

EXHIBIT A

**Eolas' Proposed Claim Constructions and Identification of
Intrinsic and Extrinsic Evidence for the '906 and '985 Patents**

No.	Claim Term, Phrase, or Clause	Eolas' Proposed Construction	Extrinsic Evidence	Intrinsic Evidence (cites are to '985 specification) ¹
1.	“automatically [invoking / invoke] [the / said] executable application”	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: automatically calling or activating the <u>executable application</u> .	Microsoft Press Computer Dictionary at 196, “invoke” (c. 1991)	3:38-42
2.	“executable application is automatically invoked by the browser”	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: <u>executable application</u> is automatically called or activated by the browser.	Microsoft Press Computer Dictionary at 196, “invoke” (c. 1991)	3:38-42
3.	“workstation”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: a computer system connected to a network that serves the role of an information requester		Fig 2; Fig 3; Fig 4; 1:30-37; 4-17-52; 6:8-15; 6:39-53
4.	“network server”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: a computer system that serves the role of an information provider		Fig 2; Fig 3; Fig 4; 4:25-52; 6:39-50
5.	“executable application”	any computer program code, that is not the operating system or a utility, that is launched to enable an end-user to directly interact with data	Prior construction of term in <i>Eolas Techs. v. Microsoft Corp.</i> , 2000 U.S. Dist. LEXIS	Fig. 5; Fig. 9; 6:41-50; 6:53-59; 8:49-64; 9:13-36; 9:41-43; 9:48-51; 9:59-64; 10:66-11:4; 12:51-13:16; 13:22-26

¹ U.S. Patent No. 7,599,985 is a continuation of U.S. Patent No. 5,838,906. The specifications of the two patents are nearly identical.

No.	Claim Term, Phrase, or Clause	Eolas' Proposed Construction	Extrinsic Evidence	Intrinsic Evidence (cites are to '985 specification) ¹
			18886, at *56 (N.D. Ill. Dec. 28, 2000) <i>aff'd</i> by <i>Eolas Techs., Inc. v. Microsoft Corp.</i> , 399 F.3d 1325, 1336 (Fed. Cir. 2005)	
6.	"object"	text, images, sound files, video data, documents or other types of information that is presentable to a user of a computer system		Fig. 1; Fig. 5; 1:59-2:1; 2:12-44; 2:53-59; 3:23-25; 3:31-45; 4:53-56; 5:7-16; 5:31-42; 6:16-35; 6:39-41; 9:32-36
7.	"type information"	<p>No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p>any information used by the browser to identify and locate the executable application, and may include the name of an application associated with the <u>object</u></p>	<p>Prior construction of term in <i>Eolas Techs. v. Microsoft Corp.</i>, 2000 U.S. Dist. LEXIS 18886, at *56 (N.D. Ill. Dec. 28, 2000)</p>	12:60-13:31
8.	"file"	<p>No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p>a named collection of data.</p>	<p>Microsoft Press Computer Dictionary at 144, "file" (c. 1991)</p> <p>The New IEEE Standard Dictionary of Electrical and</p>	

No.	Claim Term, Phrase, or Clause	Eolas' Proposed Construction	Extrinsic Evidence	Intrinsic Evidence (cites are to '985 specification) ¹
			Electronics Terms at 498, "file" (c. 1993)	
9.	"[first] hypermedia document"	<p>No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p>a document that allows a user to click on images, sound icons, video icons, etc., that link to other objects of various media types, such as additional graphics, sound, video, text, or hypermedia or hypertext documents</p>		Fig. 1; 1:59-2:34
10.	"[first] distributed hypermedia document"	<p>No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p><u>[first] hypermedia document</u> that allows a user to access a remote data object over a network.</p>		Fig. 1; 1:59-2:34; 5:24-40
11.	"file containing information to enable a browser application to display [, on] [said/the] [client workstation,] at least [a / said] portion of [a / said] distributed hypermedia document"	<p>No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p>the <u>file</u> contains information to allow the browser application to display at least part of a <u>distributed hypermedia document</u>.</p>		Fig. 1; 1:59-2:34; 5:24-40
12.	"text format"	<p>No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p>text that initiates processing.</p>		17:6-24:9
13.	"embed text format"	<p>No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p> <p><u>text format</u> for embedding an <u>object</u>.</p>		12:55-13:19
14.	"embed text format, located at a first location in said first distributed hypermedia"	<p>No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean:</p>		Fig. 1; 1:59-2:34; 5:24-40; 12:55-13:19

No.	Claim Term, Phrase, or Clause	Eolas' Proposed Construction	Extrinsic Evidence	Intrinsic Evidence (cites are to '985 specification) ¹
	document”	<u>embed text format</u> located at a first location in the <u>first distributed hypermedia document</u> .		
15.	“embed text format [which] correspond[s/ing] to [a / said] first location in the document”	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: <u>embed text format</u> which relates to a first place in the document.		12:55-13:19
16.	“distributed application”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: an application that may be broken up and performed among two or more computers.		Fig. 6; 10:63-11:10.
17.	“A computer program product . . . comprising a computer usable medium having computer readable program code physically embodied therein, said computer program product further comprising: computer readable program code for causing said client workstation to execute a browser application”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean that: the computer program product that includes a computer usable medium having computer readable program code for causing the client workstation to execute a browser application.		2:67-3:3; 8:49-53; 8:65-67; 9:7-9; 9:14-17
18.	“computer readable media encoded with software”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: computer readable media having software.		2:67-3:3; 8:49-53; 8:65-67; 9:7-9; 9:14-17
19.	“pars[e/es/ed/ing]”	No construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: to break an input into smaller pieces.	Microsoft Press Computer Dictionary at 259, “parse” (c. 1991)	Fig. 7A; Fig 7B; Fig 8A; Fig 8B; 9:7-11

No.	Claim Term, Phrase, or Clause	Eolas' Proposed Construction	Extrinsic Evidence	Intrinsic Evidence (cites are to '985 specification) ¹
20.	"identify[ing] an embed text format"	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: identifying an <u>embed text format</u> .		12:55-13:19
21.	"an embed text format . . . is identified"	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: an <u>embed text format</u> is identified.		12:55-13:19
22.	"specifies the location of at least a portion of [an / said] object"	No further construction of this term is needed. In the alternative, to the extent a construction is deemed necessary, this term should be construed to mean: specifies the location of at least part of an <u>object</u> .		Fig. 1; Fig. 5; 1:59-2:1; 2:12-44; 2:53-59; 3:23-25; 3:31-45; 4:53-56; 5:7-16; 5:31-42; 6:16-35; 6:39-41; 9:32-36

Eolas Contends That No Claim Limitations In Any Of The Asserted Claims Of The Patents In Suit Should Be Governed By 35 U.S.C. § 112(6). It Provides The Identified Corresponding Structure Below In The Alternative Only.

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
1.	claim 6 of the '906 patent	computer readable program code for causing said client workstation to execute a browser application to parse a first distributed hypermedia document to identify text formats included in said distributed hypermedia document and to respond to predetermined text formats to initiate processes specified by said text formats;	Figs. 4-7; 8:36-11:17; 12:50-14:63
2.	claim 9 of the '906 patent		
3.	claim 10 of the '906 patent		
4.	claim 6 of the '906 patent	computer readable program code for causing said client workstation to utilize said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server, wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation, wherein said first distributed hypermedia document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location of at least a portion of an object external to the first distributed hypermedia document, wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and wherein said embed text format is parsed by said browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable an end-user to directly interact with said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window.	Figs. 4-8; 8:36-11:17; 12:50-16:8
5.	claim 10 of the '906 patent		
6.	claim 7 of the '906 patent	wherein said executable application is a controllable application and further comprising: computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation via interprocess communications between said browser and said controllable application.	Figs. 5-6, 8, 10; 6:63-7:6; 8:56-12:49; 14:64-16:7; 16:28-16:46

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
7.	claim 7 of the '906 patent	computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation via inter-process communications between said browser and said controllable application.	Figs. 5-6, 8, 10; 6:63-7:6; 8:56-12:49; 14:64-16:7; 16:28-16:46
8.	claim 9 of the '906 patent		
9.	claim 10 of the '906 patent		
10.	claim 8 of the '906 patent	wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched.	Figs. 5-6, 8, 10; 6:63-7:6; 8:56-12:49; 14:64-16:7; 16:28-16:46
11.	claim 13 of the '906 patent	wherein additional instructions for controlling said controllable application reside on said network server, wherein said computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation includes: computer readable program code for causing said client workstation to issue from the client workstation, one or more commands to the network server; computer readable program code for causing said network server to execute one or more instructions in response to said commands; computer readable program code for causing said network server to send information to said client workstation in response to said executed instructions; and computer readable program code for causing said client workstation to process said information at the client workstation to interactively control said controllable application.	Figs. 4-6; 1:45-60; 5:24-5:38; 6:50-7:6; 8:36-12:50
12.	claim 9 of the '906 patent		
13.	claim 10 of the '906 patent	wherein additional instructions for controlling said controllable application reside on said network server, wherein said computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation includes: computer readable program code for causing said client workstation to issue, from the client workstation, one or more commands to the network server; computer readable program code for causing said network server to execute one or more instructions in response to said commands; computer readable program code for causing said network server to send information to said client workstation in response to said executed instructions; and	Figs. 4-6; 1:45-60; 5:24-5:38; 6:50-7:6; 8:36-12:50

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
		computer readable program code for causing said client workstation to process said information at the client workstation to interactively control said controllable application; and wherein said additional instructions for controlling said controllable application reside on said client workstation.	
14.	claim 13 of the '906 patent	computer readable program code for causing said client workstation to issue from the client workstation, one or more commands to the network server;	Figs. 4-6; 1:45-60; 5:24-5:38; 6:50-7:6; 8:36-12:50
15.	claim 9 of the '906 patent		
16.	claim 10 of the '906 patent		
17.	claim 13 of the '906 patent	computer readable program code for causing said network server to execute one or more instructions in response to said commands;	Figs. 4-6; 1:45-60; 5:24-5:38; 6:50-7:6; 8:36-12:50
18.	claim 9 of the '906 patent		
19.	claim 10 of the '906 patent		
20.	claim 13 of the '906 patent	computer readable program code for causing said network server to send information to said client workstation in response to said executed instructions;	Figs. 4-6; 1:45-60; 5:24-5:38; 6:50-7:6; 8:36-12:50
21.	claim 9 of the '906 patent		
22.	claim 10 of the '906 patent		
23.	claim 13 of the '906 patent	computer readable program code for causing said client workstation to process said information at the client workstation to interactively control said controllable application.	Figs. 5-6, 8, 10; 6:63-7:6; 8:56-12:49; 14:64-16:7; 16:28-16:46
24.	claim 9 of the '906 patent		

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
25.	claim 10 of the '906 patent		
26.	claim 14 of the '906 patent	wherein said additional instructions for controlling said controllable application reside on said client workstation.	Figs. 5-6, 8, 10; 6:63-7:6; 8:56-12:49; 14:64-16:7; 16:28-16:46
27.	claim 9 of the '906 patent	computer readable program code for causing said client workstation to utilize said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server, wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation, wherein said first distributed hypermedia document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location of at least a portion of an object external to the first distributed hypermedia document, wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and wherein said embed text format is parsed by said browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable interactive processing of said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window; wherein said executable application is a controllable application and further comprising: computer readable program code for causing said client workstation to interactively control said controllable application of said client workstation via inter-process communications between said browser and said controllable application; wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched; and wherein additional instructions for controlling said controllable application reside on said network server, wherein said computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation includes: computer readable program code for causing said client workstation to issue, from the client workstation, one or	Figs. 4-8, 10; 6:63-7:6; 8:36-16:46

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
		<p>more commands to the network server; computer readable program code for causing said network server to execute one or more instructions in response to said commands; computer readable program code for causing said network server to send information to said client workstation in response to said executed instructions; and computer readable program code for causing said client workstation to process said information at the client workstation to interactively control said controllable application.</p>	
28.	claim 16 of the '985 patent	<p>software comprising computer executable instructions . . . and when the software is executed operable to: receive, at the client workstation from the network server over the network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; cause the client workstation to utilize the browser to: respond to text formats to initiate processing specified by the text formats; display at least a portion of the document within the browser-controlled window; identify an embed text format corresponding to a first location in the document, the embed text format specifying the location of at least a portion of an object external to the file, with the object having type information associated with it; utilize the type information to identify and locate an executable application external to the file; and automatically invoke the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window.</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46
29.	claim 17 of the '985 patent	<p>claim 16 where: the information to enable comprises text formats.</p> <p>The method of claim 20 where: the information to enable comprises text formats.</p>	Figs. 7-8; 12:31-37; 12:51-13:2; 13:36-16:7.
30.	claim 21 of the '985 patent	<p>The method of claim 24 where: the information to enable comprises text formats.</p> <p>The method of claim 28 where: the information to enable comprises</p>	

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
31.	claim 25 of the'985 patent	text formats.	
32.	claim 29 of the'985 patent	The method of claim 32 where: the information to enable comprises text formats.	
33.	claim 33 of the'985 patent	The method of claim 40 where: the information to enable comprises text formats.	
34.	claim 41 of the'985 patent		
35.	claim 18 of the'985 patent	claim 17 where: the text formats are HTML tags.	Figs. 7-8; 12:31-37; 12:51-13:2; 13:36-16:7
		The method of claim 21 where: the text formats are HTML tags.	
36.	claim 22 of the'985 patent	The method of claim 25 where: the text formats are HTML tags.	
		The method of claim 29 where: the text formats are HTML tags.	
37.	claim 26 of the'985 patent	The method of claim 33 where: the text formats are HTML tags.	
		The method of claim 41 where: the text formats are HTML tags.	
38.	claim 30 of the'985 patent		
39.	claim 34 of the'985 patent		
40.	claim 42 of the'985 patent		
41.	claim 19 of the'985 patent	claim 16 where: the information contained in the file received comprises at least one embed text format.	Figs. 7-8; 12:31-37; 12:51-13:2; 13:36-16:7
		The method of claim 20 where: the information contained in the file received comprises at least one embed text format.	
42.	claim 23 of the'985 patent	The method of claim 24 where: the information contained in the file	

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
43.	claim 27 of the'985 patent	<p>received comprises at least one embed text format.</p> <p>The method of claim 28 where: the information contained in the file received comprises at least one embed text format.</p>	
44.	claim 31 of the'985 patent	<p>The method of claim 32 where: the information contained in the file received comprises at least one embed text format.</p>	
45.	claim 35 of the'985 patent	<p>The method of claim 40 where: the information contained in the file received comprises at least one embed text format.</p>	
46.	claim 43 of the'985 patent		
47.	claim 20 of the'985 patent	<p>communicating via the network server with at least one client workstation over said network in order to cause said client workstation to: receive, over said network environment from said server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; execute, at said client workstation, a browser application, with the browser application: responding to text formats to initiate</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46
48.	claim 24 of the'985 patent	<p>A method for running an executable application in a computer network environment . . . the method comprising: enabling an end-user to directly interact with an object by utilizing said executable application to interactively process said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser-controlled window, wherein said network environment is a distributed hypermedia environment, wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia document within said browser-controlled window, wherein said executable application is external to said file, wherein said client workstation executes the browser application, with the browser application responding to text formats to initiate processing specified by the text formats, wherein</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
		<p>at least said portion of the document is displayed within the browser-controlled window, wherein an embed text format which corresponds to said first location in the document is identified by the browser, wherein the embed text format specifies the location of at least a portion of said object external to the file, wherein the object has type information associated with it, wherein the type information is utilized by the browser to identify and locate said executable application, and wherein the executable application is automatically</p>	
49.	Claim 28 of the '985 patent	<p>software comprising an executable application . . . operable to: cause the client workstation to display an object and enable an end-user to directly interact with said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser controlled window, wherein said network environment is a distributed hypermedia environment, wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia document within said browser-controlled window, wherein said executable application is external to said file, wherein said client workstation executes said browser application, with the browser application responding to text formats to initiate processing specified by the text formats, wherein at least said portion of the document is displayed within the browser-controlled window, wherein an embed text format which corresponds to said first location in the document is identified by the browser, wherein the embed text format specifies the location of at least a portion of said object external to the file, wherein the object has type information associated with it, wherein the type information is utilized by the browser to identify and locate said executable application, and wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format.</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46
50.	claim 32 of the '985 patent	<p>communicating via a network server with at least one client workstation over said computer network environment in order to cause said client workstation to: receive at said client workstation, over said computer network environment from said server, at least</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
		<p>one file containing information to enable a browser application to display, on said client workstation, at least a portion of a distributed hypermedia document within a browser-controlled window; utilize an executable application external to said file to enable an end-user to directly interact with an object while the object is being displayed within a display area created at a first location within the portion of the distributed hypermedia document being displayed in the browser-controlled window, with said network server coupled to said computer network environment, wherein said computer network environment has at least said client workstation and said network server coupled to the computer network environment, wherein said computer network environment is a distributed hypermedia environment, wherein said client workstation executes the browser application, with the browser application responding to text formats to initiate processing specified by the text formats, wherein at least said portion of the document is displayed within the browser-controlled window, wherein an embed text format which corresponds to said first location in the document is identified by the browser, wherein the embed text format specifies the location of at least a portion of said object external to the file, wherein the object has type information associated with it, wherein the type information is utilized by the browser to identify and locate said executable application, and wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format.</p>	
51.	claim 40 of the '985 patent	<p>communicating via the network server with at least one remote client workstation over said computer network environment in order to cause said client workstation to: receive, over said computer network environment from the network server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; execute, at said client workstation, a browser application, with the browser application: responding to text formats to initiate processing specified by the text formats; displaying, on said client workstation, at least a portion of the document within the browser-controlled window; identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object; identifying and locating an executable</p>	Figs. 4-8, 10; 8:20-10:62; 12:51-16:7; 16:28-46

No.	Claim	Term Allegedly Governed by 112(6)	Corresponding Structure
		<p>application associated with the object; and automatically invoking the executable application, in response to the identifying of the embed text format, in order to enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window, wherein the executable application is part of a distributed application, and wherein at least a portion of the distributed application is for execution on the network server.</p>	

MICROSOFT PRESS®

COMPUTER DICTIONARY



®

PUBLISHED BY

Microsoft Press

A Division of Microsoft Corporation

One Microsoft Way

Redmond, Washington 98052-6399

Copyright © 1991 by Microsoft Press, a division of Microsoft Corporation.

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Library of Congress Cataloging-in-Publication Data

Microsoft Press computer dictionary : the comprehensive standard for business, school, library, and home.

p. cm.

ISBN 1-55615-231-0

1. Computers--Dictionaries. 2. Microcomputers--Dictionaries.

I. Microsoft Press.

QA76.15.M54 1991

004.16'03--dc20

91-9904

CIP

Printed and bound in the United States of America.

4 5 6 7 8 9 MLML 6 5 4 3 2

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

Distributed to the book trade outside the United States and Canada by Penguin Books Ltd.

Penguin Books Ltd., Harmondsworth, Middlesex, England

Penguin Books Australia Ltd., Ringwood, Victoria, Australia

Penguin Books N.Z. Ltd., 182-190 Wairau Road, Auckland 10, New Zealand

British Cataloging-in-Publication Data available.

Acquisitions Editor: Marjorie Schlaikjer

Project Editor: Mary Ann Jones

Technical Editors: David Rygmyr, Jeff Hinsch, Mary DeJong, Dail Magee, Jr.

Manuscript Editor: Pamela Beason

Copy Editor: Alice Copp Smith



fiche See microfiche.

field A location in a record in which a particular type of data is stored. For example, EMPLOYEE-RECORD might contain fields to store Last-Name, First-Name, Address, City, State, Zip-Code, Hire-Date, Current-Salary, Pay-Grade, Title, Department, Last-Increase-Date, and so on.

Individual fields have their own specifications as to maximum length and the type of data (for example, alphabetic, numeric, or financial) that can be placed in them. The facility for creating these specifications usually is contained in the data definition language (DDL).

In relational database management systems, fields are called columns.

field-effect transistor See FET.

field-programmable logic array Abbreviated FPLA; also known as programmable logic array (PLA). An integrated circuit containing an array of logic circuits in which the connections between the individual circuits, and thus the logic functions of the array, can be programmed after manufacture, typically at the time of installation (in the field). The programming can be performed only once, and it is typically done by passing high current through fusible links on the chip.

field separator Any character that separates one field of data from another. See also delimiter.

FIFO See first in, first out.

fifth-generation computer See computer.

fifth normal form (5NF) See normal form.

file A complete, named collection of information, such as a program, a set of data used by a program, or a user-created document. A file is the basic unit of storage that enables a computer to distinguish one set of information from another. A file might or might not be stored in human-readable form, but it is still the "glue" that binds a conglomeration of instructions, numbers, words, or images into a coherent unit that a user can retrieve, change, delete, save, or send to an output device.

file allocation table A table or list maintained by some operating systems to keep track of the status of various segments of disk space used for file storage. Files on a disk are stored, as space allows, in fixed-size groups of bytes (characters) rather

than from beginning to end as neat, continuous strings of text or numbers. A single file can thus be scattered in pieces over many separate storage areas. A file allocation table enables the operating system to maintain a "map" of available disk storage space so that it can mark flawed segments that should not be used and can find and link the pieces of a file. In the MS-DOS operating system, the file allocation table is commonly known as the FAT (pronounced "fat").

file attribute A restrictive label attached to a file that describes and regulates its use—for example, hidden, system, read-only, archive, and so forth. In MS-DOS, this information is stored as part of the file's directory entry.

file backup See backup.

file compression The process of reducing the storage space required for a file. See also data compression.

file control block Abbreviated FCB. A small block of memory temporarily assigned by a computer's operating system to hold information about a file that has been opened for use. A file control block typically contains such information as the file's identification, its location on disk, and a pointer that marks the user's current (or last) position in the file.

file conversion The process of transforming the data in a file from one format to another without altering the meaning of its contents—for example, converting a file from a word processor's format into its ASCII equivalent.

file extension See extension.

file extent See extent.

file format The structure of a file that defines the way it is stored and laid out on the screen or in print. The format can be fairly simple and common, as are files stored as "plain" ASCII text, or it can be quite complex and include various types of control instructions and codes used by programs and by printers and other devices. Examples include RTF (Rich Text Format), DCA (Document Content Architecture), PICT, DIF (Data Interchange Format), DXF, TIFF (Tag Image File Format), and EPSF (Encapsulated PostScript Format).

file fragmentation A condition in which files are

broken
rated se
quence
crowde
blocks
fragmen
it can e
the dis
mented
redistrib

In a
not stor
accumu
Most d:
grams
ciency
occupie

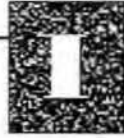
file gap
file han
ber) tha
or, som
radio of

file-han
sist in
files. M
handlin
comple
are ofte

file head
file lay
records
record
layout.

file libr
maintai
cess to :

file mai
ing, ins
altering
copying
nal to e
from th
databas
format
forming
file man



inverted list A method for creating alternative locators for sets of information. For example, in a file containing data about cars, records 3, 7, 19, 24, and 32 might contain the value "Red" in the field COLOR. An inverted list (or index) on the field COLOR would contain a record for "Red" followed by the locator numbers 3, 7, 19, 24, and 32. *Compare* linked list.

inverted list database A database similar to a relational database but with the following differences:

- The rows (records or tuples) of an inverted list table are ordered in a specific physical sequence, independent of any orderings that may be imposed by means of indexes.
- The total database can also be ordered, with specified logical merge criteria being imposed between tables.
- Any number of search keys, either simple or composite, can be defined. Unlike the keys of a relational system, these search keys are arbitrary fields or combinations of fields.
- No integrity or uniqueness constraints are enforced.
- Neither the indexes nor the tables are transparent to the user, as they would normally be in a relational system.

Because of these differences, it is much more difficult for the database management system to assure data consistency, integrity, and security with an inverted list database than with a relational system.

inverted structure A file structure in which record keys are stored and manipulated separately from the records themselves.

inverter In electronics, a logic circuit that inverts (reverses) the signal input to it—for example, inverting a high input to a low output. An inverter is also a device that converts direct current (DC) into alternating current (AC).

invoke To call or activate; used in reference to commands and subroutines.

I/O *See* input/output.

