

Title: On the analysis of compensation correctness

Author(s): Vaz, Catia^{1,2}; Ferreira, Carla³

Source: Journal of Logic and Algebraic Programming

Volume: 81 **Issue:** 5 **Pages:** 585-605 **DOI:** 10.1016/j.jlap.2012.04.002 **Published:** Jul 2012

Document Type: Article

Language: English

Abstract: One fundamental idea of service-oriented computing is that applications should be developed by composing already available services. Due to the long running nature of service interactions, a main challenge in service composition is ensuring correctness of transaction recovery. In this paper, we use a process calculus suitable for modelling long running transactions with a recovery mechanism based on compensations. Within this setting, we discuss and formally state correctness criteria for compensable processes compositions, assuming that each process is correct with respect to transaction recovery. Under our theory, we formally interpret self-healing compositions, that can detect and recover from faults, as correct compositions of compensable processes. Moreover, we develop an automated verification approach and we apply it to an illustrative case study. (C) 2012 Elsevier Inc. All rights reserved.

Author Keywords: Long Running Transactions; Compensations; Compensation Correctness; Failure Handling; Process Calculi; Model Checking

KeyWords Plus: Web Services; Calculus; Transactions; Semantics

Reprint Address: Vaz, C (reprint author), Inst Politecn Lisboa, DEETC, ISEL, Lisbon, Portugal.

Addresses:

1. Inst Politecn Lisboa, DEETC, ISEL, Lisbon, Portugal
2. I&D Lisboa, Inst Engn Sistemas & Comp, Lisbon, Portugal
3. Univ Nova Lisboa, CITI Dept Informat, FCT, P-1200 Lisbon, Portugal

E-mail Address: cvaz@cc.isel.ipl.pt

Funding:

Funding Agency	Grant Number
Portuguese FCT	SFRH/BD/45572/2008
FCT (INESC-ID) through the PIDDAC Program funds	

Publisher: Elsevier Science Inc

Publisher Address: 360 Park Ave South, New York, NY 10010-1710 USA

ISSN: 1567-8326

Citation: Vaz C, Ferreira C. On the analysis of compensation correctness. Journal of Logic and Algebraic Programming. 2012; 5 (81): 585-605.