

Denali DQS – Taking Test Practices Places!

Keywords-

Denali, Data Quality, Data Mining, QA,
Testing Practices, Data Quality Knowledge
Base, Business Intelligence Semantic Model

Gunjan Jain & Raj Kamal, Microsoft Corp.

Email: gunjain@microsoft.com,
rajkamal@microsoft.com

Microsoft Corporation

Building -2, Microsoft Campus

Gachibowli , Hyderabad (A.P.)

500032

Denali DQS Taking Test Practices Places!

Testing activities have sadly become more of verifying that every "t" is crossed and every "i" is dotted, while they should ideally be focussing on ascertaining the viability of the word thus formed. Within testing itself, data is an important spoke in the wheel – without good test data you are not really testing all the conditions for failure and it will take you off-guard at later point when it happens in production.

Quality of data is critical to Quality of Service (QoS) provided by applications. Having Bad or Incomplete data sometime can be more damaging than having no data at all. Poor data quality can seriously hinder or damage the efficiency and effectiveness of organizations and businesses. If not identified and corrected early on, defective data can contaminate all downstream systems and information assets which will have a strong cascading effect. The growing awareness of such repercussions has led to major public initiatives like the "Data Quality Act" in the USA and the "European 2003/98" directive of the European Parliament. Enterprises must present data that meet very stringent data quality levels, especially in the light of recent compliance regulations standards.

If we wait until wrong decisions are made by customers relying on the bad or inaccurate data - we are in trouble. Data quality is often overlooked and yet there is no well-defined testing process or that provides guidance around ways to improve, maintain and monitor the data quality in any data-centric project. With the absence of any specialized testing tools that can aid in testing for data quality makes the entire situation even tougher. However the bad times are now over and with tools like upcoming SQL Server "Denali" Data Quality Services (DQS) we can turn the tables around. This artifact will show you how by using Denali DQS but it can be also applied on the other tools which have similar capabilities.

But before we go down any further, we want to take this opportunity to make it very clear that this work is not to talk/evangelize Denali DQS features but to exhibit how they can be leveraged by QA/Test team making use of exciting customer assisting knowledge discovery from data, domain values

Data Quality Testing – Why it's not popular?

- ✚ Lot of emphasis goes on testing UI and the code / logic / business rules but data is treated as a second class citizen
- ✚ DQ until recently didn't get its due attention and hence was ignored and unfashionable
- ✚ Lack of Testing Tools to do it (*Well, just wait, now there is one*)
- ✚ Lack of skillset (our Testers are traditionally trained in testing APIs or functional aspects of applications more than the data flowing through it)
- ✚ Even if teams wants to do it, there is no good guidance/documentation around it
- ✚ The effort required to test data quality is often underestimated
- ✚ Teams assume that Data quality is only important for Business Intelligence and Data Warehousing projects and we are not applicable.

Well, times are changing now. Customers are paying lot of attention to data quality now and the commitment is shown from major players including Microsoft and Denali DQS is a very first step in that direction.

Breaking the ice

Data quality is defined as the degree to which the data is suitable for usage in the required business processes. The quality of data can be defined, measured and managed through various data quality metrics such as completeness, conformity, consistency, accuracy, duplication etc.

- ✚ Incorrect data can result from user entry errors, corruption in transmission or storage, or mismatched data dictionary definitions. Aggregating data from different sources that use different data standards can result in inconsistent data, as can applying an arbitrary rule or overwriting historical data. Incorrect data affects the ability of a business to perform its business functions and to provide services to its customers, resulting in a loss of credibility and revenue, customer dissatisfaction, and compliance issues. Automated systems often are unable to work with incorrect data, and it wastes the time and energy of people performing manual processes. Incorrect data can wreak havoc with data analysis, reporting, data mining, and warehousing.
- ✚ High-quality data is critical to the efficiency of businesses and institutions. An organization of any size can use DQS or similar tool to improve the information value of its data, making the data more suitable for its intended use. A data quality solution can make data more reliable, accessible, and reusable. It can improve the completeness, accuracy, conformity, and consistency of your data, resolving problems caused by bad data in business intelligence or data warehouse workloads, as well as in operational OLTP systems.

