant identifies the real party in interest as International Business Machines Corp. App. Br. 3.

Introduction

According to Appellant, the claimed subject matter relates to a shadow mechanism for initiating, at a remote processor, a break in a program code (103) running on a local processor (101). Spec. ¶ 4, Fig. 1. In particular, breakpoints inserted in the program code (103) are stored at the process space of the remote processor (111) as shadow hardware segments (113), whereas corresponding replica thereof are stored as shadow segments (107) at the process space of the local processor. *Id.* Upon detecting a break event data in the shadow segments (107), the local processor (101) halts execution of the local code at the remote processor (111) for subsequent debugging. *Id.*



Representative Claim

Independent claim 1 is representative, and reads as follows:

> 1. A method of initiating a break in program code running in process space of a local processor, the method comprising:
>
> running the program code in the process space of the local processor;
>
> providing access to remote hardware comprising a remote processor coupled to one or more remote hardware memory segments, wherein the program code running in the process space of the local processor uses a defined service interface to access to the remote hardware;
>
> accessing the remote processor coupled to the one or more remote hardware memory segments with the program code, wherein the program code includes a predefined break event for said remote processor couple to the one or more remote hardware memory segments;
>
> providing a shadow mechanism in said process space of the local processor with one or more shadow segments respectively corresponding to each of the one or more remote hardware memory segements to be accessed by the program code, wherein a corresponding copy of data in the one or more remote hardware memory segments is maintained by the shadow mechanism in said process space of the local processor;
>
> monitoring the shadow mechanism for the break event, wherein the monitoring is performed by the local processor executing the program code;
>
> detecting occurrence of the break event indicated in said one or more shadow segments corresponding to an occurrence of the break event in the one or more remote hardware memory segments; and

> halting execution of the program code in response to the detecting of the break event as incicated in said one or more shadow segments.

Prior Art References

Hotley et al.	US 4,257,101	Mar. 17, 1981
Phillips et al.	US 5,321,828	June 14, 1994
Coker	US 5,371,878	Dec. 6, 1994
Kau et al.	US 5,684,997	Nov. 4, 1997
Barry (“Barry2”)	US 2001/0032305 A1	Oct. 18, 2001
Barry et al. (“Barry”)	US 2001/0049763 A1	Dec. 6, 2001
Long et al.	US 8,230,149 B1	July 24, 2012
Han et al.	US 8,319,521 B1	Nov. 27, 2012

Rejections on Appeal

Claims 1–5, 9–13, and 17–22 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Barry, Barry2, Phillips and/or Coker and/or Han, Long, and Hotley. Final Act. 2–17.

Claims 6–8 and 14–16 stand rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Barry, Barry2, Phillips and/or Coker and/or Han, Long, Hotley, and Kau. Final Act. 17–20.

ANALYSIS

Appellant argues that the combination asserted by the Examiner does not teach or suggest the following language of independent claim 1:

> providing a *shadow mechanism in said process space of the local processor with one or more shadow segments* respectively corresponding to each of the *one or more remote hardware memory segments* to be accessed by the program code, wherein a

> corresponding copy of data in the *one or more remote hardware memory segments* is maintained by the shadow mechanism in said process space of the local processor.

App. Br. 9–10 (emphasis added).

In particular, Appellant argues that Barry's shadow instruction registers are shadow locations for access by a co-processor or a host processor depending on the access mode (e.g., interrupt vector fetch mode or instruction fetch mode). *Id.* (citing Barry ¶ 45). According to Appellant, Barry's disclosure of the shadow locations does not teach local memory locations corresponding to remote memory locations. *Id.* at 10. Further, Appellant argues Coker's shadow system does not operate in the process of the local processor, as required by the claim. *Id.* at 13.

Appellant's arguments are persuasive. The disputed claim limitations, as emphasized above, require a shadow mechanism in the process space of the local processor with a shadow segment corresponding to a copy of data in a remote hardware memory segment maintained by the shadow mechanism. We agree with the Examiner that Coker discloses respective memory correspondence between a target system and a host system, wherein the target system includes copy of data stored in the host system, and teaches a shadow segment in the process space of the local processor corresponding to a shadow segment in process space of a remote processor. Ans. 8 (citing Coker, Fig. 1). However, we agree with Appellant that Coker's shadow system (28) does not operate in the process space of the local processor. Instead, Coker's shadow system (28) is outside of the local processor (target system). Cocker, Fig. 1.

Because Appellant has shown at least one reversible error in the Examiner's rejection, we do not sustain the Examiner's rejection of claim 1. Accordingly, we reverse the obviousness rejection of claim 1, as well as the obviousness rejection of claims 2–5, 9–13, and 17–22, which also suffer the deficiencies noted above.

Regarding the obviousness rejection of claims 6–8 and 14–16, the Examiner does not rely on the additional teachings of Kau to cure the deficiency of the obviousness rejection noted above. Final Act. 17–20. Accordingly, we likewise do not sustain the obviousness rejection of these claims for the reasons set forth above.

DECISION

For the above reasons, we reverse the Examiner's obviousness rejection of claims 1–22.

REVERSED