



**Grid-Tools**  
The power of test data

Customer user story  
Large UK Government Agency

## Case Study: Large UK Government Agency As told by the Project Manager

### The scale of the project

- 5 initial conversions for existing projects involving data, user training and deployment
- 5 additional projects
- Total current user base of 180 in teams ranging from one or two developers to 60+
- 100K Test Cases / Test Packs
- 4K Standardized Cases
- 3,420 Tables
- 690K rows of data
- 45 discrete initial schema Versions
- Performance and Volume Testing completed for 500 simulated users on our internal network

### The task at hand

In our organisation a lot of time was invested in data management for development and testing. Five business critical projects used an out-of-date tool to manage the data.

The largest project had five years of data created by team of up to 120 developers. Activity Based Cost Modelling showed us that 10% of the development time was spent dealing with test data, not an insignificant figure. It was estimated that it would have taken the development teams 6 months to re-create this data (approx 1 years' downtime) and it was therefore imperative there was minimum disruption while the data was converted and the new tool was implemented. We had to be sure a developer that used data one week would be able to use the same data the next with minimum downtime. A percentage of this data was not actually being used, but still needed to be validated. Some of the existing data was corrupted and needed to be repaired to ensure its suitability for reuse.

We were also dealing with an unusual environment in the sense that we have multiple operating systems and multiple versions of Oracle. Our problem was that the commercially available tools wouldn't have met the requirements of the task.

### The need for the correct tool

Although there were other tools in the market that would generate data, we needed a bespoke solution that certain suppliers couldn't or would not provide. We needed to work with a dynamic yet flexible company with the right knowledge. Grid-Tools matched the demands of this task and were able to support us with the right tool and plenty of support to our requirements. Some projects had mandated the usage of the existing data generator tool in the development lifecycle. This meant that developers had to use the tool to create or modify the data detail and, for future reference specify, where in the existing tool the data was stored. Independent assurance of the data occurred to ensure that processes were being followed.

**“This developed into a solution which exceeded the initial objectives and ended up actually saving us money. ROI is estimated at approx £1 million in terms of ‘work days’ saved.”**

**Project Manager, Large UK Government Agency**

Not all projects used the old tool in the same way, some were more basic in their usage and some were very advanced, thus Grid-Tools had to provide complex solutions to meet all requirements.

Initially this project was set-up and defined as a 'Risk Mitigation Project' to replace the out of date tool. However, this developed into a solution which exceeded the initial objectives and ended up actually saving us money. ROI is estimated at approx £1 million in terms of 'work days' saved.

The increased functionality of the Grid-Tools product over the existing product meant that many actions were considerably simplified and accelerated. For example, migrating of data forward to a new release in the old tool took one month of manual effort. However, the improved tool can do this task in less than a day. We have also found that some projects that couldn't formally migrate their data now can, due to the Grid-Tools data cleansing. This is a huge advantage to us.

We have now expanded the use of the tool for testing to 5 additional projects that previously didn't manage the test data in a standardised format. This is due to the flexibility of the Grid-Tools product. We intend to make Datamaker the tool of choice on the majority of our Oracle related projects.

### **The next steps**

A new investment project is being set up to deploy the Database Compare functionality of Datamaker to 180 existing users, additional development users and to System Test teams (another 300 potential users). Initial analysis shows a return on investment of at least £1m over two years. This will replace two existing tools with a standard product and provide consistent tooling support and software licence savings.

## ***Taking the TeDiuM out of Test Data Management***

The nature of our business is that we have a mandatory 2 release per year schedule. These releases are at set times and cannot be moved. Due to this constraint, many of the projects had the principle that test data, from a development perspective, should be designed and stored with the concept of being reusable. We needed to reuse and share data amongst each development community. Developers took a lot of time and effort (10% of their time) in shaping this test data and over 5 years had build up a substantial amount of valuable data. One project had up to 160 testable functional areas with 1700 testable areas and almost 15,000 individual test cases. Some projects also used parallel development, meaning that they were working on up to 3 software releases at the same time. This meant that developers could be working on data and code from System Test, new functionality and live issues at the same time. Obviously the database schemas differed with each release and the development tool used was very selective in that it did not run when the testing database did not match the internal model database. So data had to be kept for all these releases.

This sounds like a high requirement to meet. Well, fortunately, there was an onsite tool which did this job. It had been developed in-house and had grown to be invaluable to the development process. However, like all software it needed enhancing and, as this had been written in VB3, this was going to be difficult. As the performance degraded and as the amount of data stored was far beyond anything ever imagined at the inception of the tool, actions were taken to upgrade the database and migrate everything over, but this was too problematic. At this point, the existing tool was seen as a red risk to the company and a programme was started to replace the tool with something more stable. Numerous tools were investigated that could assist with test data and many companies had fine offerings, but they were unsuitable as we would not be able to convert our existing test data into their format. For us - the main focus of the exercise was always going to be the conversion of the existing stored test data.

Like many home grown products we had little in the way of formal build documentation. We had an abundance of support documentation - but there was very little info on the requirements of the initial application, so we found that basic issues like how we stored the data was not clear.

## The project essentially had 4 components:

### Conversion and validation of the existing data

Data conversion was an onerous task. The initial problem being that, although all the tests had a correct set of data that was in constant use, there was also a set of data that was not correctly maintained. Obviously it is particularly difficult to validate and convert in this situation as this extra data hindered any attempt to work a sensible algorithm.

### Development of the tool

The tool was developed to mimic the existing system, including many complex areas such as generics and data substitution variables, which were reusable modules of data. Another issue was that the existing user base was very varied, some using the existing tool to its full capacity and some using a more basic structure.

### Implementation of the tool

After the software had been written and tested, we ran PVT of 500 virtual users to measure capacity. After this was successfully done, we carried out a staged rollout, converting the individual project data and installing the software project by project. This was an onerous activity, as we had to remove the data from SQL server, convert into an Oracle format and get it to the provider (Grid-Tools). They then converted the data and we merged it into the database. Then, project by project, we closed-off the existing tool.

### Maintenance and rollout to other business areas

After the initial risk was mitigated, we looked to improve our ROI by pushing the product out to other areas which did not have a central data repository. Currently this product is being installed in our data warehouse and has the potential to be used in all projects that use an Oracle database.

The best thing about working with Grid-Tools is that we could request enhancements that would allow the tool to be used by other projects. For Example, the initial projects did not use clob datatypes - but some of the new adopters will. We are confident that our provider can provide these necessary enhancements in a suitable time frame. We are also working on a database compare feature with Grid-Tools to be utilised across testing a development and have confidence that we can continue to strengthen our partnership and continue to save money as, for this new component, analysis also shows a return on investment of at least £1m per year.

## Contact Grid-Tools

Grid-Tools Ltd  
Oxford, United Kingdom

UK: +44 (0) 1865 884 600  
USA: 1-866-563-3120

[sales@grid-tools.com](mailto:sales@grid-tools.com)  
[support@grid-tools.com](mailto:support@grid-tools.com)

**Grid-Tools** specializes in data creation, data masking and test data management tools, solutions and services. Their experienced personnel have been writing and developing solutions for large companies in both the private and public sectors for nearly 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 with the referential integrity of production environments for testing and development. Datamaker™ offers three methods for managing and generating data in testing and development inclusive of database subsetting, data obfuscation (data masking and data de-identification) and synthetic data creation.

The **Grid-Tools** methodology consists of using the “data-centric” approach to testing. Their focus is to ensure the quality of the test data you are using is of the right quality for successful testing and development.