NeRES – Networks for Reconfigurable Embedded Systems

Summary of discussions
(report – annex C)

5/Apr/2007

Rapporteur: Paulo Pedreiras

1 Initial address

1.1 The challenges of real-time distributed reconfiguration
***Luis Almeida* - Univ. Aveiro**

This presentation addressed the global topic of dynamic reconfiguration and discussed the wide diversity of perspectives when referring to such topic, highlighting the absence of an agreed taxonomy. Then it focused on the impact of the network in complex DES, and pointed out the problems of static approaches as well as those based on modes. Then it proposed a communication architecture that reconciles flexibility with real-time guarantees, thus well suited for supporting reconfigurability in DES. The proposed architecture follows a master-slave approach integrating on-line scheduling, admission control and dynamic updates to the communication requirements, and it is the basis of the Flexible Time-Triggered communication paradigm.

1.2 Discussion:

Hector Benitez highlighted the lack of metrics to assess the performance of this type of systems (efficiency, system complexity, level of flexibility or reconfigurability, etc)

2 Session 1

2.1 Component-Container-Connector Middlewares for dynamic reconfiguration support
***Frédéric Loiret* - CEA**

This presentation identified two application domains, *strongly constrained* and *less constrained*. In the former, static approaches are typically found, possibly using multiple modes, while the latter is more amenable to dynamic reconfiguration, e.g., to cope with SW updates during the system lifecycle. Focusing on the latter ones, it exposed the limitations of typical component and middleware-based approaches and proposed using a container/component middleware in which components are merely functional and are wrapped in containers that embed all non-functional aspects including connectors to support inter-component communication.

2.2 FRESCOR: contract-based resource management in distributed systems
***Daniel Sangorrin* - Univ. Cantabria**

The FRESCOR framework manages all the system resources (CPU, memory, energy, etc) via contracts. Applications may specify QoS parameters in some cases with acceptable ranges (e.g. min-max budget, period), importance, stability, etc, thus allowing the specification of QoS ranges. The system manages the several contracts and allocates the possible resources. The presentation focused on the network, describing the set-up of a distributed transaction.

2.3 Discussion:

- Frédéric Loiret, about the resource management in FRESCOR: can it be defined a table with different states (e.g. bad, average, good) and corresponding resource usage (e.g. 10%, 25%, 75%) for each component? Dynamic adjustment of resource usage could be achieved using the best possible QoS values specified in the QoS tables (QoS management).
  - Daniel Sangorrin: Such management can be made online. The FRESCOR framework includes admission control, so requests can be rejected or accepted once guaranteed. Different resources may have different degrees of importance (Energy, CPU, ..) to tailor the QoS management according to the specific needs.

- Ricardo Marau: When to reconfigure? How much time does it need to reconfigure?
  - Daniel Sangorrin: It should be possible to bound the time required to renegotiate a contract.

- Tullio Facchinetti. Both middlewares seem too complex. How much resources do they require?
  - Daniel Sangorrin: Online analysis must be light. It is possible to use FPGA-based HW support for some special cases.
  - Frédéric Loiret: The component-oriented middlewre is being implemented in a low-end platform. Should not take so much space...

- Luis Almeida: What about industrial connections?
  - Daniel Sangorrin: FRESCOR includes several industrial partners of different sectors, for example, Visual Tools for CCTV systems on airports, subways, etc. and Thales for component-based system design.
- Luis Almeida: And would these middleware approaches be useful for the particular case of the Intelligent Networks Group of Portugal Telecom – Inovação?
  - Miguel Santos: Difficult to integrate foreign middlewares in the current framework. The ideas, however, are interesting and relevant.

3 Session 2

3.1 Real-Time Networks from the Middleware Perspective
Hector Perez **- Univ. Cantabria**

This presentation addressed the role of the network in the development of distributed real-time platforms and the support for distributed transactions from the middleware layer. It proposed using communication ports as the schedulable network entities and addressed the requirements of such entities from both the scheduling and reconfiguration perspectives.

3.2 Real-Time Support in Middleware
Marisol Garcia Valls **- Univ. Carlos III de Madrid**

This presentation showed an overview of work developed in the group concerning middleware and OS for networked, resource constrained nodes. It focused on service-oriented approaches and dynamic reconfiguration based on QoS profiles targeting the support for portability (i.e. execution in different platforms). Integration of RT communication protocols is needed to assure overall RT behavior. The current main research lines are i) the QoS management (HOLA-QoS), ii) basic middleware (Drequiem RT-RMI) and dynamic composition and reconfiguration of service-based applications.

3.3 Discussion:

- Luis Almeida (to Hector Perez): Who triggers the reconfiguration? The network layer?
  - Hector Perez: It is not the network but a higher layer. The application sends some parameters (e.g. priorities) and the network does the reconfiguration autonomously.

- Ricardo Marau: In case there is a need for fragmentation where should it be done? Within the middleware?
  - Hector Perez: If done in the middleware, it may cause interoperability problems, thus it should be done by the application.

