UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS CO., LTD.,
Petitioner

v.

DODOTS LICENSING SOLUTIONS LLC,
Patent Owner

Case IPR2023-00701

U.S. Patent No. 8,510,407 B1

DECLARATION OF JOHN KEMBEL

I, John Kembel, declare as follows.

I. Introduction

1. I am one of the named co-inventors for United States Patent No. 8,510,407. At the time of the invention, I was the Chief Technology Officer and a co-founder of DoDots, Inc.

2. Along with my twin brother, George Kembel, we founded DoDots, Inc. with a goal of disrupting and changing the paradigm of how internet content was created by companies and how it was delivered to users.

3. As discussed below in detail, we realized that existing web browsers had significant limitations -- trapping content in monolithic documents, and requiring users to navigate from site to site, one page at a time. This presented an unexpected opportunity for a novel internet technology, one that broke out of the page-based browsing model and instead packaged interactive content in forms that felt more like "apps", while maintaining all the flexibility of the web.

4. At the time, other (large) technology companies were taking a range of approaches to advancing the web, including: developing portals, media channels, news streams, even pursuing the webification of operating systems. We took a different approach, breaking out of the dominant browser and web-page paradigm, and envisioning a more "app"-centric internet. This paradigm proved to be prophetic as it is now reflected in the ubiquitous App and App ecosystem.

DoDots Exhibit 2021
Samsung v. DoDots - IPR2023-00701

II. Identifying a Long-Felt Need & a Non-obvious Innovation Opportunity

5. As a threshold matter, I am not a lawyer, but I have been informed by counsel that courts may often look at "evidence that an invention satisfied a long-felt and unmet need that existed on the patent's filing date." With this in mind, there was certainly the existence of a need to change how users accessed and interacted with content on the internet. At the time of invention, my brother and I began to appreciate that there was a constellation of needs from users and content providers that was not being met by the primary means of accessing internet content at the time, a web browser. Specifically, the limitations of web browsers and page-based web sites kept companies from engaging customers directly, content creators from developing interactive app-like experiences, and users from easily accessing the information and functionality they wanted from the burgeoning internet.

6. As our patents note, "internet content and web applications are designed to fill the entire web page." *See* '083 patent, 2:5-35. "Users and application developers therefore have limited control over the presentation of internet content: content is typically trapped within the frame of the browser." *Id.* Under this paradigm, the internet at the time of the invention had the following limitations:

a) Internet content was typically viewed by users one page at a time, with each page displayed serially in a browser application window which took up the majority of a user's computer screen real estate.

b) Internet content was designed by companies and web developers for delivery in the form of web pages. Even if content was modularized, it was reassembled into full web pages and served up as part of a larger document or site.

c) There was a distinction both visually and architecturally between the "viewer application" (browser) and the "content/document" (web page) such that a browser window was not tailored to the content being displayed, but rather mediated the display of any web page.

d) Internet content was effectively limited and trapped within the "frame" of the browser (viewer application). Therefore, content and web developers were limited in how interactive their content could be as well as in the types of user experiences they could offer.

e) Although internet programming technologies (such as Java script, CSS, layers, flash, etc.) gave web pages more functionality, the pages had limited access to application functionality such as controlling the window and frame, the size of the frame, branding, application behavior such as size and menu items, etc.

See also '083 patent, 2:60-3:29.

7. As we began to commercialize our product, it became apparent that multiple people had recognized the limitations of accessing internet content with a web browser.

8. In fact, leading experts in the field recognized that the paradigm of web browsers navigating web pages was holding the internet back. In a February 21, 2000 article published in Forbes magazine, the author wrote "Tim Berners-Lee, the creator of the World Wide Web, thinks [the browser] stifles creativity. Jakob

Nielsen, the reigning guru of Web usability, thinks [web browsers are] a disgrace. Bill Gates wishes it would just go away and become 'part of the operating system.'" *See* Ex. 2022 (Gimein, Mark, "Imagining a Web Beyond the Browser," Forbes, Feb. 21, 2000 (DODOTS0026225)).

9. These experts and others pointed to the "bookification" of the internet and emphasized that it created a slow and unwieldy process of browsing web content by clicking on links to navigate forward or backward, step-by-step. Moreover, the paged-based paradigm of the web led to information being predominantly organized in hierarchical structures akin to document outlines – which worked well for news, articles, and other static content, but was cumbersome and counterintuitive as content became increasingly interactive.

