Trials@uspto.gov
571-272-7822

Paper No. 51
Date Entered: March 27, 2014

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

KYOCERA CORPORATION
MOTOROLA MOBILITY, INC
Petitioner

v.

SOFTVIEW LLC
Patent Owner

Case IPR2013-00007
Case IPR2013-00256
Patent 7,461,353 B2

Before BRYAN F. MOORE, BRIAN J. McNAMARA, and
STACEY G. WHITE, Administrative Patent Judges.

McNAMARA, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. 42.73

BACKGROUND

On March 29, 2013, in Paper 11, the Board entered a Decision to Institute an *inter partes* review on the following challenges raised by Kyocera Corporation to the patentability of claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 118, 138, 139, 149, 183, 252, 283, and 317 ("Challenged Claims") of U.S. Patent No. 7,461,353 B2 (the '353 Patent) owned by Softview LLC ("Patent Owner"):

> Challenged Claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 118, 138, 139, 149, 183, 252, 283, and 317as obvious under 35 U.S.C. § 103 over the combination of Zaurus[1], Pad++[2];
>
> Challenged Claim 66 as obvious over the combination of Zaurus, Pad++ and SVG[3];

[1] Power Zaurus Personal Digital Assistant Documentation ("Zaurus"), Ex. 1004

[2] Bederson, Benjamin B. and Hollan James D., Pad++: A Zoomable Graphical Interface System, CHI '95 Mosaic of Creativity, May 1995; Bederson, Benjamin B. and Furnas, George W, Space-Scale Diagrams: Understanding Multiscale Interfaces, CHI '95 Proceedings, 1995; Bederson, Benjamin B., et al, A Zooming Web Browser, SPIE, Vol. 2667, 260-71, May 1996; Bederson, Ben and Meyer, Jon, Implementing a Zooming User Interface: Experience Building Pad ++, Software-Practice and Experience, Vol. 28(1), 1101-35, Aug. 1998; Bederson, Benjamin B., et al., Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics, Journal of Visual Languages and Computing, Vol. 7, 3-31, 1996; Pad++ Reference Manual Version 0.2.7, published July 9, 1996; Pad++ Programmer's Guide Version 0.2.7, published June 10, 1996 (collectively, "Pad++"), Ex. 1006

[3] Ferraiolo, Jon, Scalable Vector Graphics Requirements: W3C Working Group Draft, Oct. 29, 1998. ("SVG"), Ex. 1007

Challenged Claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 118, 138, 139, 149, 183, 252, 283, and 317 as obvious over the combination of Zaurus, Tsutsumitake[4], and Hara[5];

Challenged Claim 66 as obvious under 35 U.S.C. § 103 over the combination of Zaurus, Tsutsumitake, Hara, and SVG.

IPR2013-00256, brought by Motorola Mobility LLC, raised the same challenges and later was joined to this proceeding. IPR2013-00256, Paper 10. Kyocera Corporation and Motorola Mobility are referred to collectively as "Petitioner."

On July 19, 2013, Patent Owner filed a response brief. (PO Resp., Paper 25). Petitioner filed a Consolidated Reply to Patent Owner's Response. (Paper 28). Patent Owner filed a Motion to Exclude. (Mot. to Exclude, Paper 41) and Petitioner replied (Reply to Mot. to Exclude, Paper 42). An oral hearing was held on January 7, 2014, concurrent with the oral hearing in related consolidated proceeding, IPR2013-00004/IPR2013-00257, between the same parties.

In this Final Written Decision, we determine pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73 that Petitioner has shown, by a preponderance of the evidence, that all challenged claims are unpatentable under 35 U.S.C. § 103.

