

Title: How can we solve transaction problems?

Summary: IBM computer science Joel Branch and his colleagues are working on a modeling and monitoring system to help companies analyze - and correct - process misfires.

Featured researcher: Joel Branch

Duration: 7 minutes, 21 seconds

Introduction:

Who hasn't had to wait for a loan approval or a university registration to go through? If a process takes too long, you're going to "vote with your feet." In this episode of [Computer Science Spotlight](#), IBM computer scientist Joel Branch talks about the work that he and his colleagues are doing on a project called Transaction Process Monitor, or TRPM, that will help companies troubleshoot transaction glitches within hours or days instead of weeks - or years.

Presentation:

My name is [Joel Branch](#). The project which I'm working on is transaction process monitor, TRPM for short.

In the large-scale enterprise, there are a couple of major challenges when it comes to IT systems management, and that's modeling and that's monitoring. Let's talk about modeling.

THE MODELING COMPONENT

The problem is, you need to have a model of your IT system upon which to do some sort of subsequent application, such as monitoring. So how do you get this model?

If you have a large-scale complex IT system, it's not guaranteed to be homogeneous in nature. This system might have evolved over years of upgrades, years of merging between different companies. And to put the icing on the cake, there's very little documentation about how this IT system looks, and specifically how a transaction that goes through this IT system should facilitate. Should it go to this particular state? Should this state follow this state? What percentage of a time should a particular transaction go into this state?

So let's give an example: A large-scale banking enterprise.

When someone makes a stock trade, what does that transaction look like in the IT system? What databases are touched? Geographically, what server is touched? We want to know what the anomalies are to this.

The transactions we're dealing with are very much tied to the application-level events, basically events that will pop up into some sort of log-dump, whether that log be XML, whether it be text-based, whether it be in a database file. Those are these footprints, if you would call them, that

we're basically trying to monitor.

When we come to monitoring, the problem is, number one, we don't have these models. But number two, a lot of these components in this IT system have shipped without monitoring instrumentation. How do we do monitoring on a system which just was not designed in the first place to have some sort of monitoring instrumentation involved in it?

In a perfect world, you would have everything be in harmony with one another. And from an operating point of view, they are in harmony with one another. It takes a lot of work to make that happen. But from the point of view of modeling and monitoring such heterogeneous systems, that's something that just doesn't naturally evolve from the company's point of view.

So the solution to this problem that we've developed in the transaction process monitor project is a utility that provides technology that starts from raw transaction data sources, such as logs, such as database files, XML format files, whatever have you, basically footprints of a transaction's activity and behavior in some IT system. And what we do is we process these footprints in a semantically agnostic manner and present to a domain expert some first-cut result of what a transaction model looks like in this

particular system.

And after we have that candidate model for [the] domain expert, then we have facilities that basically step a domain expert through a process of iteratively adding some semantic understanding onto these transaction models.

So after some iterations, after some labeling, after some sort of editing on these models, the domain expert then can verify that this is, in fact, the representation of a model of this particular transaction in an IT system.

So only a modeler, a domain expert, a human, can offer this insight. Even if this is a semiautomatic process, we still can get the modeler to a point which he or she can start inspecting models in a matter of hours or days rather than a matter of months or even over a year. So there's still significant time-savings on behalf of the organization.

THE MONITORING COMPONENT

The challenge of monitoring in real-time is that there's very little monitoring instrumentation already involved into this system. And even if there were monitoring instrumentation, the system can be very heterogeneous in nature. So that poses a challenge of doing some sort of coordination between different proprietary monitoring formats, instrumentation technologies and such.

So the challenge is to take these same footprints that we use for the modeling part of TRPM, apply some statistical algorithms and basically bring a probabilistic view of where a transaction is in the system in terms of its status. We can't give a 100 percent accurate idea of where a transaction is in a system, but using these footprints and using some clever statistical methods, we can give a domain expert a pretty good idea of exactly where a transaction may be stuck in a system, or if a transaction is doing just fine.

TRPM's BUSINESS VALUE

I've always had an affinity towards business. And given our problem statement, there have been a lot of things ignored when it comes to the intersection of business processes and the IT backbone of that business. You know, at the end of the day, for just about all large organizations, daily business activity - the ability to generate revenue, to retain customers, to remain competitive in this global marketplace - relies on the health of your IT backbone. And that's not just from the point of view of making sure your employees have e-mail or can check their e-mail on their BlackBerry or can instant-message one another. No, your day-to-day business model is so tied into distributed, global computer systems.

If we can solve this problem, then we can make a lot of people happy, not from the point of view of just collecting revenue and remaining competitive, helping our clients remain competitive, but creating an environment where you can plug in this tool, where you can execute this service, and within a matter of days say, "I have a model of what transaction behavior looks like in my system." And that's invaluable for today's enterprise.

You've been listening to Computer Science Spotlight, an IBM podcast. Now available on iTunes.

Series producer: Barbara Finkelstein

Music: © [Modern Jazz Samba by Kevin MacLeod](#)