10. The web browser also meant whoever owned the initial browser landing page, search engines and portals became the mediator of the user experience separating users from direct access to the content and services they might want and companies from their customers. And the rising internet disconnected companies from their customers, customers from their solution providers and developers and users. It anointed a new class of intermediaries (web portals, search engines and more) to navigate the web page by page document model.

11. Indeed, the 2000 Forbes article noted that the prevailing knowledge was that people "hate[d] the browser because it turns the Web into nothing more than 'pages.' That metaphor works well for reading text, but less well for other purposes. Like a book, the browser functions best when one is looking at one page, and much worse if one wants to look at several at once. It does little to help people exchange information (unless pasting Web addresses into email is your ideal information-management solution). When it comes to building applications, adapting software to the 'page' makes for clunky design." *See* Ex. 2022 at DODOTS0026225.

12. This point was echoed by another article stating that "[t]he big problem with the typical browser is that it relies on the metaphor of pages in a book; you can view only one page at a time then click on another. That is fine for displaying a simple brochure, but it's a big step backward as far as users' interfaces are concerned. Some experts liken it to the limited interfaces of computers in the 1970s. The printed page is a wonderful thing for communication but it is crippling what we can do on the internet." *See* Ex. 2023, Lyons, D., "Browser Bashers," Forbes, May 15, 2000 (DODOTS0026105).

13. As such, navigating the internet was a tedious process. One article noted that "There's too much happening, and accessing even the simplest stuff - a quickie stock quote or weather forecast - takes too long. Sure, I can bookmark

favorite sites or customize a portal page. But those involve surfing from site to site. What if I'm not in the mood, or simply don't have the time? Why must I be exposed to a full Web page each time I want to digest a slice of the material on display?"*See* Ex. 2024, Baig, E., **"**Bits of Organization Clear Clutter," USA Today, (DODOTS 0026099).

14. Elaborating on this point, another article noted that browsers were an issue because "[l]inks go only in one direction instead of two, so it is difficult to retrace your steps; it takes too many mouse clicks to get from one place to another; the arrangement of information sometimes makes no sense. And many sites are just plain ugly." *See* Ex. 2023 at DODOTS0026105-06.

15. Put another way, "the basic problem is that typical Web sites arrange information in a hierarchical structure. You spot an underlined 'link,' click on it and go down one step, then another, then another-then realize you're in the wrong place. So, you back up, back up, back up, then start all over again on a different ladder." *Id*. at DODOTS0026104.

16. This archaic structure was simply cumbersome. "Nobody would expect to explore the world's biggest library page by page, yet some would say that's exactly what it's like trying to get around on the Internet" using a web browser. Navigating, using a web browser was neither "intuitive nor as visual as it

ought to be." *See* Ex. 2025, Heim, K., "Tools turn browser into galaxy," SJ Mercury News, July 7, 2000 (DODOTS0026228).

17. In fact, Donald Norman – recognized interaction design guru and President of Unext's learning division – at the time noted, "We need to just forget the browser. The browser has set us back ten years." *See* Ex. 2023 at DODOTS0026106-07.

18. In sum, a "growing dissatisfaction with the way the Web looks and works. Many critics blame the browser." And, another expert noted "Web usability is a disgrace. … For seven years, we've had close to zero progress in enhancing the user's main tool for using the internet." *Id*. at DODOTS0026105.

19. People craved change. In May 18, 2000, an article in the Wall Street Journal also noted that "[f]or lots of people, the Web has ceased to be about idle browsing. Instead, they've settled on a routine collection of sites related to their work or interests, …. When you approach the Web that way, the standard browser experience becomes tedious." *See* Ex. 2026, Mossberg, W., "New Metabrowsers Allow You To Create a Quilt of Web Pages," Wall Street Journal, May 18, 2000 (DODOTS0026232).

20. Because of these issues several companies were working on ways to change the basic architecture of standard web pages. *Id.* For example, Microsoft sought to merge the internet into the operating system through "webification" of

their ubiquitous Windows products. Likewise, other companies created web portals, media channels, and search engines. But none of those strategies moved in the direction our company eventually took.

21. This is where we directed our energies because the limitations of web browsers and page-based web sites were clear. We believed that the potential of the internet extended far beyond simply consuming information and content trapped in documents.

III. Inventing & Developing a Novel Internet Technology: The Dot

22. To address these limitations, we applied a rigorous design and innovation process to create a new and novel set of internet technologies. We realized that by breaking out of the dominant page-based paradigm of the web, companies would be able to engage their customers directly, web developers would be able to develop new app-like experiences, and users would be able to more easily access the information and interactive functionality they wanted from the rapidly growing internet.

