

Publication: Data Management Review

Column Title: Data Warehouse Delivery

Author: Douglas Hackney

Headline: Know When to Bail

Issue: July/August 1997

[begin copy]

I have some friends who race motorcycles. Some do it for a living, most do it for fun. Some race motorcycles in the dirt, others race motorcycles on pavement. All of them have discovered two essential, immutable laws of the universe: 1) gravity always wins and 2) pavement hurts. Thus the segment who race on the dirt.

In pavement based motorcycle racing all of the ambulatory riders share an inherent understanding of the most fundamental rule of survival in this world of speeds of over 180mph, where knee pucks (sacrificial plastic knee pads) are dragged along the pavement in the corners, and only a thin layer of leather separates their bodies from the raw, grinding buzz saw of the pavement: Know When to Bail.

With a pavement racing motorcycle, there are only a few square inches of tire surface that support the machine and the rider at these tremendous speeds and nearly flat lean angles. The men and women who excel at this sport can understand and exploit the limits of that few square inches of life prolonging traction better than you can the limits of a long lunch. When they detect the instant when they have exceeded the magic balance between available traction and g-forces, they instantly, and without hesitation, shed themselves of their motorcycle, put as much distance as possible between themselves and the tumbling, limb snapping machine, roll with the crash, and rise to walk and race another day. This is known as "the art of the bail."

Much of life shares this key to success and survival. You must "know when to bail." When is it time to walk away from an old faithful car that is beginning to "nickel and dime" you to death? When is it time to do the last backup on the old devoted PC war horse who's time is long past, and who's OS is incapable of supporting today's applications? When is it time to put as much distance as possible between you and the train wreck of a data warehousing project you've become entwined with? Here are some clues:

1. It's a solution looking for a problem.

If you're on a project who's motto is "build it and they will come", get out now. I recently had lunch with a vendor rep who has a client that invested years and millions on an IT driven data warehouse project. They've now got six months before the next budget cycle to find some users, pilot those users, demonstrate ROI, expand the user base, and justify their lives. They are very, very desperate. Neither one of us held out much hope. The only chance for data warehouse or data mart success is to build systems that are focused solutions to clear, discernible and definable business pain. If you're not in this position, keep your personal belongings packed in a few small boxes, and be ready to evacuate on short notice.

2. The team is filled with techno-centric, inflexible, bit warriors.

These systems are built to serve the needs of users. Period. If your team is worshipping at the alter of technology on yet another resume padding exercise, seek shelter. This one's headed for a nasty crash in the first few laps.

3. Everything is bet on silver bullet technology.

And what if it doesn't work out? Anybody looked at alternatives? What if the sales rep was really close to making quota and it was quarter end? Suppose they might have stretched the truth on scalability or capability just a little? Just like surviving a motorcycle ride on the street, you must always have an exit path.

4. There's little to no resources budgeted for maintenance and sustenance.

What processes and resources do you have developed, committed and tested for change requests? Addition requests? Changes in source systems? New user groups? Is Bob, the new guy, really going to be able to maintain all that undocumented code in the extracts? I guess we'll find out, won't we? If this is your world, try to get the cube closest to the fire escape.

5. There's no meta data, and no plans for any.

So the business has never had a source of integrated, scrubbed, historical data before, eh? I'll bet your new system is going to be a big hit, at least initially. At least with the knowledge workers. I guess you'll never be able to serve the majority audience of information consumers, will you? And it's also a safe bet that your utilization rates will slowly degrade, while your support costs skyrocket. This is a slow death scenario, so there's plenty of time to orchestrate your move to the web e-commerce application server project.

Just like the maestros of motorcycling, who delicately slide both ends of their bike around 160mph sweeping corners while balancing the weight of the bike on their dragging knee pad, you must carefully listen to the howl of your tires as you negotiate the twisting turns of a data warehouse project. You must know, intuitively, when to hang on and gut it out, and relish the accolades of victory; and when to throw the project away, and live to fight another day. Know when to bail.

[end copy]