Let's meet Denali DQS

In its latest avatar, Microsoft SQL Server (codenamed Denali) takes on the onus of ensuring the following –

- ✚ Knowledge Management, which involves creating and maintaining a Data Quality Knowledge Base (DQKB), is including supporting 'knowledge discovery' – an automated computer-assisted acquisition of knowledge from a data source sample, that is reused for performing various data quality operations, such as data cleansing and matching.
- ✚ Data Quality Projects, which enable correcting, standardizing and matching source data according to domain values, rules and reference data associated with a designated DQKB.
- ✚ Administration with regards to monitoring the current and past DQ processes and defining the various parameters governing the overall DQ activities in server.

The main concept behind DQS is a rapid, easy-to-deploy, and easy-to-use data quality system that can be set up and used practically in minutes. DQS is applicable across different scenarios by providing customers with capabilities that help improve the quality of their data. Data is usually

generated by multiple systems and parties across organizational and geographic boundaries and often contains inaccurate, incomplete or stale data elements

The following scenarios are the data quality problems addressed by DQS in SQL Server "Denali".

Data Quality Issue	Description																				
Completeness	Is all the required information available? Are data values missing, or in an unusable state? In some cases, missing data is irrelevant, but when the information that is missing is critical to a specific business process, completeness becomes an issue. Example: if you have an email field where only 50,000 values are present out of a total of 75,000 records, then the email field is 66.6% complete.																				
Conformity	Are there expectations that data values conform to specified formats? If so, do all the values conform to these formats? Maintaining conformance to specific formats is important in data representation, presentation, aggregate reporting, search, and establishing key relationships. Example: The Gender codes in two different systems are represented differently; in one system the codes are defined as 'M', 'F' and 'U' whereas in the second system they appear as 0, 1, and 2.																				
Consistency	Do values represent the same meaning? Example: Is revenue always presented in Dollars or also in Euro?																				
Accuracy	Do data objects accurately represent the "real-world" values they are expected to model? Incorrect spellings of product or person names, addresses, and even untimely or not current data can impact operational and analytical applications. Example: A customer's address is a valid USPS address. However, the ZIP code is incorrect and the customer name contains a spelling mistake.																				
Validity	Do data values fall within acceptable ranges? Example: Salary values should be between 60,000 and 120,000 for position levels 51 and 52.																				
Duplication	Are there multiple, unnecessary representations of the same data objects within your data set? The inability to maintain a single representation for each entity across your systems poses numerous vulnerabilities and risks. Duplicates are measured as a percentage of the overall number of records. There can be duplicate individuals, companies, addresses, product lines, invoices and so on. The following example depicts duplicate records existing in a data set: <table border="1" data-bbox="475 1458 1417 1648"> <thead> <tr> <th>Name</th> <th>Address</th> <th>Postal Code</th> <th>City</th> <th>State</th> </tr> </thead> <tbody> <tr> <td>Mag. Smith</td> <td>545 S Valley View D. # 136</td> <td>34563</td> <td><Anytown></td> <td>New York</td> </tr> <tr> <td>Margaret smith</td> <td>545 Valley View ave unit 136</td> <td>34563-2341</td> <td><Anytown></td> <td>New-York</td> </tr> <tr> <td>Maggie Smith</td> <td>545 S Valley View Dr</td> <td></td> <td><Anytown></td> <td>NY.</td> </tr> </tbody> </table>	Name	Address	Postal Code	City	State	Mag. Smith	545 S Valley View D. # 136	34563	<Anytown>	New York	Margaret smith	545 Valley View ave unit 136	34563-2341	<Anytown>	New-York	Maggie Smith	545 S Valley View Dr		<Anytown>	NY.
Name	Address	Postal Code	City	State																	
Mag. Smith	545 S Valley View D. # 136	34563	<Anytown>	New York																	
Margaret smith	545 Valley View ave unit 136	34563-2341	<Anytown>	New-York																	
Maggie Smith	545 S Valley View Dr		<Anytown>	NY.																	

*Sourced from MSDN FAQs on Denali DQS

The DQS knowledge base approach enables the organization, through its data experts; to efficiently capture and refine the data quality related knowledge in a Data Quality Knowledge Base (DQKB).

