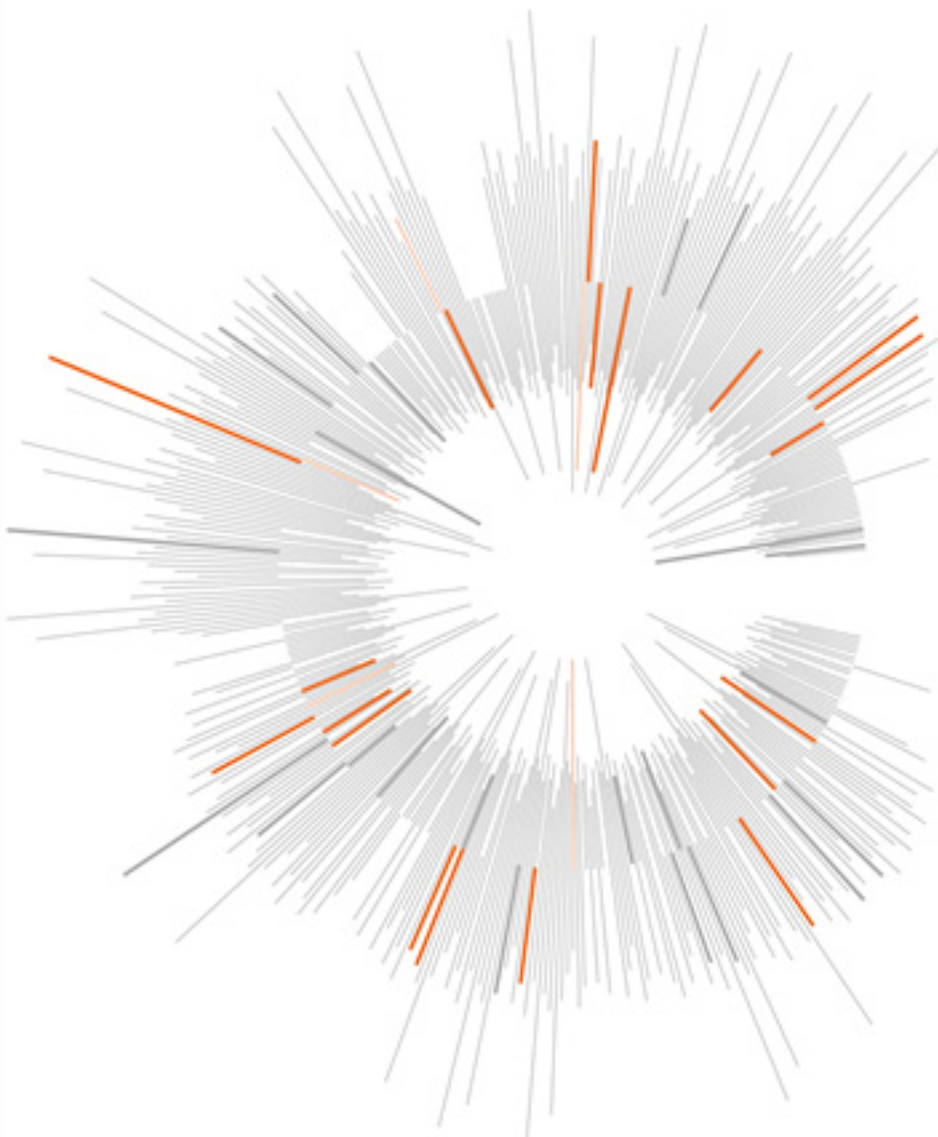


The Top Ten Misconceptions about Automated Test Data Creation

By Richard Fine



Introduction

An overwhelming percentage of projects are either masking production data, a practice with a high risk of data leakage, or having developers create test data by hand – clearly a time-consuming and arduous process. Yet, there are automated tools to create test data, such as Grid-Tools' Datamaker. So, why aren't developers using them? We've collected here some of the reasons that we most commonly encounter, along with the truth about each one.

1) The technology for automated data creation doesn't exist yet.

We've been helping customers create their test data using our Datamaker tool for over 5 years now; so, no, we're pretty sure that it exists. In fact, you might say that we're absolutely certain it does. We'd be happy to give you a demo if you don't believe us.

You might be wondering why you've not heard of automated test data creation before. Data masking is the fashionable technique, and has been around a bit longer, so it can sometimes overshadow the alternatives – but 'fashionable' does not mean 'best.' Automated test data creation is newer and less widely used right now, but every new technology begins life as the replacement for whatever ailing, inferior approach came before it.

2) Automated test data creation is time-consuming and expensive.

A recent systems merger across 47 projects between two large international banks saw project engineers learn, configure, and use Datamaker™ to generate all their initial test data within one month. Datamaker's data bulking scripts can double the amount of data you have, as fast as your database infrastructure can manage it. When compared to the time it takes to create data manually, or to the time it takes to thoroughly mask production data to an acceptable standard, automated test data creation is anything but slow.

Having a centrally managed, synthesised set of test data also dramatically decreases the amount of time engineers spend finding and manipulating test data – data re-use increases, and any requirements for new data can be met incredibly quickly. Less engineering time spent handling test data means more time to spend fixing the defects the data exposes...

3) The data doesn't test the important bits; only production or human-created data will know where to focus.

You'd think that it would be important for the data to stress-test the most critical parts of the system; code that must not fail should have more test cases than code where faults are more tolerable. How would automatically generated test data know which parts are critical, without some serious hand-holding by the developers?

