

Internal distribution code:

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

**Datasheet for the decision
of 11 May 2017**

Case Number: T 0728/12 - 3.5.06

Application Number: 05752104.9

Publication Number: 1769349

IPC: G06F9/46

Language of the proceedings: EN

Title of invention:

System and method for interfacing an application to a distributed transaction coordinator

Applicant:

Computer Associates Think, Inc.

Headword:

Distributed transactions/COMPUTER ASSOCIATES THINK

Relevant legal provisions:

EPC 1973 Art. 84

Keyword:

Claims - clarity (no)

Decisions cited:

Catchword:



Beschwerdekammern
Boards of Appeal
Chambres de recours

European Patent Office
D-80298 MUNICH
GERMANY
Tel. +49 (0) 89 2399-0
Fax +49 (0) 89 2399-4465

Case Number: T 0728/12 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 11 May 2017

Appellant: Computer Associates Think, Inc.
(Applicant) One Computer Associates Plaza
Islandia, New York 11749 (US)

Representative: Dunlop, Hugh Christopher
Maucher Jenkins
26 Caxton Street
London SW1H 0RJ (GB)

Decision under appeal: **Decision of the Examining Division of the**
European Patent Office posted on 6 October 2011
refusing European patent application No.
05752104.9 pursuant to Article 97(2) EPC.

Composition of the Board:

Chairman W. Sekretaruk
Members: R. de Man
M. Müller

Summary of Facts and Submissions

- I. The applicant (appellant) appealed against the decision of the Examining Division refusing European patent application No. 05752104.9.
- II. The Examining Division decided that the then main request and first and second auxiliary requests did not meet the clarity requirement of Article 84 EPC. In *obiter dicta*, it gave reasons why the subject-matter of the independent claims of all requests lacked inventive step.
- III. With the statement of grounds of appeal, the appellant submitted a main request and first and second auxiliary requests corresponding to the requests considered in the contested decision with minor amendments.
- IV. In a communication accompanying a summons to oral proceedings, the board expressed the preliminary view that the requests on file did not comply with the requirements of clarity and inventive step.
- V. In a letter dated 11 April 2017, the appellant replaced its requests with an amended "replacement main request" based on the previous second auxiliary request.
- VI. Oral proceedings were held on 11 May 2017. At the end of the oral proceedings, the chairman announced the board's decision.
- VII. The appellant requests that the decision under appeal be set aside and that a patent be granted on the basis of the claims of the replacement main request.

VIII. Claim 1 of the replacement main request reads as follows:

"A computer readable medium storing computer executable instructions thereon for managing distributed transactions, the instructions when executed on one or more processors configuring the one or more processors to:

provide, by an engine (122), an interface between a client application (120) operating in a first runtime environment (116) and a distributed transaction coordinator (126) operating in a second runtime environment (118) different from the first runtime environment (116), wherein the interface is operable to modify or convert calls and parameters between the first runtime environment (116) and the second runtime environment (118) and to generate operation mechanisms to interface the communication and exchange of data between the first and second runtime environments (116,118) for compatibility of the execution threads, control blocks, and other mechanisms of the first runtime environment used by the distributed transaction coordinator (126) during a distributed transaction with the second runtime environment;

receive, by the engine (122), a distributed transaction request from the client application (120);

in response to the request, enlist, by the engine (122), the client application (120) with the distributed transaction coordinator (126) using the interface;

cause, by the engine (122), a first resource manager (104a) to enlist in the distributed transaction with the distributed transaction coordinator (126);

cause, by the engine (122), a second resource manager (104b) to enlist in the distributed transaction with the distributed transaction coordinator (126);

lock, by the distributed transaction coordinator (126), connections between the distributed transaction coordinator (126) and each of the first and second resource managers (104) to prevent any call pertaining to the distributed transaction to the first and second resource managers (104) until enlistment of the first and second resource managers (104) with the distributed transaction coordinator (126) is complete; and

unlock, by the distributed transaction coordinator (126), connections between the distributed transaction coordinator (126) and each of the first and second resource managers (104) after the first resource manager (104a) and the second resource manager (104b) are enlisted with the distributed transaction coordinator (126) to permit calls pertaining to the distributed transaction to the first and second resource managers (104),

wherein the first resource manager (104a) is operable to perform a first portion of the distributed transaction and the second resource manager (104b) is operable to perform a second portion of the distributed transaction after enlisting with the distributed transaction coordinator (126)."