23. We envisioned a new way to package and distribute interactive web content outside of the web browser in small configurable windows that felt more like native software applications than static web pages. By packaging and delivering select web content in lightweight and configurable windows, new internet-powered and interactive "app"-like experiences were made possible across

a range of computer and mobile devices. We called these new web-based applications "Dots".

24. We founded a venture-backed startup ("DoDots") to pursue this vision and to commercialize the technology. At the time, our company's marketing materials described Dots as follows:

See Ex. 2027 – DoDots Brochure (DODOTS0025768).

25. Each Dot could be tailored for a specific purpose and deliver select information and interaction features accordingly – such as for weather, games, dictionaries, maps & traffic, financial information, or e-commerce. As an example, here's a screenshot of one of the first Dots offered by ecommerce leader mySimon:

mySimon Dots - Price Comparison Service in Dots
When you want to buy, you want to buy, not shop! mySimon has become the Internet's favorite shopping tool, offering price comparisons with quick results. The mySimon Dot offers an ever-ready search window so that you can get right to your results with minimal hassle.

Ex. 2033 at DODOTS0024868.

26. Dots could be branded by the companies or web developers that created them, and they could be distributed directly to their end-users, rather than having to reach customers through content mediators such as portals and search engines. Moreover, instead of navigating cumbersome browsers to different web sites (page-by-page), users could simply open up the Dots they wished to use. And, because Dots lived outside the browser, they could stay visible and persistent as users used other native software applications. An example of the range of Dots could been seen in the example below:

DoDots Exhibit 2021
Samsung v. DoDots - IPR2023-00701

Id. Ex. 2028 at 11.

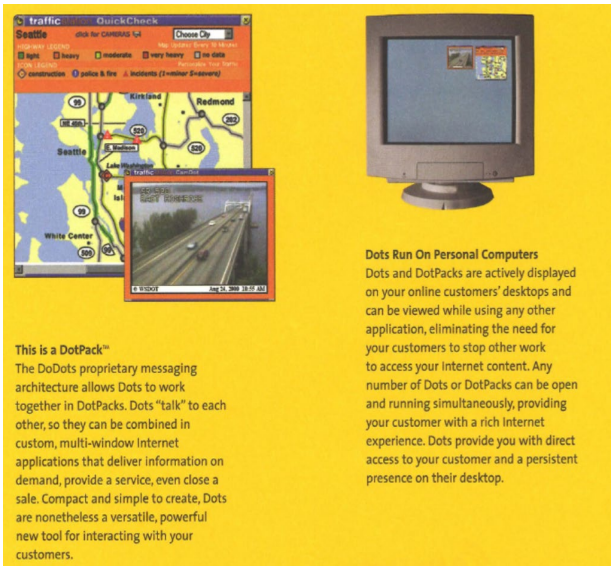
27. Because Dots could be persistent on a user's computer or device, companies could make timely information and interactive applications available directly to their customers, as described below.

Deliver News and Data on Time in Real Time
CNET uses Dots to distribute timely industry news, product reviews, and shopping information. Like all Dots, CNET Dots can be configured to open upon startup to provide viewers with industry news when they turn on their PCs or other devices at the beginning of the day. Dots can be set either to sit on top of other applications or recede to the background—but still update in real time to meet users' information needs.

Ex. 2027 at DODOTS0025771.

28. And, since Dots were independent of each other, multiple Dots could be used at the same time. Users could collect, organize, and share Dots in sets called "DotPacks". DotPacks could be easily opened and closed with a single

click. This simplified task switching and helped users seamlessly integrate useful Dot applications into their day-to-day computing experience. Our marketing materials explained this as follows:



Id. at DODOTS0025768.

29. Leveraging these capabilities, companies could offer sophisticated, real-time, multi-dot "application" experiences, tailored for their customers' unique needs. Moreover, companies could better track and understand how their users interacted with specific content. The following financial Dots show this:

Provide and Gather Information

Some companies already use Dots to package and distribute financial information specific to each customer's requests. DoDots technology allows you to "cluster" information such as charts, customized portfolios, industry research, and—eventually—advertising and transaction capabilities in a single DotPack. While the customer gets information that encourages more transactions, you gain a better understanding of how your content is being accessed and used.

Id. at DODOTS0025773.

