

Internal distribution code:

- (A) [-] Publication in OJ
- (B) [-] To Chairmen and Members
- (C) [-] To Chairmen
- (D) [X] No distribution

**Datasheet for the decision
of 5 December 2018**

Case Number: T 2567/16 - 3.5.06

Application Number: 06705012.0

Publication Number: 1875356

IPC: G06F9/445

Language of the proceedings: EN

Title of invention:

METHOD FOR DISTRIBUTING COMPUTING BETWEEN SERVER AND CLIENT

Applicant:

Airscape Technology Pty. Ltd.

Headword:

Client recursively transforming itself/AIRSCAPE

Relevant legal provisions:

EPC 1973 Art. 84

Keyword:

Claims - clarity (no)

Decisions cited:

Catchword:



Beschwerdekammern
Boards of Appeal
Chambres de recours

Boards of Appeal of the
European Patent Office
Richard-Reitzner-Allee 8
85540 Haar
GERMANY
Tel. +49 (0)89 2399-0
Fax +49 (0)89 2399-4465

Case Number: T 2567/16 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 5 December 2018

Appellant: Airscape Technology Pty. Ltd.
(Applicant) Level
4, 6 Bridge Street
Sydney NSW 2000 (AU)

Representative: Anderson, Oliver Ben
Venner Shipley LLP
200 Aldersgate
London EC1A 4HD (GB)

Decision under appeal: **Decision of the Examining Division of the
European Patent Office posted on 3 June 2016
refusing European patent application No.
06705012.0 pursuant to Article 97(2) EPC.**

Composition of the Board:

Chairman W. Sekretaruk
Members: M. Müller
G. Zucka

Summary of Facts and Submissions

- I. The appeal is against the decision of the examining division, with reasons dispatched on 3 June 2016, refusing European patent application No. 06 705 012 for lack of novelty or inventive step.
- II. Notice of appeal was filed on 15 August 2016, the appeal fee being paid on the same day. A statement of grounds of appeal was received on 13 October 2016. The appellant requested that the decision be set aside and that a patent be granted on the basis of claims 1-21 according to a main or two auxiliary requests as filed with the grounds of appeal.
- III. In an annex to a summons to oral proceedings, the board informed the appellant of its preliminary opinion that the claimed invention did not comply with Articles 83 and 84 EPC 1973. Some tentative remarks on inventive step were also made, pursuant to Article 56 EPC 1973.
- IV. In response to the summons, the appellant did not submit any arguments or amendments but informed the board, with a letter dated 27 November 2018, that it would not attend or be represented at the oral proceedings. The oral proceedings were then cancelled.
- V. Claim 1 of the main request reads as follows:

"A method for use in running a display-based computer application, the application comprising a plurality of application segments, an application segment consisting of a set of dynamically generated displays, the method involving distributing computing between a server on a server computer and a client on a client computer, the server and client computers operatively interconnected

by way of a computer network, characterised in that the method comprises the steps of:

(a) providing, by the server computer, a generic client engine to the client computer, the client engine comprising an application manager, one or more display managers and one or more load managers;

(b) providing a collection of individual application subset definition files each relating to one of said application segments and defining component or layout characteristics within the one or more displays associated with the respective application segment;

(c) wherein, on receipt of successive application subset definition files from the server, the client recursively transforms itself under instruction from the application subset definition files to provide the respective application segments."

Claim 1 of auxiliary request 1 is identical to that of the main request, except that step (c) is now labelled "(d)" and a new step © has been inserted which reads as follows:

"... (c) enabling a user to navigate between application segments using dynamically generated logical pathways activated via the displays of the application segments, wherein the dynamically generated logical pathways are generated based upon information, stored in a current application subset definition file, that is transferred to the server, wherein, in response to receiving the information, the server employs business logic running on the server to serve a next application subset definition file to the client device; ..."

Claim 1 of auxiliary request 2 is identical to that of the main request, except that the following phrase has been added to the end of step (a):

"... the client engine maintaining state information between client-server requests/responses ..."

and that the following paragraph has been added to the end of step (c):

"... wherein the one or more display managers include means for data synchronisation across display subset transitions both internally within an application subset and externally across application subsets."

All requests also contain independent claims 12 and 13 for a computer program and a distributed computing system, respectively, which correspond to independent method claim 1.

Reasons for the Decision

1. The following reasons are based on the board's preliminary opinion, to which the appellant has chosen not to respond.

The invention

2. The application is concerned with web applications which require complex ("rich") user interfaces, especially if the available bandwidth is small (see page 1, lines 23-28; page 5, lines 1-3; page 12, lines 21-22).

