



# 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.
15/095,230	04/11/2016	Shruthi Mittal	90035363	7039
146568	7590	09/21/2020	EXAMINER	
MICRO FOCUS LLC 500 Westover Drive #12603 Sanford, NC 27330			BOURZIK, BRAHIM	
			ART UNIT	PAPER NUMBER
			2191	
			NOTIFICATION DATE	DELIVERY MODE
			09/21/2020	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):

software.ip.mail@microfocus.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

*Ex parte* SHRUTHI MITTAL and ARUN PAI

---

Appeal 2019-003107  
Application 15/095,230  
Technology Center 2100

---

Before JASON V. MORGAN, DEBORAH KATZ, and JOHN A. EVANS,  
*Administrative Patent Judges.*

MORGAN, *Administrative Patent Judge.*

DECISION ON APPEAL  
STATEMENT OF THE CASE

Introduction

Pursuant to 35 U.S.C. § 134(a), Appellant<sup>1</sup> appeals from the Examiner's decision to reject claims 1, 3–10, and 12–14. Claims 2, 11, and 15 are canceled. Final Act. 2. We have jurisdiction under 35 U.S.C. § 6(b).

We AFFIRM.

---

<sup>1</sup> We use the word “Appellant” to refer to “applicant” as defined in 37 C.F.R. § 1.42. Appellant identifies the real party in interest as EntIT Software LLC (Appeal Br. 3), which has since been renamed to MICRO FOCUS LLC (Pat. Assign. Reel 050004, Frames 0001, 0028, 0041–42, 0106).

### Summary of the Disclosure

Appellant's claimed subject matter relates to generating a service design specifying operational requirements for deploying an application on an application platform. Abstract.

### Representative Claims

(Disputed Limitations Emphasized and Bracketing Added)

1. A method for deploying applications on application platforms, the method comprising:

receiving an input application design of an application, wherein the input application design defines dependencies for the application for components of an application platform based on capability attributes and characteristic attributes corresponding to the application, the capability attributes providing deployment requirements of the application and the characteristic attributes providing specifications of the capability attributes corresponding to the components of the application platform;

identifying, based on the capability attributes and characteristic attributes, at least one candidate design comprising at least one of the components of the application platform, the at least one candidate design being a configuration of the application platform to deploy the application;

based on the at least one candidate design, generating a service design specifying operational requirements for deploying the application on the application platform,

wherein identifying the at least one candidate design comprises:

[1] *examining components of each of a plurality of designs of the application platform to identify eligible designs having components which conform with the deployment requirements of the application;*

[2] *ascertaining deployment requirements of the components of the each of the eligible designs; and*

identifying an eligible design as a candidate design when the deployment requirements of the components of the eligible design are satisfied by one or more components of the application platform.

6. The method as claimed in claim 1, [3] *wherein identifying the at least one candidate design is based on role-based access rules associated with a user deploying the application.*

7. The method as claimed in claim 1 further comprising:

[4] *creating a checksum for the service design based on a unique identifier associated with the service design;*

[5] *comparing the checksum of the service design with a checksum of another application to be deployed, the checksum of the another application being based on a unique identifier associated with the another application and revision and version of application components of the another application;*  
and

reusing the service design for deploying the another application based on the comparison.

The Examiner's rejections and cited references

The Examiner rejects claims 1, 8–10, and 14 under 35 U.S.C. § 102(a)(1) as being anticipated by Agrawal et al. (US 2008/0021873 A1; published Jan. 24, 2008) (“Agrawal”). Final Act. 5–15.

The Examiner rejects claims 3–6 and 12 under 35 U.S.C. § 103 as being unpatentable over Agrawal and Narayanaswamy et al. (US 7,069,553 B2; issued June 27, 2006) (“Narayanaswamy”). Final Act. 15–21.

The Examiner rejects claims 7 and 13 under 35 U.S.C. § 103 as being unpatentable over Agrawal, Maes et al. (US 2015/0100684 A1; published Apr. 9, 2015) (“Maes”), and Spivak et al. (US 2012/0266156 A1; published Oct. 18, 2012) (“Spivak”). Final Act. 21–25.

## ADOPTION OF EXAMINER’S FINDINGS AND CONCLUSIONS

We agree with and adopt as our own the Examiner’s findings as set forth in the Answer and in the Action from which this appeal was taken, and we concur with the Examiner’s conclusions. We have considered Appellant’s arguments, but do not find them persuasive of error. We provide the following explanation for emphasis.

## ANALYSIS

Claims 1, 3–5, 8–10, and 14

In rejecting claim 1 as anticipated, the Examiner finds that Agrawal’s disclosed search for candidate infrastructure elements to satisfy at least one requirement attribute and employment of criteria to determine the best candidate infrastructure elements discloses [1] “examining components of each of a plurality of designs” and [2] “ascertaining deployment requirements of the components of the each of the eligible designs.” Final Act. 8–9 (citing Agrawal ¶¶ 48, 49); Ans. 23–26 (further citing Agrawal ¶ 27, Fig. 6).