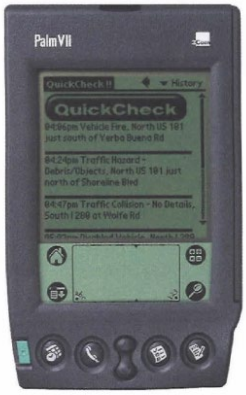
30. Additionally, because Dots enabled interactive web content and applications to be displayed in smaller, dedicated windows, Dots could naturally be made available on various devices with displays of all sizes, as noted below:

New Distribution for New Devices

DoDots technology will allow you to distribute content and applications to a wide range of portable and other next-generation Internet acces devices. With Dots, it's easy to scale and optimize your content and applications for delivery, display, and performance on any device.

Id . at DODOTS0025772.

31. Importantly, this also allowed for Dots to offer a seamless experience across customer mobile devices as seen in the following excerpt:



Dots Go Mobile
The DoDots digital infrastructure allows you to automatically extend Dot-based content and applications to your customers' mobile devices, letting you reach your customers with relevant, branded content or applications anytime and anywhere. Our solution allows you to optimize content delivery for each device, while providing a seamless customer experience, on and off the desktop.

Wherever the Internet Goes, Dots Go
The DoDots extensible architecture will allow you to deliver Dot-based solutions to virtually any Internet-enabled device and is designed to accommodate the future innovations of the industry, our partners, and our developers. As the Internet moves off the desktop and as customer behavior evolves, our technology assures long-term viability for your content, applications, and services. Wherever the Internet goes, Dots will go.

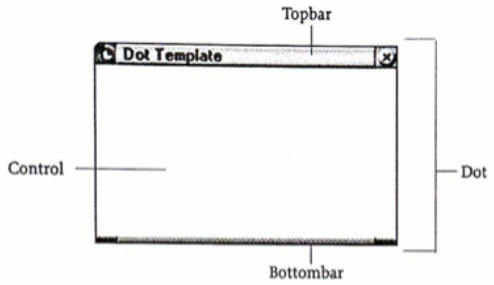
Id. at DODOTS0025769.

32. At the heart of the Dot architecture was the notion of "Dot Templates" (also known as "Dot Definitions"). Dot Templates defined the "package" – the attributes of the small, dedicated windows – for each Dot's web content and interactive applications, as described in the excerpt below:

DoDots Technical Overview

THE DOT™ DEFINES THE PACKAGE FOR CONTENT AND APPLICATIONS

Dots™ are packaged pieces of branded Internet content and applications. The Dot descriptions are constructed of three primary elements: the Topbar, the Bottombar, and the Web Control. Groups of Dots, called DotPacks™, can work together, using Dot messaging, to give users a full application experience. Through the DevDot™ and Dot messaging, developers have access to all the elements of Dots and DotPacks, and can create branded, smart packages targeted to specific customers.



Ex. 2028 – Technical Overview at 1.

33. Dot Templates could consist of three primary elements: a Topbar, a Bottombar, and a Web Control. Every element could be designed and branded in a manner consistent with the Dot's functionality and purpose. *Id.*

34. Through the Dot Templates, DoDots offered a lightweight and configurable system that enabled fully configurable Dots with content in interactive applications that existed outside the browser.

35. Additionally, Dot-enabled devices were described in the following excerpts:

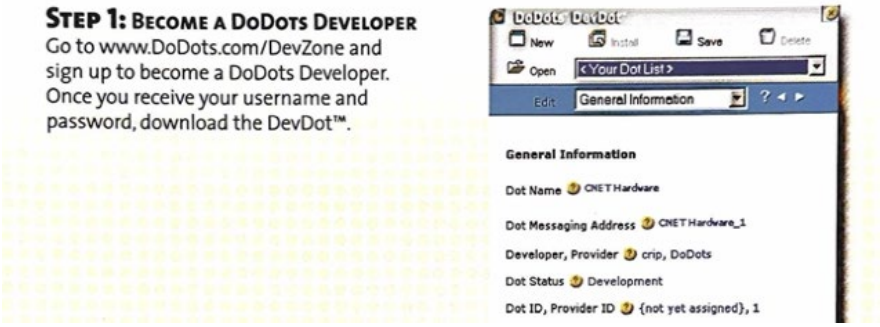
THE DODOTS CLIENT SOFTWARE PROVIDES THE FOUNDATION FOR THE DOT EXPERIENCE

Users are asked to Dot-enable their computers the first time they request a Dot. Dot-enabling a computer installs a copy of the DoDots client software. The user also receives a unique Dot-user account, a Welcome Dot, and the HomeDot. Users organize Dots and collect more Dots with the HomeDot. After Dot-enabling their computers, users are ready to collect and use Dots.