I/O-bound *See* input/output-bound.

ion-deposition printer An electrophotographic page printer similar to a laser printer but based on a more expensive technology. These printers, used mainly in high-volume data-processing environments, typically operate at speeds from 30 to 90 pages per minute. Like other electrophotographic printers, ion-deposition models use an electrostatically charged drum. Rather than converting some form of light into an electrostatic charge, however, as laser, LED, and LCD printers do, ion-deposition printers charge the drum by applying an ion stream directly to it. Ion-deposition printers typically use a method of fusing toner to paper that is fast and does not require heat. This method leaves the paper a little glossy, however, making it unsuitable for business correspondence. In addition, ion-deposition printers tend to produce thick, slightly fuzzy characters. *Compare* laser printer, LCD printer, LED printer; *see also* electrophotographic printers, nonimpact printer, page printer.

IO.SYS One of two hidden system files installed on an MS-DOS startup disk. IO.SYS in IBM releases of MS-DOS (called IBMBIO.COM) contains device drivers for peripherals such as the display, keyboard, floppy- and hard-disk drives, serial port, and real-time clock. *See also* MSDOS.COM.

IPC *See* interprocess communication.

IPL *See* initial program load.

IR *See* infrared.

IRG *See* inter-record gap.

IRGB Acronym for Intensity Red Green Blue, a type of color encoding originally used in IBM's Color/Graphics Adapter (CGA) and continued in the EGA (Enhanced Graphics Adapter) and VGA (Video Graphics Array). The standard 3-bit RGB color encoding (specifying eight colors) is supplemented by a fourth bit (called Intensity) that uniformly increases the intensity of the red, green, and blue signals, resulting in a total of 16 colors. *See also* RGB.

ISA Abbreviation for Industry Standard Architecture. An unofficial designation for the bus design of the IBM PC/XT, which allows various adapters to be added to the system by means of inserting plug-in cards into expansion slots. Originally introduced with an 8-bit data path, ISA was expanded in 1984,

when IBM introduced data path. Compared slots themselves or 16-bit slots. two separate expansion so that a single An 8-bit expansion in a 16-bit slot slots), but a 16-bit in an 8-bit slot Architecture.

ISAM *See* indexed **ISDN** Abbreviated Network, a work work evolving The goal of telephone lines, versions, with transmission ranging from music, and video types of communication which carries a second), and a formation at other devices simple, stand-alone (possibly the ISDN is a more extensive **ISO** Abbreviated Standardization International Standardization which is representative organization—The ISO work communication primary among accepted ISO/C the interactive communications **ISO 9660** An ROM, adopted for Standardization



intosh computers. Information about the configuration of the system is stored in parameter RAM. *See also* CMOS RAM.

parent/child A term describing a relationship between processes in a multitasking environment in which the parent process calls the child process and most often suspends its own operation until the child process aborts or is completed.

Also, a relationship between nodes in a tree data structure in which the parent is one step closer to the root (that is, is one level higher) than the child.

parity The quality of sameness or equivalence. With computers, parity usually refers to an error-checking procedure in which the number of 1's must always be the same—either even or odd—for each group of bits transmitted without error. If parity is checked on a per-character basis, the method is called vertical redundancy checking, or VRC; if checked on a block-by-block basis, the method is called longitudinal redundancy checking, or LRC. Parity is used for checking data transferred within a computer or between computers.

In typical modem-to-modem microcomputer communications, parity is one of the parameters that must be agreed upon by sending and receiving parties before transmission can take place. The following types of parity are used:

Type	Description
Even parity	The number of 1's in each successfully transmitted set of bits must be an even number.
Odd parity	The number of 1's in each successfully transmitted set of bits must be an odd number.
No parity	No parity bit is used.
Space parity	A parity bit is used and is always set to 0.
Mark parity	A parity bit is used and is always set to 1.

See also parity bit, parity check, parity error.

parity bit An extra bit used in checking for errors in groups of data bits transferred within or between computer systems. With microcomputers, the term is frequently encountered in modem-to-modem

microcomputer communications, in which a parity bit is often used to check the accuracy with which characters are transmitted. In parity checking of this type, the sending computer adds a parity bit to each group of data bits, each of which represents a single character. The setting of the parity bit depends on the type of parity used. With even parity, for example, the parity bit is set to 1 whenever it is needed to make the total number of 1's (data bits plus parity bit) an even number; with odd parity, the parity bit is set to 1 whenever it is needed to make the total number of 1's an odd number. The receiving device counts the number of 1's in each arriving group of data and parity bits; if the number is odd when it should be even, or vice versa, the device can assume that one of the bits was transmitted incorrectly and that an error occurred.

parity check The use of parity to check the accuracy of transmitted data. *See also* parity, parity bit.

parity error An error in parity that indicates an error in transmitted data. *See also* parity, parity bit.

park To position the read/write head over a portion of a disk that stores no data (and therefore can never be damaged) or beyond the surface of the disk, prior to shutting down the drive, especially in preparation for moving it. Parking can be performed manually, automatically, or (as is typical) by a disk utility program.

parse To break input into smaller chunks so that a program can act upon the information. Compilers have parsers for translating the commands and structures entered by a programmer into machine language. A natural-language parser accepts text in a human language such as English, attempts to determine its sequence structure, and translates its terms into a form the program can use. Database management programs and expert systems often support natural-language parsing. A user could ask such a program to *display the relationship between inflation and home buying in the last decade*; the program might break the sentence apart and interpret it in the following way:

<i>display</i>	Present the results as a chart.
<i>the relationship between</i>	Do a linear regression analysis.



