Cris Pedregal Martin · cris@acm.org · Dept. Computer Science, University of New Mexico

Research Statement

All computer systems fail; good systems fail in tolerable ways. Typically industrial-strength, "good" systems support some notion of failure recovery, which enables them to predictably and even gracefully cope with failures. The problem is that in industrial-strength systems, conflicting and intertwined functional, performance, and reliability requirements result in tradeoffs encoded in the lowest levels of the implementation. Thus, while sharing the same essential ingredients and goals, each new system must be designed and built from scratch, with the attendant higher and less predictable costs, decreased degree of confidence (which must be established anew). A goal of my research is to make recoverable, industrial-strength computer/information systems easier to understand, specify, design, and ultimately, build.

My approach is to study the use of abstraction in and the relationships and tradeoffs between system components. The use of abstraction (modularization, information hiding, etc.) may help or hinder the conception, design, and analysis of real systems. For example, in my dissertation I exposed structural similarities and hidden assumptions in transactional systems by introducing abstraction to the realm of recovery support. But Massalin's dissertation showed substantial improvements in operating systems by abandoning modularity and layers. Resource tradeoffs are pervasive in computer systems. For example, the relationship between primary (e.g., RAM) and secondary (e.g., magnetic disk) storage in terms of latency, throughput, etc., is deeply reflected in the basic algorithms of a database transaction engine.

In my dissertation I applied this approach to the problem of supporting recovery properties. My thesis was that a range of diverse systems (databases, workflows, mobile and e-commerce systems) share similar recovery requirements, which are obtained by composing a small set of basic ingredients. Framing these ingredients and their relationships leads to precise yet intuitive characterizations of these systems by exposing the expectations of future behavior (e.g., failure-recovery) between their components and with respect to the infrastructure they rely on. I proposed a framework (FL, failure liveness) inspired in ACTA, to specify systems using the notions of protocols (prescriptions of the order in which events may happen) and guarantees (promises that events will happen given certain conditions). Examples of systems I characterized include the recovery component of a database system a la ARIES, a mobile database system, and an electronic-commerce system (see my papers). For example, I used the FL framework to hierarchically decompose an electronic-commerce scenario so as to establish how a global property (money is exchanged for goods) is supported by component properties (e.g., the bank honors its credit-card authorizations) and in turn by that component's internals (the database system records credit-card authorizations under failure atomicity), and their use of the infrastructure (recovery system writes to disk are later retrievable). This decomposition is formalized in each level's proof of its guarantees based on the level below's. Thus the framework is widely applicable as it enables proving properties (beyond recovery) of a system formed by autonomous components (e.g., the merchant, bank, customer, etc., form the whole e-commerce system yet each retains its autonomy).

My framework's ability to specify and reason about recovery allows better choices of recovery requirements and implementation, and generally increases the confidence on the systems built. Further work in this area includes several directions. One is to apply the framework to new example scenarios, e.g., workflows, and various advanced transaction models, to establish its breadth and propose it as the natural yardstick to compare systems, and especially their

infrastructure needs, with respect to recovery. Another is to refine the framework and add quantitative components to it: currently, the specifications expose where the tradeoffs will have to be made but the quantitative measures must be derived ad-hoc. Ultimately, the aim is to make recovery, or perhaps a generalization of the concept of guarantee, into a first-class concept in the design of systems.

A new research direction I am starting is the study of novel architectures and their tradeoffs (e.g., PDAs and other small devices, embedded to general-purpose systems and applications). The motivation is understanding how to deploy traditional and novel services, and apply algorithms in particular, to novel architectures, as well as to bring back lessons relearned in the frugal confines of small devices to general-purpose computer systems. I have identified a promising set of architectures, services, and applications to study where the tradeoffs change or break down in a realistic environment.

In addition to core Computer Science, I am interested in the boundaries with "non-technological" issues related to the increasingly critical influence of information technology in society. An example is the study of decay risks of information technology infrastructure in Latin America, due to economic, social, and political factors. Other interesting issues include characterizing, and formulating policies about, data quality, use, dissemination, access and provenance control, etc., are problems for whose solution sound technical foundations will be necessary but will also require interdisciplinary collaboration.

PDF, HTML: http://www-ccs.cs.umass.edu/cris/cris-research.html
Modified 2002.12.10 by Cris Pedregal Martin