Ex. 2028 – Technical Overview at 4-5. The Dot-enabled devices could access collection of Dots – get new Dots, open and close specific Dots and DotPacks, even share Dots and DotPacks.

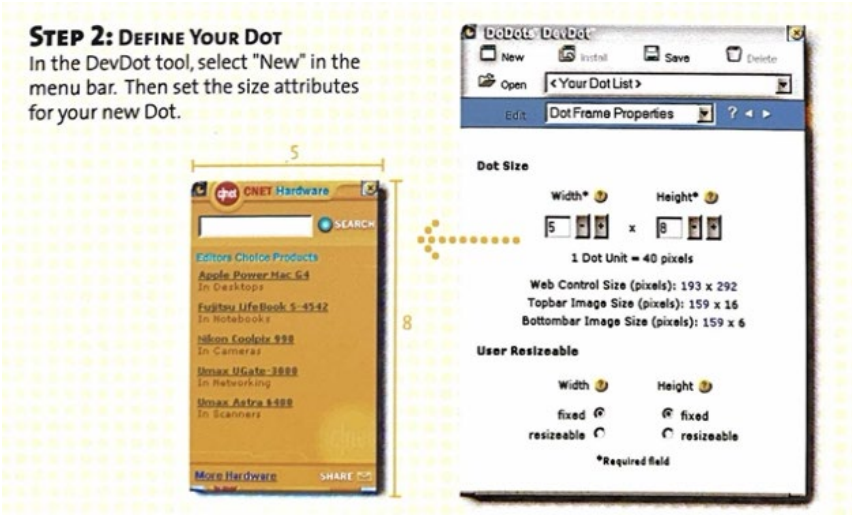
36. As we continued to refine and develop the Dots technology, we created a digital infrastructure and development platform for content creators and web developers to create and publish Dots.

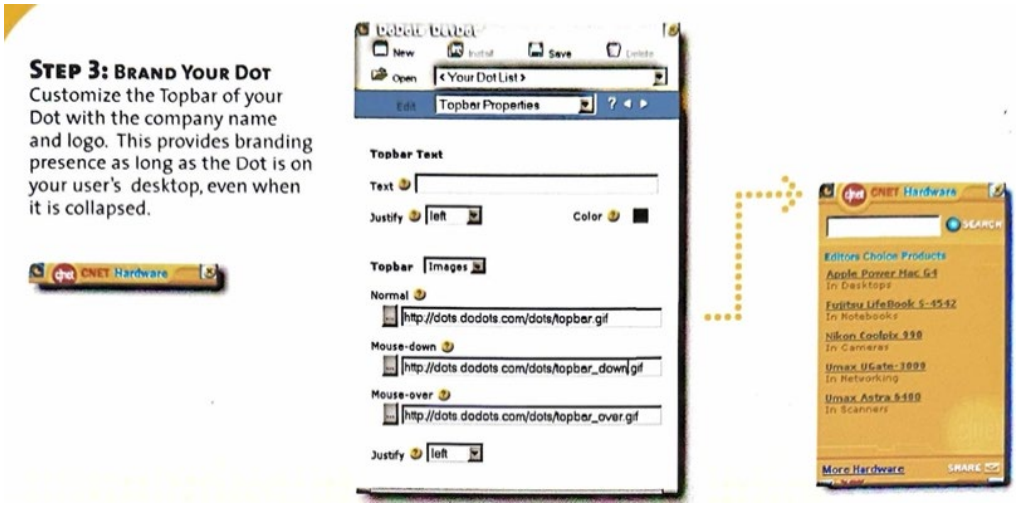
37. Becoming a Dot Developer was easy, as it did not require advanced software coding skills, but rather was accessible to anyone versed in the fast-emerging web design and development practices. As such, we launched a Dot Developer program and made development tools and documentation available to companies, brands, and independent developers.



Ex. 2029 (DoDots Developer Brochure) at 1.

38. Using our tools, a Dot Developer could quickly define a new Dot, set its window size and attributes, and customize its branding. This was done as follows:





Id. at 1-2.

39. A Dot Developer could create the interactive web content and application functionality for the Dot. Anything that could be rendered in a browser could be packaged in a Dot. Moreover, this "Dot Content" could be hosted and served by their own standard web servers. Companies and developers appreciated this as developing Dots did not require extra expensive tooling or custom hardware.

Dot Development: Easy and Inexpensive
The first step in Dot development is designing and generating content for the Dot. Any content that can be displayed in a Web page can be displayed in a Dot, including text and graphics, streaming media, Flash, HTML, JavaScript, and secure transactions. Dot developers can use their existing tools and methods to develop Dot content.