- IX. The appellant's arguments as relevant to this decision are discussed in detail below.

Reasons for the Decision

1. The appeal complies with the provisions referred to in Rule 101 EPC and is therefore admissible.

The invention

2. The invention is concerned with facilitating distributed transactions. The background section of the application explains that distributed transactions consist of a set of actions to be performed on multiple database systems or "resource managers" while respecting the "ACID" properties (Atomicity, Consistency, Isolation, Durability). Such transactions may be implemented using the two-phase commit protocol. In a first phase of this protocol, a transaction manager requests each resource manager involved in the transaction (referred to as "enlisted" resource managers) to prepare to commit. If all enlisted resource managers successfully prepare, then in a second phase the transaction manager broadcasts a commit decision. The detailed description and the claims refer to the transaction manager as the "distributed transaction coordinator" (DTC).

Clarity of claim 1

3. Independent claim 1 of the sole request on file is concerned with the process of initiating a distributed transaction and enlisting resource managers in the distributed transaction.

The transaction is initiated by a client application operating in a first runtime environment. The DTC operates in a second runtime environment which is "different" from the first runtime environment. To allow communication between the client application and the DTC, an "interface" is provided by an "engine".

In response to a distributed transaction request from the client application, the engine uses the interface

to enlist the client application with the DTC. It then "causes" first and second resource managers to enlist in the distributed transaction with the DTC.

4. The claim further specifies, in particular, the following features, which are amended versions of features objected to in the contested decision for lack of clarity and which are central to the appellant's inventive-step argument:

- lock, by the DTC, connections between the DTC and each of the first and second resource managers to prevent any call pertaining to the distributed transaction to the first and second resource managers until enlistment of the first and second resource managers with the DTC is complete; and
- unlock, by the DTC, connections between the DTC and each of the first and second resource managers after the first resource manager and the second resource manager are enlisted with the DTC to permit calls pertaining to the distributed transaction to the first and second resource managers.

5. These "lock" and "unlock" features are based on Figure 2 and on the passage of the description of the published application on page 14, line 31, to page 15, line 5, which reads as follows:

"A request thread for enlisting resource managers 104 is generated at step 220. Next, at step 222, the connection with resource managers 104 is locked to block the application thread until enlistment of resource managers 104 is complete. If the enlistment of resource managers 104 is not complete at decisional

step 224, then, at step 226, DTC 126 waits for a period of time. Once the enlistment of resource managers 104 is complete at decisional step 224, then, at step 228, DTC 126 unlocks the connection with resource managers 104, thereby allowing calls to resource managers 104 to start a distributed transaction."

Thus, the calls "pertaining to the distributed transaction to the first and second resource managers" referred to in the claimed "lock" and "unlock" features are calls made by "the application thread".

6. In the description, the term "the application thread" lacks an antecedent. In its communication, the board suggested that it referred to the client application. The appellant submitted that this was incorrect and that it followed from the description on page 17, lines 3 and 4, that the application thread was managed by the engine.

The board notes that in either case the application thread is not part of the DTC. Indeed, in the main embodiment of the present application depicted in Figure 1, both the client application and the engine operate in the first runtime environment, whereas the DTC operates in the second runtime environment.

7. It follows that, according to the "lock" feature of claim 1, by locking connections between itself and each of the two resource managers, the DTC prevents calls to the resource managers made by an entity which is not the DTC. In the board's view, this casts doubt on what is meant by "locking a connection". Although a connection being "locked" could be understood as meaning that calls made over the connection are blocked, in the present case the calls being blocked

are not made by either side of the connection. The appellant was not able to shed light on this issue.

8. The board concludes that the "lock" feature and the corresponding "unlock" feature render claim 1 of the sole request on file unclear within the meaning of Article 84 EPC 1973. The appeal is therefore to be dismissed.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:

The Chairman:



A. Wolinski

W. Sekretaruk

Decision electronically authenticated