- Luis Almeida (to Marisol Garcia Valls): How is the service composition done?
  - Marisol Garcia Valls: If carried out at the application level it could result in problems to non-independent tasks that share buffers (e.g. some multimedia applications). It should be carried out within the system/middleware and it could be either offline or online. In the latter case, if the application tolerates the reconfiguration time, you can replace services to manage the QoS but it is rather application-dependent.

4 Session 3

4.1 Server-based scheduling in CAN: Subsystem Integration of Real-Time Systems
***Thomas Nolte* – Malardalen University**

This presentation focused on the automotive applications domain. It highlighted the trend towards integration, with several subsystems sharing the same ECUs as a means to reduce the number of active components used in cars. The PROGRESS project was also briefly reviewed, which aims at the integration of RT systems in a predictable way. Then the presentation focused on the Server-CAN, which is a control layer placed in each node that controls the access to a CAN bus, thus permitting to bound the interference among “users”, which are scheduled by a master system, using a server-based approach.

4.2 Dynamic Reconfiguration in Dependable Distributed Embedded Systems
***Julian Proenza* –Universidad de las Islas Baleares**

Dynamic reconfiguration (DR) claims to bring benefits in QoS and dependability areas. The author claims that DR benefits in the context of the former area seem more consensual but conversely, DR in dependable systems arises several questions.
Dynamic reconfiguration may improve the dependability of systems taking advantage of unintended redundancy already present in many systems, but in a limited way. However adding resources to manage the dynamic reconfiguration my degrade the dependability since the system will be more complex and thus less reliable. It may also lead to high overhead and system correctness becomes more difficult to prove.

A possible approach to reconcile reconfigurability and dependability is to handle fault tolerance at the lowest levels in the system architecture. Then, the diversity of fault scenarios must be reduced by using adequate hardware solutions, e.g. restricting the nodes fault modes, such as provision of fail-silence and use of communication protocols that guarantee consistency.

4.3 Reconfigurability in MANETs

***Tullio Facchinetti / Giorgio Buttazzo* –Universita di Pavia, Scuola Superiore Sant'Anna**

This presentation showed an overview on MANETs and the main research issues in this kind of networks. These networks are inherently reconfigurable due to mobility, which creates several challenges such as the trade off between energy and performance, the control of transmission power, the provision of self-localization, enforcing reachability, carrying out message scheduling and topology management.

4.4 Discussion:

- Luis Almeida (to Thomas Nolte): Problem addressed is integration. How can we relate integration and reconfiguration?
  - Thomas Nolte: The reconfiguration techniques are used during development to facilitate the system development (as a development strategy).

- Hector Perez: Reconfigurability as a goal may also lead to a different approach for system development that may be useful during the development phase, given it more freedom.

- Luis Almeida (to Julian Proenza). Aren't flexible systems more reliable, .e.g. by providing graceful degradation?
  - Julian Proenza: That is a romantic view. In practice it is not so clear due to the problems from higher complexity and overhead normally needed to support the extra flexibility.
  - Luis Almeida: Yes but the extra complexity is needed to check the reconfigurations, to assure safety.
  - Julian Proenza: The trade-off seems to be completely application-dependent.

- Manuel Barranco: There is also a clear relationship between fault-tolerance techniques and reconfiguration. For example, the replacement of active by backup units is a kind of reconfiguration.

- Lucia Lo Bello: Reconfiguration techniques may require more CPU and energy. In some circumstances, it may be better to not reconfigure and manage the degraded services at an higher layer.
  - Luis Almeida: Not reconfiguring might imply a loss of an important service... In this topic of reconciling flexibility with dependability, it seems that what is needed is to bound flexibility somehow, in order to derive some kind of guarantees (common agreement).
  - Joaquim Ferreira: One important aspect is when to reconfigure. How to know if a failure is transient or permanent? How to know if an overload is temporary or permanent? Depending on the situation, it might be preferable not to reconfigure.

- Luis Almeida (to Tullio Facchinetti): Can we give RT guarantees concerning communications in MANETs, given their mobility?

- Tullio Facchinetti: There is ongoing work towards that direction, e.g., using group coordination to keep connectivity. However the medium is open and so only soft guarantees are plausible. And there are other dynamic reconfiguration aspects that may also help in achieving RT communication, such as maintaining alternative paths to prevent path disruption and to carry out energy management to prevent important nodes to run out of power.

5 Session 4

5.1 Intelligent Networks Reconfiguration

***Miguel Santos* – Portugal Telecom Inovação**

This presentation focused on "intelligent newtorks" in the telecommunications area, which handle prepaid services, from mobile telephony and video streaming to SMS and MMS. Incoming calls are processed at a high layer in the system architecture, to decide if the call should be accepted or if a given message should be played, etc. Upon a connection to the billing system, it should provide a yes/no answer within 100ms. This is a high availability system in which dynamic reconfiguration is supported via replication and switching between replicas after a synchronization procedure. Each service replica alone is only reconfigured offline. It must be brought down, updated and re-activated later on.
There is, currently, no global node scheduling. The maximum number of calls is fixed, and does not depend on the node load. Dynamic reconfiguration to adapt these parameters online and avoid manual/offline updates would be very welcome.