Ex. 2027 at DODOTS0025772.

40. In addition to offering interactivity through the development of Dot Content, Dot Developers could also leverage the DoDots "Dot Messaging"

protocol to give users a full, application-like experience. Dot Messages enabled Dot Content to modify a Dot's behavior, communicate with and access additional HomeDot services, and interact with other Dots. Examples of Dot messages included:

Examples

Send a message to the system:

#system

Send a message to this dot:

#.:

Send a message to this control:

#.:.

Send a message to the dot with ID=7: (note that dot ID's are volatile and should not be hard-coded into your scripts)

#7:

Send a message to the first dot of kind=A that can be found in the system:

#A#any:

Send a message to the control with ID=5 in the first open dot of kind=B found in the system:

#B#open:5

Send a message to this dot's control with ID=3

#.:3

Send a message to the first hidden control of kind=X that can be found in this dot

#.:X#closed

Send a message to the first visible control of kind=Y that can be found in the first closed dot of kind=C:

#C#closed:Y#open

Messages

System messages

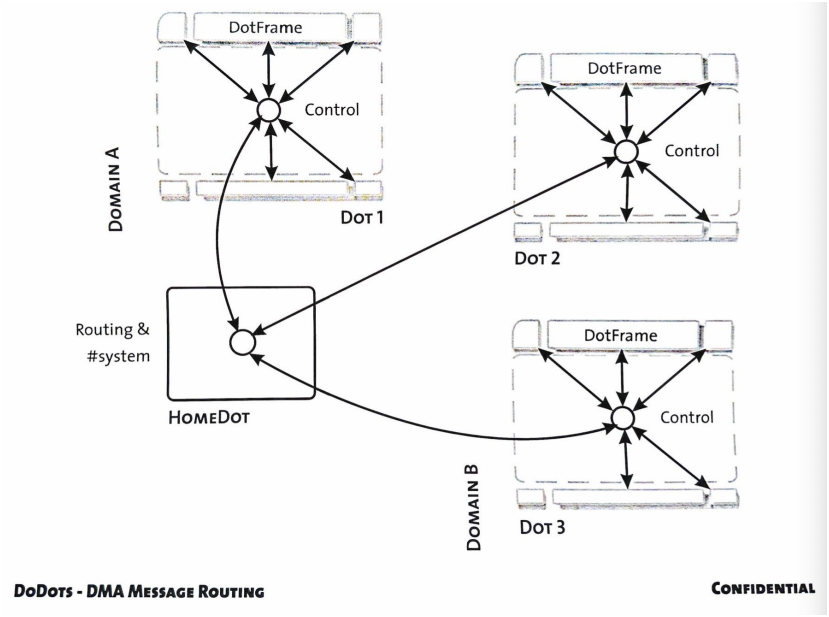
#refresh
#install-dot <dot-url>
#have-dot <dot-address>
#delete-dot <dot-address>
#quit
#save-to-server
#get-screen-width
#get-screen-height
#close-all-dots
#open-preset <preset-name>
#take-preset <preset-name>
#get-dot-ids <dot-address>

Dot messages

#set-title
#set-size <width> <height>
#set-width <width>
#set-height <height>

Ex. 2045 at 2.

And, Dot messages could be routed as follows:



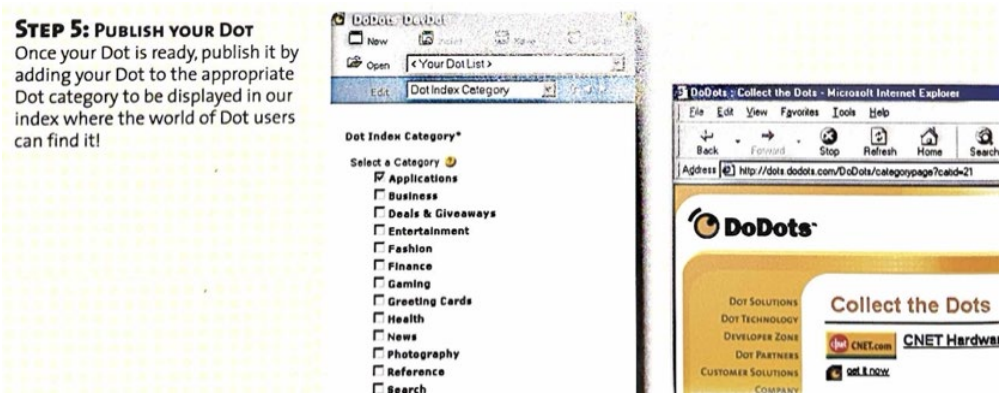
Ex. 2045 at 3.