*inflation and
home buying*

The independent and
dependent variables,
respectively.

in the last decade

Use data from 1980–89.

See also natural language.

partition A logically distinct portion of memory or a storage device that functions as though it were a physically separate unit. The MS-DOS operating system, for example, can divide a hard disk into a primary partition and an extended DOS partition, each of which behaves as if it were physically distinct from the other.

In database programming, a subset of a database table or file. In a horizontal partition, data is separated by rows or records; in a relational database management system, horizontal partitions can usually be created based on primary key values. In a vertical partition, data is separated by columns or fields. Vertical partitioning (or projection of columns) often occurs during normalization of database design. Database files may be horizontally or vertically partitioned across multiple nodes in a distributed database.

Pascal A concise procedural language, designed 1967–71 by Niklaus Wirth. Pascal, a compiled, structured language, built upon ALGOL, simplifies syntax while adding data types and structures such as subranges, enumerated data types, files, records, and sets. Acceptance and use of Pascal exploded with Borland International's introduction in 1984 of Turbo Pascal, a high-speed, low-cost Pascal compiler for MS-DOS systems that has sold over a million copies in its various versions. Even so, Pascal appears to be losing ground to C as a standard development language on microcomputers. See also ALGOL, C, compiled language.

pass In programming, the carrying out of one complete sequence of events—for example, one pass through a program loop ("WHILE *x* is less than 10, DO this") or one scan of a program by a compiler or an assembler (in preparation for converting program instructions into a form the computer can carry out).

In another sense, to forward a piece of data from one part of a program to another. See also pass by address, pass by value.

pass by address Also called pass by reference. A means of passing an argument or parameter to a subroutine. The calling routine passes the address (memory location) of the parameter to the called routine, which can then use that address to retrieve or modify the value of the parameter. Compare pass by value; see also argument, call.

pass by reference See pass by address.

pass by value A means of passing an argument or parameter to a subroutine. A copy of the value of the argument is created and passed to the called routine. When this method is used, the called routine can modify the copy of the argument but it cannot modify the original argument. Compare pass by address; see also argument, call.

password A security measure used to restrict access to computer systems and sensitive files. A password is a unique string of characters that a user types in as an identification code. The system compares the code against a stored list of authorized passwords and users. If the code is legitimate, the system allows the user access, at whatever security level has been approved for the owner of the password.

password protection The use of passwords as a means of allowing only authorized users access to a computer system or its files.

paste To insert text or a graphic that has been cut or copied from one document into a different location in the same or a different document. See also cut and paste.

patch In programming, to repair a deficiency in the functionality of an existing routine or program, generally in response to an unforeseen need or set of operating circumstances. Patching does not necessarily imply sloppiness in implementing a solution to a problem: Patching is a common means of adding a feature or a function to an existing version of a program until the next version of the software, which presumably will have that feature or function included in its design, is released. Compare hack, kludge.

path A route from one point to another. In communications, a path is a link between two nodes (stations) in a network. In other contexts, a path is a route through a structured collection of informa-

tion, as in a disk. In a data structure in a tree to a

In programming instructions a routine. In theory under path is a log tree of infer leading to a

In file system the operati retrieving fil for example called access unique numl an open file that used by is the course drive, such a clude any n the maximu the system ()

In graph segments or text. The illu on a copy of

Path



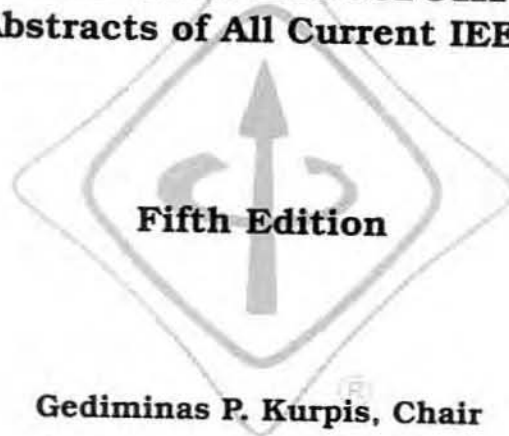
The qual

Path.

pathname is of the direct rent directo pathname \, the file myf

IEEE Std 100-1992

**The New IEEE Standard Dictionary
of Electrical and Electronics Terms**
[Including Abstracts of All Current IEEE Standards]



Gediminas P. Kurpis, Chair

Christopher J. Booth, Editor

IEEE

cont



IEEE

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1993 by the
Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1993
Printed in the United States of America

ISBN 1-55937-240-0

*No part of this publication may be reproduced in any form,
in an electronic retrieval system or otherwise,
without the prior written permission of the publisher.*

IE
mitte
and t
Mem
They
oped
ject v
have
dard.