So that brings us to DQKB, what exactly is this thing? Well, *DQS is a knowledge-driven solution, and in its heart resides the DQKB. A DQKB stores all the knowledge related to some specific type of data sources, and is maintained by the organization's data expert (often referred to as a data steward). For example, one DQKB can handle information on an organization's customer database, while another can handle employees' database.*

The DQKB contains data domains that relate to the data source (for example: name, city, state, zip code, ID). For each data domain, the DQKB stores all identified terms, spelling errors, validation and business rules, and reference data that can be used to perform data quality actions on the data source.

Test Teams & DQS: The 'π' Matrix

Indeed, with all above, we can actually help communities like the Test Teams do great stuff on data quality. Elaborated are the ways and means for accelerated adoption of Denali DQS or similar tool in testing life cycle. We are particularly looking at adoption of DQS in a more pronounced way within the testing community, largely due to their active involvement in quality assurance tasks pertaining to data quality as well as the bouquet of features Denali brings to the table for the same.

The 'π' Matrix is nothing but a Phase V/s Issue Matrix (hence 'PI') indicating various states a project can be in depending on the amount of Data Quality issues encountered and the stage in the Test Execution Cycle.



Figure:1 – The PIE Diagram

As indicated in the above diagram, we have four sets of teams with respect to their state of data quality in projects. The **Superstars**, to begin with, are already a lot which is basking in the glory of a job well done – it is this state the other three need to aim for.

The **Flyers** on the other hand, are having an ideal platform to move to the Superstars category should they continue to focus attention on Data Quality. The usual bane in this situation is that of complacency or other forms which all eventually lead to a lack of focus on Data Quality – this is what ideally forms the mantra for teams in this quadrant; continue their focus on Data Quality.

The teams tagged in **Traps** are primarily projects where bugs have piled up in the early stages of testing itself arising out of bad data quality. Since Traps are observed at early stages of test execution cycle, by providing a renewed focus on Data Quality checks, these can be effectively countered. Negligence can otherwise pretty much move them into the Nightmare quadrant. It would be worthwhile to identify a few ways in which Traps can gradually move into Superstars.

- ✚ Since DQ issues are observed at early stages in test execution which result into bugs stacks, immediate cleansing of the data is advised before aggravating the damage – this is essentially the ‘damage control’ part.
- ✚ Client should be informed about the data quality (if data is provided by the client) and should also be informed about the risk of having faulty data in the system. This is the ‘alerting’ part of the steps.
- ✚ It should be kept in mind that if DQ issues are not fixed until the UAT phase, client satisfaction will take a beating and this will have a direct impact on the rapport of the company.
- ✚ Lastly, due to data issues, many bugs can remain uncovered. It is suggested that DQ implementation should be part of test planning once data issues are observed in PoC phase to avoid major pitfalls later.

The last quadrant talks of ‘The **Nightmares**’ which pretty much signal a project gone awry due to data quality issues. In these cases, DQ issues are observed very late in the test execution cycle.

- ✚ Once you are in Nightmare, please understand, it is almost impossible to move into Superstar quadrant. It is clearly because of lack of DQ implementation at early stages that the project has taken a beating.
- ✚ One should take a lesson from the failure and learn to religiously add DQ implementation right from the planning phase of the next release including DQ implementation as part of the scope, design and testing. Measuring & monitoring DQ on daily/weekly basis and taking actions against important DQ issues to fix them early has already been suggested.

DQS On-boarding Strategy for Test teams

The below strategy map outlines our objective and the contributors towards the same.

As indicated, we can drive the adoption of Denali DQS by means of increasing awareness among the teams who are yet to know the manifold benefits and within those who know, there can be an increased adoption rate if such teams start using it.