41. Once a Dot's Content was designed and developed, a Dot developer finished defining their new Dot by specifying an URL pointing to where that content was hosted.

Ex. 2029 – Dot Developer Guide at 2.

42. A key benefit of allowing companies and developers to host their own Dot Content was that the content could be delivered directly to end users without going through any intermediaries such as portals, search engines, or even browsers.

43. Finally, to publish a new Dot, a Dot Developer simply needed to provide enough information to index it appropriately and for end users to find it.



Ex. 2029 – Dot Developer Guide at 2.

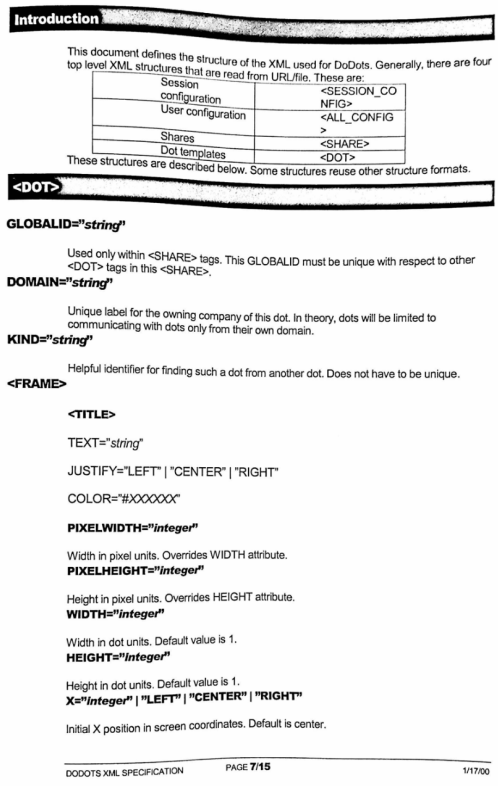
44. All of the relevant characteristics of a Dot's design could be described in XML as described below:

Content Publishing

DoDots live packages are called ***Dots***. ***Dot Templates*** are content package definitions from which ***Dot Instances*** (or just ***Dots***) are created. Dot Templates and Dots are defined and managed as small XML records.

Ex. 2027 at DODOTS0025850.

45. DoDots maintained XML schemas for valid Dot Templates (along with Dot Shares, User Profiles, and Sessions) . Here's an excerpt from such a spec:



Introduction

This document defines the structure of the XML used for DoDots. Generally, there are four top level XML structures that are read from URL/file. These are:

Session configuration	<SESSION_CONFIG>
User configuration	<ALL_CONFIG>
Shares	<SHARE>
Dot templates	<DOT>

These structures are described below. Some structures reuse other structure formats.

<DOT>

GLOBALID="*string*"

Used only within <SHARE> tags. This GLOBALID must be unique with respect to other <DOT> tags in this <SHARE>.

DOMAIN="*string*"

Unique label for the owning company of this dot. In theory, dots will be limited to communicating with dots only from their own domain.

KIND="*string*"

Helpful identifier for finding such a dot from another dot. Does not have to be unique.

<FRAME>

<TITLE>

TEXT="*string*"

JUSTIFY="LEFT" | "CENTER" | "RIGHT"

COLOR="#*XXXXXX*"

PIXELWIDTH="*integer*"

Width in pixel units. Overrides WIDTH attribute.

PIXELHEIGHT="*integer*"

Height in pixel units. Overrides HEIGHT attribute.

WIDTH="*integer*"

Width in dot units. Default value is 1.

HEIGHT="*integer*"

Height in dot units. Default value is 1.

X="*integer*" | "LEFT" | "CENTER" | "RIGHT"

Initial X position in screen coordinates. Default is center.

Ex. 2045 at 4.

46. When a new Dot was published the Dot's description was saved to the DoDots application server, and the Dot became immediately accessible to end users.

How Dots Are Served

When a Dot developer creates and publishes a Dot, a new Dot description is saved to the DoDots application server. The Dot description defines the Dot package and behavior. URLs in the Dot description point to the Dot content, which is hosted on the Dot provider's Web server.

When installed, the DoDots client software sets up and maintains a HomeDot™ on the user's computer or other access device. The HomeDot acts as a manager of all Dots the user might collect. The client software also creates a user account on the DoDots application server the first time it connects with the server.

Ex. 2027 at DODOTS0025772.