Use
Stand
sure,
scope
time &
throug
users
every
than f
that it
presen
have tl

Com
ested p
for cha
text, to

Inter
of port
need fo
will ini
represe
that an
interest
tees are
except i
siderati

Conn
address

IEEE
and Ele
involve
not ass
obligati

filament power supply (electron tube). The means for supplying power to the filament. See: **power pack.** [45]

filament voltage. The voltage between the terminals of a filament. See: **electrode voltage (electron tube); electronic controller.** [119]

file (1) (computing systems). A collection of related records treated as a unit. Note: Thus in inventory control, one line of an invoice forms an item, a complete invoice forms a record, and the complete set of such records forms a file. [20], [85]

(2) (data management) (software). A set of related records treated as a unit. For example, in stock control, a file could consist of a set of invoice records. See also: **data file; data set; logical file.** 610.5-1990, 610.12-1990

(3) (information transfer). One named collection of data. 949-1985w

(4) (microprocessor operating systems). A set of related records usually treated as a named unit of storage. 855-1985

file access mode. The type of access allowed for a given file and a given user. For example, the file access mode for a given file might be read-only access for one user, and read/write access for another. Syn: **access type.** 610.5-1990

file attribute. A property, feature, or characteristic of a file. 610.5-1990

file cleanup. The removal of superfluous data from a file. Syn: **file tidying.** 610.5-1990

file directory. (A) A list of files and their locations within a computer system. See also: **catalog.** (B) A list of the files and their locations on a particular storage device or volume. 610.5-1990

file gap (1) (computing system). An area on a storage medium, such as tape, used to indicate the end of a file. [20], [85]

(2) (data management). An unused area on a data medium between the end of one file or group of data and the beginning of another file or group of data. 610.5-1990

file layout. The arrangement and structure of data in a file. Syn: **file organization.** 610.5-1990

file-locking. See: **lock.** 610.5-1990

file maintenance (1) (computing systems). The activity of keeping a file up to date by adding, changing, or deleting data. [20], [85]

(2) (data management). The activity of adding, changing, or deleting data in a file as needed. 610.5-1990

file name. (A) One or more characters used to identify a file. (B) A name associated with a set of file data or output data. 610.5-1990

file organization. (A) See: **file layout.** (B) The order of physical records within a file that determines the access method to be implemented in order to use the file. 610.5-1990

file processing (computer applications). The periodic updating of one or more master files to reflect the effects of current data, often from a transaction file. For example, a monthly run updating the inventory file. 610.2-1987

file tidying. See: **file cleanup.** 610.5-1990

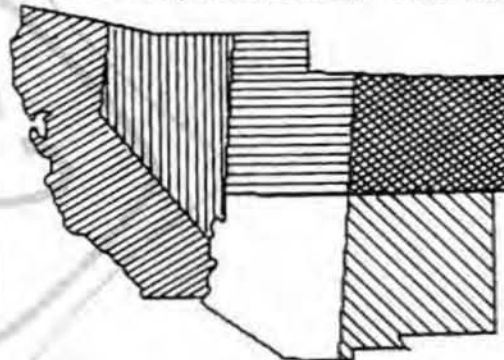
filiform corrosion. See: **underfilm corrosion.**

fill (1) (computer graphics). To insert a color, pattern, or hatch into a closed polygon or area bounded by lines or curves. Syn: **area fill; polygon fill.** 610.6-1991

(2) (data management). See: **character fill; filler character; zero fill.** 610.5-1990

(3) (token ring access method). A bit sequence that may be either 0 bits, 1 bits, or any combination thereof. 802.5-1989

fill area. A display element that consists of a closed polygon that is hollow or filled with a uniform color, pattern, or hatch. 610.6-1991



Fill Area

fill area attribute. A characteristic of a filled region. For example, color index, interior style. 610.6-1991

filled-core annular conductor. A conductor composed of a plurality of conducting elements disposed around a nonconducting supporting material that substantially fills the space enclosed by the conducting elements. See: **conductor.** [10]

filled-system thermometer. An all-metal assembly consisting of a bulb, capillary tube, and Bourdon tube (bellows and diaphragms are also used) containing a temperature-responsive fill. A mechanical device associated with the Bourdon is designed to provide an indication or record of temperature. See: **Bourdon.** 119-1974

filled tape. Fabric tape that has been thoroughly filled with a rubber or synthetic compound, but not necessarily finished on either side with this compound. See: **conductor.** [10]

filler (filler strip) (1) (rotating machinery). Additional insulating material used to insure a tight depth-wise fit in the slot. See: **rotor (rotating machinery); stator.** [9]

(2) (of a the l (3) item that exan whic ADD plac to be pad;

filler c an a leg fill or to th issue does unwa field.

filler st

filling dielec into t requir filling insulk

filling f of ent plana cross (A) micros

$$\epsilon_r \text{ eff} = \frac{\epsilon_r}{\epsilon_r - 1}$$

where $\epsilon_r = \text{rel}$ of a mi $\epsilon_r \text{ eff} =$

(B) diele

$$\frac{\tan \delta_{ef}}{\tan \delta}$$

where $\tan \delta =$ of a mi $\tan \delta_{ef}$ line

(C) mag

$$\frac{1}{\mu_{eff}} - 1$$

$$\frac{1}{\mu} - 1$$

where