5.2 Dynamically Self-Configuring Automotive Systems

***Magnus Persson* – KTH**

Within automotive systems, the vision of the DySCAS project is to allow the vehicle to adapt itself to the changing environmental conditions, optimizing dynamically the use of the resources (CPU and networks) but focusing on the infotainment system only.
Domains of the main challenges involved are autonomic computing, control theory, mobile nets, RTOS, ... The project targets providing a middleware with adequate support for detection of new devices attached to vehicle (PDA, MP3, ..), online software upgrade, system management (e.g. load balancing) and resource optimization.

5.3 Dynamic Configuration of Communication in Industrial Automation Systems

***Marga Marcos* – Universidad del Pays Vasco**

The presentation showed an overview of the work developed in the group in recent years, which focused on industrial applications and use of standards for the communication (e.g. CORBA, XML). Examples of application domains that have been considered and in which dynamic reconfiguration could be applied: railway transport systems, telecommunication networks.
One of the interesting applications of dynamic reconfiguration is the management replicated controllers, raising aspects such as dynamic switch between controllers, load distribution during overloads, dynamic selection between alternative controllers and device mobility.
The presentation also highlighted some feedback from industry the group works with. In particular, it seems that dynamic reconfiguration is not easily accepted by the plant managers. End users do not dominate the technology so reject its use due to the lack of familiarity. Technology verification is needed prior to acceptance. Dynamic reconfiguration should be transparent to the end-user to facilitate

its acceptance

5.4 Discussion:

- Miguel Santos: From my point of view there is interest in the telecommunication industry in dynamically managing overloads; to migrate services between nodes with different loads.

- Luis Almeida: Dynamic reconfiguration might allow faster system deployment since it requires less apriori information. This might be interesting for complex systems.

- Joaquim Ferreira: (to Magnus Persson): Can we trust cars with dynamic reconfiguration? Can we they be certified?
  - Magnus Persson: The DySCAS project is addressing the infotainment, only. No safety critical systems are being considered.

- Luis Almeida (to Marga Marcos): How important is it to make dynamic reconfiguration transparent (hide it away) for industrial automation?
  - Marga Marcos: For the moment, it is seems to be important. Maybe there are sectors, such as the automotive, in which that transparency is not so important because of factors like the technological awareness of the partners involved.
  - Luis Almeida: Could the dynamic reconfiguration mechanisms be hidden behind a simplified API or components with a simple interface so that it could be easily used?
  - Marisol Garcia Valls: Just an API is not enough. Such systems should normally be more complex but the results achieved should compensate the effort.

6 Session 5

6.1 Reconfigurable Distributed Control (time delays disturbances)
***Hector Benitez* – Univ. Nacional Autonoma de Mexico**

This presentation focused on how to incorporate reconfiguration in the control laws of automatic feedback control, to make the global control cope with variable time delays that occur as a consequence of faults.

6.2 Automatic Reconfiguration in Distributed Avionic Fuel System Control
***José Maria Giron* – Univ. Complutense de Madrid**

The fuel system for avionics is the target of the Smart Fuel project. Particularly, the project addressed the distribution of the fuel management system in helicopters and airplanes, which is currently centralized. Distribution was based on the CAN bus, interconnecting the valves, pumps and sensors. There is already physical redundancy in the system, in terms of fuel paths between deposits and also intakes, which is used in the case of valve failures. Sensors, valves and pumps are self-aware (smart components) and inform the remaining components in case of mechanical failure. Reconfiguration deals with these kind of failures, and involves fuel rerouting.

6.3 Discussion:

- Luis Almeida (to Hector Benitez): How does the control system measure the time delays, in order to reconfigure the controller?
  - Hector Benitez: The first approach used static values. Now we are considering measuring the time delays dynamically. The sequence of relevant actions for the reconfiguration we pursue is Scheduling → time delays → controller selection. The time delays measurement can be simplified by using the global clock synchronization.

- Jose-Maria Giron: In our fuel management system, we use a distributed approach in which the components broadcast their status regularly, which allows them change state and perform reconfiguration. However, there are some occasional inconsistencies in the system status that last only a few ms. This problem is now being addressed so that we reach distributed consensus together with the dynamic reconfiguration.

7 Wrap-up

- Luis Almeida (to all participants): We have seen several different perspectives of dynamic reconfiguration, from middlewares to applications, targeting higher efficiency, graceful degradation or fault-tolerance. How do you see this diversity and how can we integrate the different views?
  - Ricardo Marau: The focus should be on the middleware. It should provide all the reconfiguration services to the applications that would use them as adequate/desired.
  - Miguel Santos: Sometimes, the resources are managed directly by the application. In such case, the reconfiguration should perhaps be handled by the application directly.
  - Manuel Barranco: There seems to be two different focus, performance and fault-tolerance. Dynamic reconfiguration could allow us to reconcile both aspects, e.g., using redundancy as extra available resources for better performance when everything is OK and giving them back for fault-tolerance when a backup is needed.

- Luis Almeida: There are, of course, many open issues in the scope of dynamic reconfiguration. One such issue, is related to its use for overload management. How to define minimum performance levels? The definition of flexible requirements, as in flexible scheduling, seems to be a tough issue that will require appropriate input from the user.