DoDots Exhibit 2021
Samsung v. DoDots - IPR2023-00701

47. End users could collect Dots they were interested in a variety of ways. And instead of downloading custom software for each Dot "application", only Dot Templates needed to be transferred to the user's device. A Dot's definition was all that was required for a new Dot instance to be rendered, pulling all necessary branding and interactive content directly from the internet.

DOTS ARE DISTRIBUTED TO USERS

In contrast to destination-based and page-based Web content, Dots are distributed to users. Users instantly collect the Dots they want. Once collected, a Dot remains with the user until the user decides to discard it. DoDots provides several ways to distribute Dots:

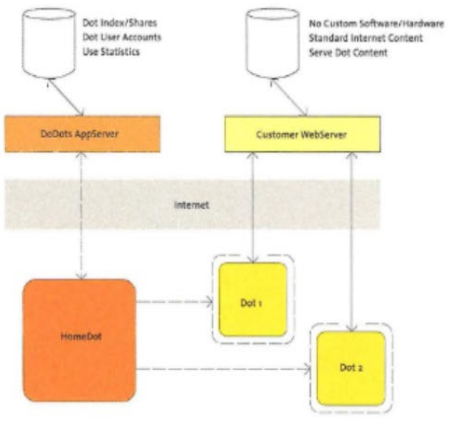
- Links to Dots can be inserted anywhere a Web link can be placed, including on Web pages and in email.
- Users can collect Dots from an index of Dots maintained by DoDots.
- Users can use DoDots' share function to quickly send copies of favorite Dots to friends and co-workers.

Ex. 2028 at 4.

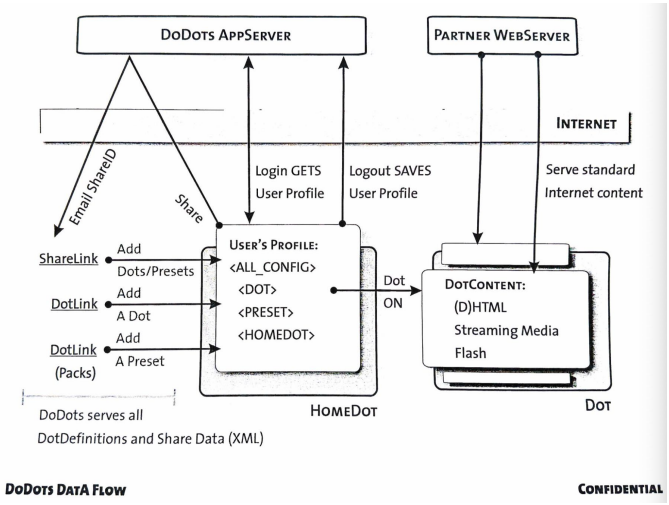
48. The Dot digital infrastructure allowed for the transmission of requests for Dot descriptions or content for dots over a web server, receive those Dots descriptions, and the presentation of information on devices after receiving that content, as seen in the exemplary diagrams below:

Technically Speaking

DoDots application servers deliver millions of Dot descriptions and user accounts. Your content is still delivered direct from your servers to support the scalability needs of your company.



Ex. 2027 at DODOTS0025773.



Ex 2045 at 1.

49. This technology was decades ahead of its time. Indeed, we believed that creating interactive web applications that felt like user-friendly apps rather than "pages" had the potential to change the way users interacted with the Web. A thought validated by the fact that our invention has been adopted across modern mobile devices.

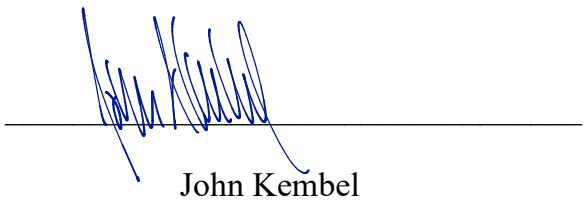
50. I was involved in the design/development and commercialization of Dots at DoDots and am also an inventor on all the DoDots patents. Based on my knowledge of Dots as well as my review of DODOTS-2026, 2027, 2028, 2029, and 2045, which is source code and detailed technical materials describing the operation/configuration of Dots, I believe that the '083 patent, '407 patent, and '545 patent accurately describes the Dot technology.

IV. Closing

51. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the results of these proceedings.

I declare under penalty of perjury that the foregoing is true and correct.

Sworn to this 20th day of February, 2024 in BOULDER, CO.



John Kembel

DoDots Exhibit 2021
Samsung v. DoDots - IPR2023-00701