Appellant argues “Agrawal discusses a single design, not examining components of a plurality of designs or candidate designs.” *Id.* at 11; Reply Br. 1–2. Appellant’s argument is unpersuasive because Agrawal discloses “determining a *list* of [the] best candidate infrastructure elements for each node.” Agrawal ¶ 48 (emphasis added). Then “[t]he candidate infrastructure elements are ranked to determine *an* infrastructure element that best satisfies the at least one requirement attribute.” *Id.* ¶ 49 (emphasis added). Thus, the list of candidate elements each represent at least one of a plurality of designs

(i.e., each design either including or excluding a particular candidate infrastructure element).

Appellant further contends the Examiner erred because “Agrawal discusses determining the infrastructure elements for individual nodes, i.e., individual parts of the distributed application.” Appeal Br. 10. Claim 1 does not, however, require examining *all* of the components of each of a plurality of designs. Rather, claim 1—in merely reciting “examining components”—encompasses examining a *subset* of such components. Agrawal’s list of candidate infrastructure elements, even if considered only for an individual node that is part of a distributed application, still represents components of a plurality of designs for the distributed application as a whole. Thus, Agrawal’s search for and analysis of candidate infrastructure elements discloses disputed limitations [1] and [2].

Accordingly, we sustain the Examiner’s 35 U.S.C. § 102(a)(1) rejection of claim 1, and the Examiner’s 35 U.S.C. §§ 102(a)(1) and 103 rejections of claims 3–5, 8–10, and 14, which Appellant does not argue separately. Appeal Br. 11.

#### Claims 6, 12

In rejecting claim 6 as obvious, the Examiner finds that Narayanaswamy’s authentication of “a person acting in the role of an EJB deployer” (Narayanaswamy 15:45–46) in providing access to a deployment system teaches or suggests [3] “wherein identifying the at least one candidate design is based on role-based access rules associated with a user deploying the application.” Final Act. 20 (citing Narayanaswamy 15:47–55);

Ans. 27–29 (further citing *Narayanaswamy* 16:2–7, 18:40–46, Figs. 4, 6, 10).

Appellant contends the Examiner erred because rather than disclosing or rendering obvious disputed recitation [3], “*Narayanaswamy* merely discusses a user in the role of deploying the application.” Appeal Br. 12–13; Reply Br. 2. Appellant’s argument is unpersuasive because, as the Examiner correctly finds, it is known to require credentials that allow access to a deployment system. *See* Ans. 27. That is, it would have been obvious to an artisan of ordinary skill that the best candidate infrastructure element of *Agrawal* would be one that the user deploying the application can access. Thus, using role-based access rules to identify candidate designs (i.e., designs that include infrastructure elements that can be accessed) would have been obvious to an artisan of ordinary skill. Therefore, we agree with the Examiner that the combination of *Agrawal* and *Narayanaswamy* teaches or suggest disputed limitation [3].

Accordingly, we sustain the Examiner’s 35 U.S.C. § 103 rejection of claim 6, and claim 12, which Appellant does not argue separately. Appeal Br. 13.

#### Claims 7, 13

In rejecting claim 7 as obvious, the Examiner finds that *Maes*’ metadata descriptions for various components of an application combined with *Spivak*’s use of checksums for integrity checking teaches or suggests [4] “creating a checksum for the service design based on a unique identifier associated with the service design” and [5] “comparing the checksum of the service design with a checksum of another application to be deployed, the

checksum of the another application being based on a unique identifier associated with the another application and revision and version of application components of the another application.” Final Act. 22 (citing Maes ¶¶ 20, 24, 25), 23 (citing Spivak ¶¶ 59–60, 62).

Appellant contends the Examiner erred because “Spivak does not mention or contemplate a checksum being used as an identifier, such as an identifier for a service design for an application platform.” Appeal Br. 14; Reply Br. 3. The Examiner, however, correctly finds that Spivak uses checksums “to identify any change” to a package. Ans. 33. Thus, the Examiner’s findings show that Spivak teaches or suggests that the checksum for one application can be compared with the checksum for another application to identify that the two applications are possibly the same or are different. Therefore, we agree with the Examiner that the combination of Agrawal, Maes, and Spivak teaches or suggests disputed recitations [4] and [5].

Accordingly, we sustain the Examiner’s 35 U.S.C. § 103 rejection of claims 7 and 13.



CONCLUSION

<b>Claims Rejected</b>	<b>35 U.S.C. §</b>	<b>References</b>	<b>Affirmed</b>	<b>Reversed</b>
1, 8–10, 14	102(a)(1)	Agrawal	1, 8–10, 14	
3–6, 12	103	Agrawal, Narayanaswamy	3–6, 12	
7, 13	103	Agrawal, Maes, Spivak	7, 13	
<b>Overall Outcome</b>			<b>1, 3–10, 12–14</b>	

TIME PERIOD FOR RESPONSE

No time period for taking subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a). *See* 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED