D.S. Smolko

A NOVEL APPROACH TO CONSISTENCY CHECKING OF DISTRIBUTED DOCUMENTS AND PROCESSING OF MEASUREMENT RESULTS IN A CONTROL SYSTEM

Saint Petersburg, Saint Petersburg State Technical University

Abstract. The research aims at providing an elegant and simple way to check consistency relations in a distributed setting. We seek to implement the most natural way of inconsistency identification, when information resources –documents - are checked at their location, rather than being moved across the network for centralised processing. We propose and evaluate a distributed architecture for consistency checking that makes use of software agents and mobility.

Modern control systems use multiple heterogeneous sources of information, and are vulnerable to the GIGO paradigm ("Garbage In-Garbage Out") with respect to inconsistent input data [1]. A successful control system must therefore provide sufficient input data analysis and control mechanisms and possess a capability to carry out checks of consistency relations between related data resources, most importantly between documents or data streams that contain measurement results.

A review of consistency management technologies for distributed resources has suggested that when using independent and partially redundant information sources, it is necessary to establish consistency between their content. Establishment of consistency status [4] for input data is a requisite for correct operation and usability of the output control signal of the control system, used in a decision making process.

This paper describes a proposed new approach to checking consistency relations between input information sources using mobile software agents to cope with resource distribution. The approach facilitates establishment of relations (links) between related distributed resources [6] without transmitting their complete content across the network [3]. We capitalise on locality of access to resources, mobility of the consistency checking agents, and provide an architecture, which establishes the infrastructure necessary to take advantage of access locality and component mobility [2,5]. We present our approach in a scenario, describe the implementation prototype of the agent architecture, and give qualitative and quantitative evaluation to the distributed checking architecture.

SCENARIO

In a measurement experiment, a significant amount of data is collected by sensors at distributed locations, interconnected by a network. The task of a control system is to use the distributed data as inputs, and produce the output depending on a relation or a set of relations between the inputs. Often, the measurements are not entirely independent, and there exist prior knowledge on the consistency relations that the data must satisfy in order to avoid the GIGO effect.

In our approach, measurement data is represented in a structured format, eXtensible Modeling Language (XML), in accordance with a data type definition DTD or an XML Schema. Consistency rules, which specify the required relationships between the data types [6], have to be developed for a particular application domain. The rules must be represented in a language [7] that extends XML and is based on first-order logic. Each data location will host a software agent middleware [5].

ARCHITECTURE

We have adopted an open and extensible architecture based on stationary components-agents that provide services locally at each distributed site, and mobile software agents, capitalizing on autonomy and migration capacity that effectively consume the stationary services during a consistency check of distributed data resources [5,2]. Analysis of some open problems in the field of distributed consistency management [2,3] necessitated functional decomposition of a distributed consistency check into a number of distinct agent roles that could be autonomously executed. The software agent architecture for distributed consistency checking, described in detail in [5,2,3], therefore involves multiple instances of patterns of a *domain* agent, *gateway* domain for interconnection of domains, *resource* interface agent and a mobile *consistency checking* agent (Fig.1).

Resource interface agent provides an abstraction from the complexity of heterogeneous and concurrent data access to the underlying resources, therefore enabling integration of the architecture with the existing systems. The agent monitors all changes in the measurement data and re-translates this data from its original format into XML documents (Fig. 1). It further reacts to data modifications by launching distributed consistency checks, which span all documents in the system, relevant to detected modifications. We have proposed an *incremental* checking approach that drastically improves performance of distributed checks while guaranteeing correctness of all established the consistency relations [5].

Domain and gateway domain agents encompass the role of a naming service within domains and inter-domain connectors, respectively. They facilitate location-transparent coordination between distributed agents and serve as interfaces to the functionality, contained within the domains.

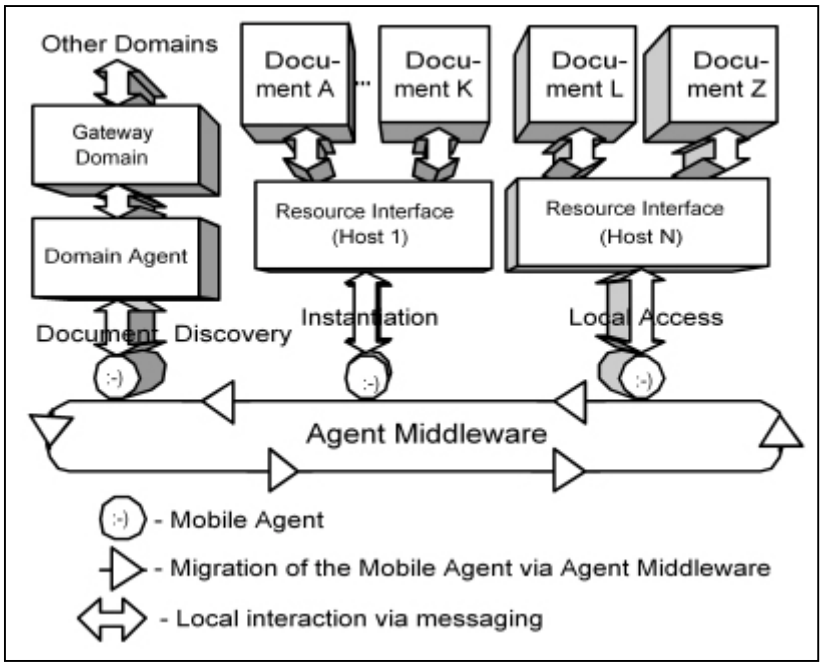

Figure 1. Single domain of a multiple-domain distributed consistency checking software agent architecture.