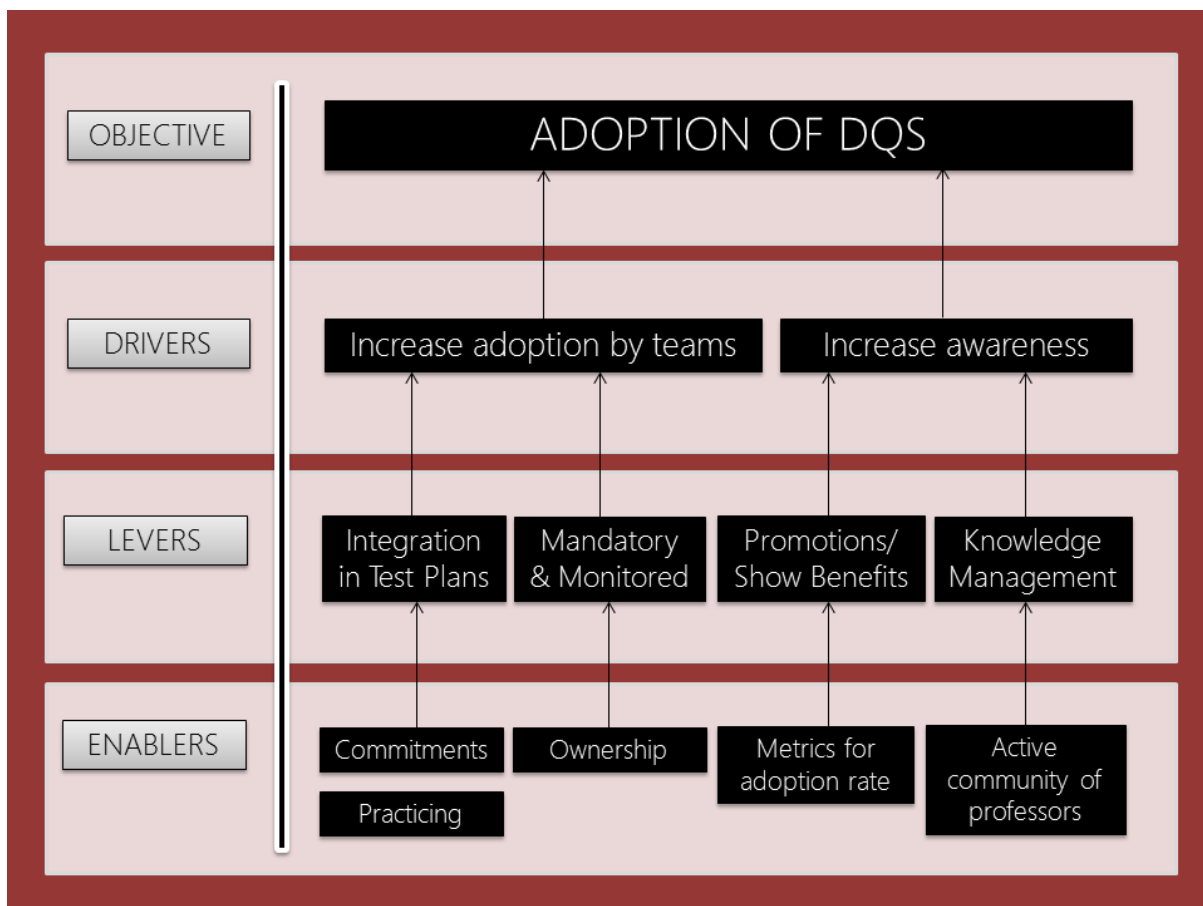


Figure:2 – The Strategy Map

In order to leverage these drivers, we again have two different sets of levers. For increasing adoption, we can tightly integrate Denali DQS usage with test plans and work towards making the same a mandatory best practice as well as monitor its usage.

To increase awareness, promotions coupled with SQL Server programmes showing benefits to end users, partners and clients are a must. To sustain such efforts, effective knowledge management is the other logical step.

Lastly, in order to rightly enable these levers, we need to have commitment, ownership and on-the-ground practice with regards to integration in test plans and monitoring of the same. Metrics highlighting the adoption rate are an indicator of the performance of promotion initiatives. And in order to have sound knowledge management, as long established, an active community of users & evangelists – or professors, is most important.

With this strategy in place, we next progress to the actual implementation ways.

DQS Use cases for Testing

Let's move to the actual use cases of DQS using Denali while testing. We will talk about two scenarios here, a preventive one and the other being a curing one.

Scenario: 1 General Data Cleansing: Preventive

Prevention is more applicable to cases where one is at the early stages of the test execution cycle, having just received databases and it is required to cleanse it with respect to issues like date format, spelling mistakes, duplicate data et al.