- 2.1 It is explained that HTML, which was designed for document publication, is a bad fit for user interface functionality, in particular because HTML "require[s] a new page to be served at each user interaction" (see page 2, lines 3-7). This is referred to as the "traditional thin-client development model" (see page 2, lines 23-27). Client-side scripting based on languages such as Java Script had helped alleviate this problem by "off-load[ing] some server-side processing to the client", thereby making the client "fatter" (see page 1, lines 8-10; page 2, line 27, to page 3, line 5), but this solution was not available for user interface layouts (page 2, lines 10-11; page 7, lines 1-9; page 11, line 30, to page 12, line 4).
- 2.2 The invention proposes to address this problem by a "smart client" which "provides many of the benefits of conventional [...] fat clients" but "with the download footprint of a thin client" (see page 2, line 23, to page 3, line 7; page 4, lines 8-10; and page 5, lines 7-10). The "smart client" owes its name "to its ability to recursively transform itself to provide the application subsets that together make up" a "complete business application" (page 4, lines 8-10; and page 12, lines 15-17). Elsewhere it is disclosed that it is a "generic applet" which can "dynamically and recursively transform itself" (page 7, lines 15-18).
- 2.3 It is proposed that "business applications" be downloaded in terms of their "individual application subset definition files", specified in a mark-up language such as XML, from which the displays are "dynamically created" at the client (see page 10, last paragraph; page 11, lines 9-16; and page 12, lines 13-15).

2.4 The client engine is said to comprise a number of "managers", three of which are claimed: one application and one or more display and load managers (see page 12, line 26, to page 15, line 3). With reference to the display managers, it is disclosed that users "navigate" through the application based upon "the logical pathways between different application subsets", the "pathways" being "dynamically generated" and cached (see page 13, lines 25-30, and page 11, penultimate paragraph). The client "dynamically transforms itself in accordance with the pathways the user traverses at the client-side" (page 12, lines 15-17).

Clarity, Article 84 EPC 1973

3. The examining division interpreted the claim language broadly, stating that several of the central terms in the claims were "so generic and without substance that they are reasonably seen to be covered by the more concrete situation in" the prior art (see reasons 7.1). The appellant was dissatisfied with the decision under appeal because it was difficult to determine the precise mapping between the features of the claimed invention and those disclosed in the prior art (see the grounds of appeal, page 3, paragraphs 2 and 5; page 5, paragraphs 3 to 5). In view of this, it is particularly important to determine how the claims must be construed.

3.1 One - if not the - central feature of the invention (in claims 1 and 13 of all requests; see also point 2.2 above) is that "the client recursively transforms itself under instruction from the application subset definition files".

- 3.2 The claims refer to a "client on a client computer" and to "a generic client engine" provided "to the client computer". The board understands both the "client" and the "client engine" to denote software executing on the client computer. The claims, however, leave open whether both are the same or what, if anything, would be the difference between them. The way in which the "client engine" is limited by its qualification as "generic" is also unclear.
- 3.3 Furthermore, the claim language fails to define what it means exactly for the client to "transform[] itself". *A priori*, this phrase could mean that the *program code* or the *data* constituting the client is modified, but also merely that the *execution state* of the client software is modified.
- 3.4 The transformation is said to take place "under instruction from" and "on receipt of successive application subset definition files" so as to "provide the [...] application segments", i.e. the "dynamically generated displays".
- 3.5 The independent claims of the main request and auxiliary request 2 specify nothing about the "application subset definition files" except that they are "provid[ed]" to the client. Specifically, they do not specify when they are provided or, thus, what might trigger the self-transformation of the "client".
- 3.6 On this account, the independent claims of auxiliary request 1 contain an additional feature according to which "a next application subset definition file" is "served" by the server after having received certain "information" and, in response, "employ[ed] business logic". The "information" is claimed to be "stored in a

current application subset definition file", and to be the basis for dynamically generating "logical pathways" which are "activated via the displays of the activation segments" and along which users can "navigate between application segments".

- 3.6.1 The board considers there to be lack of clarity on what a "logical pathway" is, how a "display" can "activate" a "pathway", and what it means for a "pathway" to be "generated", Article 84 EPC 1973.
- 3.6.2 It is also unclear from the claim language what relevance the dynamically generated logical pathways and the user navigation have for the determination of the "next application subset file". Although it is the same information, "stored in a current application subset definition file", which is the basis for the dynamic generation of logic pathways and which is transferred to the server, the claims do not require the server to be informed about the dynamically generated logical pathways or the user navigation. The additional feature is unclear for that reason, too. The description, in contrast, specifies the intention that the client "dynamically transform itself in accordance with the pathway the user traverses at the client-side" (see page 12, lines 15-17).
- 3.6.3 Literally, the additional feature only implies that the server produces "next application subset definition files" based on "current" ones, which is insufficient to overcome the clarity objection to the "application subset definition files" (see point 3.5 above).
- 3.6.4 Moreover, the additional feature does not interact with the "recursive self-transformation" feature (see point 3.3 above) and therefore cannot clarify it. For

example, even if the "displays" were defined in HTML alone, so that every new page had to be generated and transmitted separately (see the description, page 2, paragraph 2) and no client-side execution state and/or client "transformation" were available, it would be the "business logic" at the server that would determine the pages to be generated and sent in response to certain user actions.

- 3.7 In summary, the claims fail to define what the recursive self-transformation feature is meant to achieve and how it is implemented. As a consequence, the independent claims of all requests are unclear, Article 84 EPC 1973.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:

The Chairman:



L. Stridde

W. Sekretaruk

Decision electronically authenticated