The central tenets of the architecture are families of mobile software agents that perform distributed consistency checks. The proposed inter-domain coordination algorithm for construction of agent *itineraries* allows mobile agents to autonomously and independently of the user locate all distributed documents that represent related data and perform local checks at each document location. Local checking improves performance, fault tolerance and addresses security constraints demanded from modern distributed systems.

The proposed flexible software agent architecture automatically performs pro-active planning for distributed processing of the measurement data, implementing the "supervisor-worker" pattern, allowing sequential processing at distributed hosts, concurrent distributed processing and coordination of independent distributed checks.

The suitable distributed processing pattern is selected by mobile agents depending on the nature of the consistency relations being checked, with a utility function that maximises the amount of concurrent distributed computation. In this adaptive distributed information processing approach, the selected processing pattern is automatically adjusted when a change occurs in the external conditions.

1. EVALUATION

We have conducted a comprehensive evaluation of the software agent architecture for distributed consistency checking [5]. We have developed a state transition model that expresses the nature of algorithms of each component and their interactions. An implementation prototype of the architecture and its operation is described in detail in [5] on the three practical scenarios from the software engineering domain. The scenarios enable demonstration of the prototype in a disconnected operation mode, in a configuration of a local area network and near internet-wide distribution.

The conducted evaluation has confirmed *elegance* of the proposed software agent architecture, established by the degree of intuitiveness, understanding and transparency of the distributed checking process for its users. Flexibility of the architecture, ability of dynamic reconfiguration of distributed components, autonomy of mobile checking agents and automatic pro-active selection of distributed information processing patterns by agent families are among the important features of the architecture that we have assessed.

The conducted quantitative performance evaluation of the prototype has established the conditions, where significant performance advantages of the incremental checking approach are realised [5]. Best incremental check results are achieved for documents of smaller size, as tree-wise differencing of document structure for identification of modified document elements is carried out on every check.

Differencing algorithms have at least linear complexity function of document size, therefore an approach for decomposition of complex documents into semantically meaningful parts has been suggested to be able to take the best advantage of the incremental checking technology for documents of all known document types within the particular application domain.

In our benchmark, a maximum 90% check acceleration has been achieved in an incremental check versus an exhaustive check of the same relations on the same measurement data, which has confirmed significance of the suggested incremental checking approach.

Multiple-agent concurrent distributed checks have proven to achieve performance improvements [5] over traditional centralised, server-side processing techniques for consistency checking [7]. An algebraic performance model of a distributed cooperative multi-agent check has been proposed to allow more formal estimation of performance advantages of a multi-agent check.

2. CONCLUSIONS AND FUTURE WORK

The proposed software agent architecture for distributed checking of consistency relations is a novel application of the concept of mobility to the problem domain. The novelty is in achieved autonomy of a mobile checking agent that enables implementation of complex systems of coordinated agent families. The evaluation of the architecture has demonstrated numerous important advantages of our approach, which takes a step further the known concept of mobile code execution at the location of the data.

The proposed technology is particularly suitable to application domain of the distributed measurement and control systems, in which the sheer size, confidentiality or timeliness of the data render timely transport of this data to a central location for processing too expensive or impossible. The suggested architecture tackles many facets of this problem by ensuring local data access and processing.

The technology for consistency checking of distributed documents, and its application to the domain of checking distributed measurement data facilitate construction of fault tolerant control systems. Generation of an output signal from compliant input data resources facilitates system-wide information security and enhances usability of the resulting control signal.

REFERENCES

1. Smolko D.S., Mobile Agent Arhitecture For Portfolio Management In *Proceeding of Internation Conference SCM' 2001,* S.Petersburg, Russia, pp. 208-211, v.1, 2001.

2. Smolko, D., Design and Evaluation of the Mobile Agent Architecture for Distributed Consistency Management. In *Proc. 23rd Int. Conference on Software Engineering*, Toronto, Canada. Doctoral Symposium, pp. 799-801, IEEE Computer Society, 2001.

3. Smolko, D., Consistency Issues in Document Management of Distributed Group Work. In *Proceedings of Doctorate Symposium, Automated Software Engineering '99, 14th IEEE International Conference*, Orlando, USA, IEEE Computer Society, 1999.

4. Finkelstein, A. and Smolko, D., Software Agent Architecture for Consistency Management in Distributed Documents. In *Proceedings of SCI/ISAS 2000 - 6th Internatiomal Conference on Information Systems Analysis and Synthesis*, Orlando, USA, vol. IX, pp. 715-719, International Institute of Informatics and Systemics, July 2000.

5. Smolko, D.S., Software Agent Architecture for Consistency Checking of Distributed Documents, *Ph.D. Thesis*, University College London, University of London, UK, January 2002.

6. Zisman, A., Finkelstein, A., Ellmer, E., Smolko, D., and Emmerich, W., Method and Apparatus for Monitoring and Maintaining the Consistency of Distributed Documents. *International Patent*, The U.K. Patent Office Patent No GB2351165, University College London (Application No GB9914232.5, 18/06/1999).

7. Nentwich, C., Capra, L., Emmerich, W., and Finkelstein, A., http://www.xlinkit.com, July 2000-2002.