Merging an Active Database and a Reflective System: Modelling a New Several Active Meta-Levels Architecture

Laure Berti

Equipe Système d'Information Multi-Média,
Groupe d'Etude du Codage de Toulon,
Université de Toulon et du Var, B.P. 132,
F-83957 La Garde cédex, France
e-mail: berti@univ-tln.fr

Abstract. The complexity of implementing, debugging and maintaining large numbers of rules in active database systems and the need for understanding the behaviour of rules in spite of their non-deterministic execution make us suggest applying the reflective systems' approach to an active database system. The control of execution cycles, rule's flow and relating transactions processing can be made dynamically and introspectively by the active system which monitors itself [2].
Beyond actual debugging, visualisation and explanation tools – DEAR [4], REFLEX Visual Supervisor [7], Sentinel Debugger [3], OLAF [10] – our aim is to maintain consistency among ECA rules and recover the DBMS when it fails during rules execution (especially during the detection of composite events [11]) with the reflection concept. In the early stages on our research, we examine which malfunctions must be taken into account as specialized internal events in the several active meta-levels architecture we propose. Each active meta-level acts only on the level immediately below itself: the root level of the active reflective database recognizes specific situations and reacts to them without explicit user or application requests by means of ECA rules. The second active meta-level of the reflective database analyses and corrects malfunctions occuring in rules execution of the first active level.
We show how debbugging of ECA rules can be dealt with, as the reflective dimension is added to the active behaviour of the database. This dimension relates to the structure, execution model and management of active rules (indissociable with the extended transaction model [1] [5]). It enables distinctly but conjointly the concise expression, detection and consummation of:
– events which happen inside the root level of the active reflective database as specific events of the application
– and events (usually malfunctions) which occur during rules execution (cascading, looping or blocking rules...).
The greatest advantage is to separate these different kinds of events (or malfunctions) taking part (or occuring) in the execution of a set of ECA rules, in order to recommand specific recovery strategies and to modify dynamically events consummation contexts and coupling modes. The future concerns of our work will focus on implementing and validating

reflective systems' principles in [9] [6] to an active database system architecture.

References

1. Barga, R., Pu, C. Reflection on a legacy transaction processing monitor. In *Proceedings of the ACM Reflection'96*, 1996.
2. Berti, L. Bases de donnés actives et systèmes réflexifs : de la convergence au fusionnement. *submitted to Inforsid'97*, 1997.
3. Chakravarthy, S., Tamizuddin, Z., Zhou, J. A visualisation and explanation tool for debugging ECA rules in active databases. In *Proceedings of the 2nd Intl. Workshop on Rules In Database Systems*, pages 197–209, 1995.
4. Diaz, O., Jaime, A., Paton, N. DEAR: a DEbugger for Active Rules in an object-oriented context. In *Proceedings of the Intl. Workshop on Rules In Database Systems*, pages 180–193, 1993.
5. Jawadi, R., Su, S. Incorporating flexible and expressive rule control in a graph-based transaction framework. Technical Report, Department of Computer and Information Sciences, University of Florida, *UF-CIS-TR-94-030*, 1994.
6. Kornman, S. Surveillance et réflexivité: présentation du système SADE. *Revue d'Intelligence Artificielle*, vol.7 (3),pages 295–327, 1993.
7. Naqvi, W., Ibrahim, M. The REFLEX active database system. Technical Report, Database Systems Research Laboratory, University of Greenwich, *TR-CIT-DB0692* , 1992.
8. Paton, N., Diaz, O., Williams, H., Campin, J., Dinn, A.,Jaime, A. Dimensions of active behaviour. In *Proceedings on Rules In Database Systems, Workshop on Computing*, pages 40–57,1994.
9. Smith,B. Reflection and semantics in a procedural langage. PhD Thesis, MIT, Computer Science Laboratory, *TR-272*, 1982.
10. Thomas, I., Jones, A. OLAF: the GOAD active database event/rule tracer. In *Proceedings of the 7th Intl. Conference of Database and EXpert systems Applications*,pages 436–445, 1996.
11. Zukunft, O. Recovering active databases. In *Proceedings of the 2nd Intl. Workshop on Rules In Database Systems*, pages 357–371,1995.