With DQS featured in Denali, the whole process of cleansing the data is a breeze and can be done by anyone in the project team. You do not need a data administrator for this. This is the beauty of DQS-Denali. It is provided with a simple, user-friendly interface; allowing even someone not well-versed with SQL to reap benefits of data quality services using Denali.

As the diagram suggests, we are cleansing our data following general data quality standards like no data redundancy, date format etc. We simply create a knowledge base with different domain rules and pass our production data through these rules. At the end, we get our cleansed database - ready to be used further down the execution cycle.

A few dry runs showed that this helps take care of about one in every three bugs which are observed during different phases of testing because of data quality (rather, the lack of it).

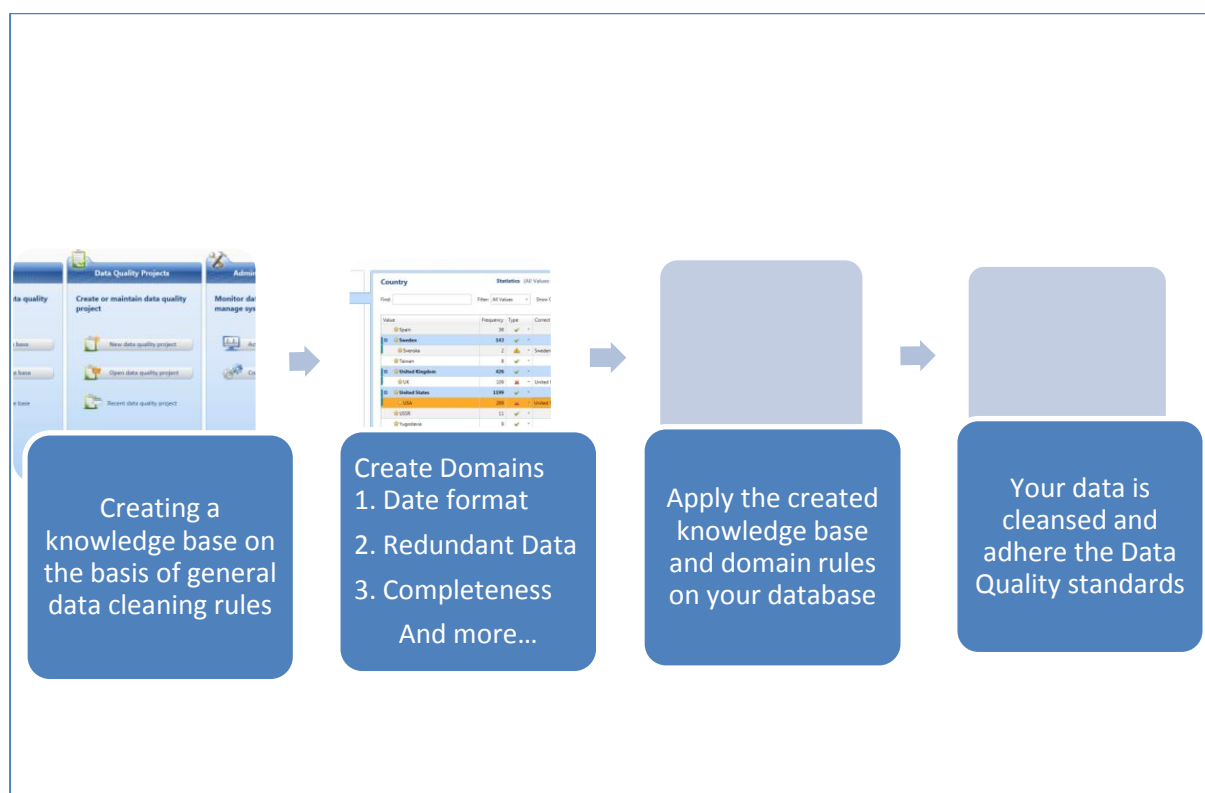


Figure:3 – General Data Cleansing: Preventive

Scenario: 2 When data issues are observed: Curing (Reactive)

This second scenario talks about the case when you have corrupt data and you have already found bugs because of data issues. Since data quality is a frontier not easily suspected by test teams, it has been found that 20% of the bugs are because of data quality but they eat into 80% of the total debugging time. So data cleansing has its own importance and benefits.

Data cleansing doesn't only depend on the general rules but it also depends on business rules like for example there can be a requirement where USER_ID is expected to be only 4 characters long. In such scenarios, we have to define business rules too in the knowledge base.