It wouldn't, of course, but the mistake is to think that focus is necessary: generated test data can stress test everything. You might still choose to focus development efforts on fixing the bugs in the most critical areas, but it's better to know about bugs in non-critical areas and to choose not to fix them, than to not know about them. It can pack any part of your schema as densely as you wish, without being limited to the density of your production data, and without spending engineering time laboriously building test data for areas that are perceived as less important.

4) The data doesn't test real-world scenarios.

What is a real-world scenario if not just another data point? There's absolutely no reason why a generated test case couldn't appear in the real world – the data will be as realistic as your specification permits or enforces.

Usually, people think that the solution to this is to use de-identified production data - but using today's production data won't prepare you for the new scenarios you'll encounter tomorrow. It's not sufficient to test your system against only that which you've seen in the past; you need to prepare it to handle that which you'll see in the future, too.

5) What we need is too complicated; it won't be possible to build a data set this complex.

If it can be built for production, then it can be built for testing – and if your requirements are that complex, then your other options are even less desirable anyway. Successfully masking production data becomes exponentially harder the more complex the data is, because there are that many more pieces of information that could be correlated to crack the data. Creating the data by hand, of course, is difficult precisely because the data set is so complex; aren't complex tasks exactly the kind of thing you should be trying to automate?

6) We need terabytes of data; data creation tools won't be able to handle that much.

If your software can handle terabytes of test data, why wouldn't ours be able to? It's possible that some data creation tools might be limited how much data they can process, but Datamaker, at least, works directly with your RDBMS to generate as much data as it can handle. Whether you're using Oracle, DB2, Teradata, Sybase, SQL Server, or even just flat files, there is no limit to the amount of data that can be generated – the only limits are those imposed by your ability to store it, and that's a problem you face regardless of how you provision the data.

7) A 'production-like' environment is the ideal; we need production data, for volume and variation.

You'd think that your system should be tested in conditions as close as possible to those that it will be under in production. You should use the same hardware, the same software, the same configuration... and the same data. But actually, that's not really what matters: at the end of the day, the most important thing is that you find and fix all the problems before you go live. Testing in different conditions to production can make it harder to find problems ahead of time, certainly, but that's not always the case. In fact, it's a common practice to change some things – such as the verbosity of log files, or the amount of data validation performed – to make it easier to find problems. So why not change the data, too?

Whatever the volume and variation of your production data may be, synthetic data can not only match it, but exceed it. It can expose every bug that your current production data could expose, and more; it does so without wasting space, without wasting test cycles, and without any of the risks to data security that production data carries.

8) The time taken to train in a new tool will be too long.

There's an up-front time investment whenever a new tool or technique is brought into a project. Data synthesis isn't alone in this, of course; it takes time to learn how to mask data too, and it's not something you can take lightly, because insufficient training translates directly to insufficient masking. Having your project create the data manually might mean less training because your staff can use whichever approaches they feel comfortable with, but the resulting hodgepodge of creation styles means a much greater amount of training for your maintenance team. Standardizing how manual test data gets created will mean training

everyone on the standard anyway, so why not pick a standard approach that circumvents manual creation entirely – automatic generation?

Let's not take this out of context, either. We run training courses for Datamaker. After a one-hour webinar and a two-hour tutorial, people are ready to begin using the software; and after a week of using it, most are ready to become certified experts.

9) It's not worth it for a one-off exercise.

Of the three main techniques available to you for data provisioning – masking your production data, creating data by hand, or using a data synthesis tool such as Datamaker – only the data synthesis tool actually stands any chance of being used again for future projects. Data masking has to be carefully planned according to the particular nature of each project's production data and the governing regulations, while hand-crafted data is useless beyond the project's immediate requirements. A data synthesis tool, meanwhile, can be used for every project you ever take on – all you have to do is plug in your new schema.

10) We'll have to redo everything whenever we begin work on a new version.

It's conceivable that there could be some data generation packages out there that work on a one-off basis, but it's by no means necessary to the technique. Datamaker has very strong support for evolving the data over multiple versions of your project, versioning it lock-step with you and allowing you to easily reconcile different versions of the data. So, it's very easy not only to continue to permit end-of-lifecycle development on old versions, but also to base new versions on extensions of your existing data. Every test case used to test the old version can be easily brought into the new, ensuring no regression bugs.

Conclusion

The fact that it's now possible to easily create synthetic test data – data that is extensive, safe, and at least as high quality as what you can create with data masking – is still new to many people, and there's a certain amount of pessimism about what's actually realistic. When you dig into the facts, however, you can see that not only is automated test data creation a viable alternative today, it's actually the best option for the majority of cases.

About Grid-Tools

Grid-Tools are specialists in test data creation, data masking and test data management. Their experienced personnel have been writing and developing solutions for large companies in both the private and public sectors for over 30 years.

The Grid-Tools Datamaker suite includes a wide range of tools for test data management including such innovative products as Datamaker, a revolutionary tool that creates and publishes quality test data from scratch whilst keeping the relationships and referential integrity of production environments. An invaluable tool for testing and development, Datamaker places the data into a central data repository so it can be used, inherited, manipulated and re-used across an entire organization. Voted “Most Innovative Testing Tool of 2008” by QA Guild, the functionality Datamaker provides has proven to be innovative, efficient and different than any other test data management product in the market. You simply will not find another tool like it!

Grid Tools

11 Oasis Business Park
Eynsham
Oxfordshire
OX29 4TP

UK: +44 01865 884 600

US: +1 866 563 3120

E: info@grid-tools.com

www.grid-tools.com

[Find us on Facebook](#)

[Follow us on Twitter](#)

[Join the Datamaker circle on LinkedIn](#)

[Subscribe to our blog](#)


DATA FIT FOR PURPOSE