THE '353 PATENT (EXHIBIT 1001)

As indicated by its title, the '353 patent is drawn to the scalable display of Internet content, e.g., Hyper Text Markup Language (HTML)-based content, cascade style sheet (CSS) content and XML content on mobile devices by enabling the content to be rendered, zoomed, and panned for better viewing on small screens

[4] Japanese Laid Open Patent Application H10-21224 ("Tsutsumitake"), Ex. 1005
[5] Japanese Unexamined Patent Application Publication H10-326169 ("Hara"), Ex. 1008

and standard monitors. Ex. 1001, col. 2, ll. 22-39, col. 5, ll. 18-22. A client side viewer receives the Internet content in an Internet browser, e.g., a micro-browser or a browser plug-in, and uses the simple vector format (SVF) originally designed to handle common computer aided design (CAD) file formats to describe the current web content. *Id.* at col. 4, ll. 43-63. Translation of the content into a scalable vector representation can be done by a third party proxy service (Fig. 1A), the content provider's web site (Fig. 1B) or at the client (Fig. 1C).

Figure 5 illustrates the logic used by the invention when translating content into a scalable vector representation. *Id.* at col. 3, ll. 50-52. Pre-rendering parsing of a received HTML document identifies elements such as tables, column definitions, graphic images, paragraphs, and line breaks, and determines where to place objects on a display. *Id.* at col. 15, ll. 45-52. When using frames, the display page is divided into multiple frame areas, which enables a single displayed page to include source code from several HTML documents. *Id.* at col. 15, ll. 33-36. During pre-rendering, each frame is examined in the sequential order it appears in the HTML document, and during further processing, actual objects are rendered in their respective positions. *Id.* at col. 15, ll. 52-57. The content is separated into objects based on logical groupings of content, and a page layout is built using bounding boxes produced for each object. *Id.* at col. 16, ll. 19-38, col. 17, ll. 15-29). The '353 Patent admits that the above steps commonly are performed by conventional browsers in the pre-rendering process, but indicates that the steps it discloses to use layout data generated in the pre-rendering process to generate a scalable vector representation of the original page content depart from the prior art. *Id.* at col. 17, ll. 30-45.

The '353 Patent discloses that generating a scalable vector representation begins by defining a page datum point as an X,Y value and a datum point as an

X,Y value for each object's bounding box. *Id.* at col. 17, ll. 45-64, col. 18, ll. 1-5. A vector between the page datum point and the datum point for each bounding box is then generated and stored. *Id*. A frame datum can also be assigned and vectors drawn from the page datum to the frame datum to establish the frame's offset and from the frame datum to each object in the frame. *Id.* at col. 18, ll. 5-16. The scalable vector representation then is completed by a reference that links each object's contents, attributes such as type (image, text), and bounding box parameters such as height and width to the object's vector. *Id.* at col. 18, ll. 18-26.

A display list of vectors for the vectorized HTML content is built, as is known from CAD arts, and a user selectable scale and offset are determined. *Id.* at col. 19, ll. 14-25. The bounding boxes are processed using the scale and offset, and a bounding box defining the limits of the display content is determined. *Id.* at col. 19, ll. 32-35. Scaling and offset can be accomplished either (i) by mapping vectors to a virtual display area in memory with much more resolution than the actual display, and reducing the scaling of the objects in the virtual display to how they will appear in the actual display, or (ii) by using a fixed reference frame corresponding to the client's screen resolution and scaling and offsetting the vectors's bounding boxes relative to the fixed frame. *Id.* at col. 19, ll. 39-56. Using the latter approach, respective offsets in X and Y (-ΔX and -ΔY) are applied to the starting point and the vectors are scaled by an amount SF, producing a new datum (starting point) for each bounding box relative to the rendered page datum, which remains fixed, but may, or may not, be displayed depending on the offset and scaling. *Id.* at col. 19, l. 58 - col. 20, l. 17. Once the bounding boxes are offset and scaled, the content (e.g., image and text) corresponding to objects having at least a part of their bounding boxes on the screen is retrieved from the client device's display list and scaled. *Id.* at col. 20, ll. 18-44. A display limit bounding

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.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase® Smarter legal research.