As indicated in the diagram, the DQS user defines business rules for the data cleansing and your data goes through the above shown steps. At the end, you get a database which is suitable for testing and will have lesser data issues.

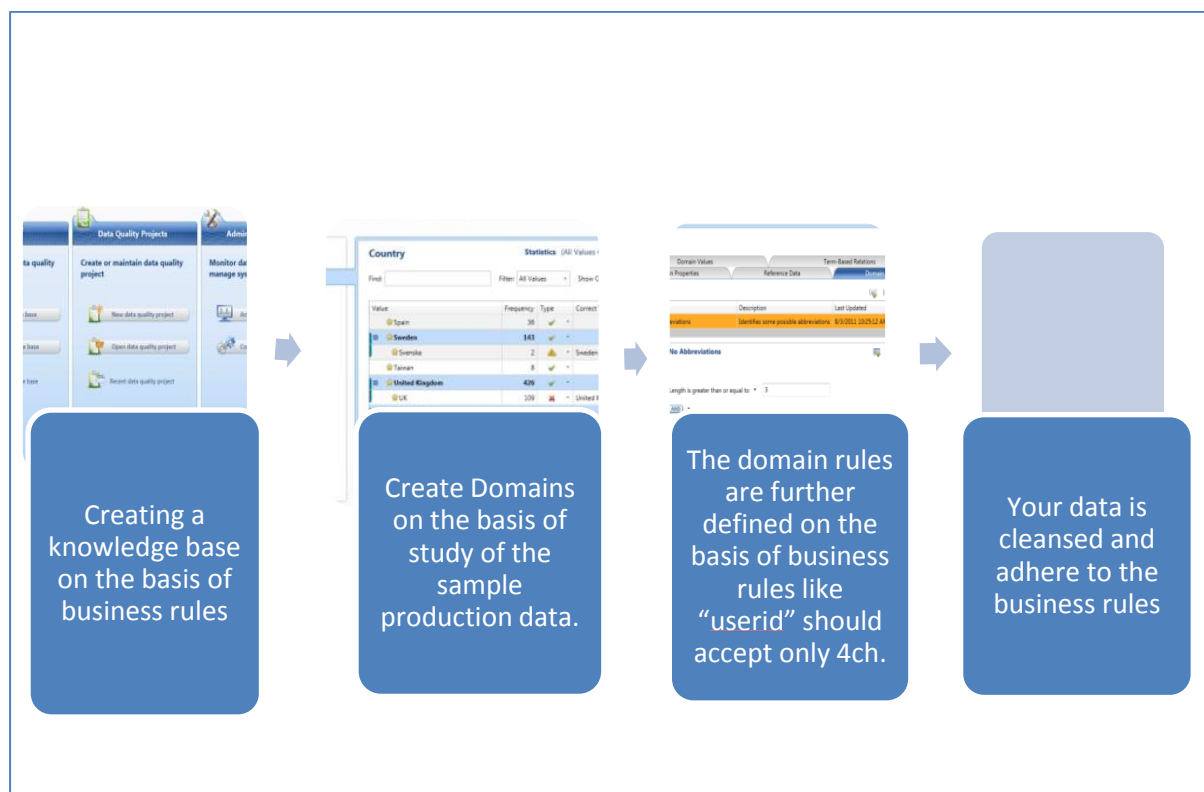


Figure:4 - When data issues are observed: Curing (Reactive)

Note: This is just a glimpse of the cool things that can be done with DQS, there are many other features and capabilities that can help you uncover bugs which are almost impossible to do with naked eyes.

Denali-DQS: Test the waters

Now that we have seen couple of use cases for it, why not test the waters yourself and try this out.

Step-1 Creation and definition knowledge base & domain management

To cleanse data, you have to have knowledge about the data. To prepare knowledge for a data quality project, you build and maintain a knowledge base (KB) that DQS can use to identify incorrect or invalid data. DQS enables you to use both computer-assisted and interactive processes to create, build, and update your knowledge base. Knowledge in a knowledge base is maintained in domains, each of which is specific to a data field. The knowledge base is a repository of knowledge about your data that enables you to understand your data and maintain its integrity.

A) The DQS Client

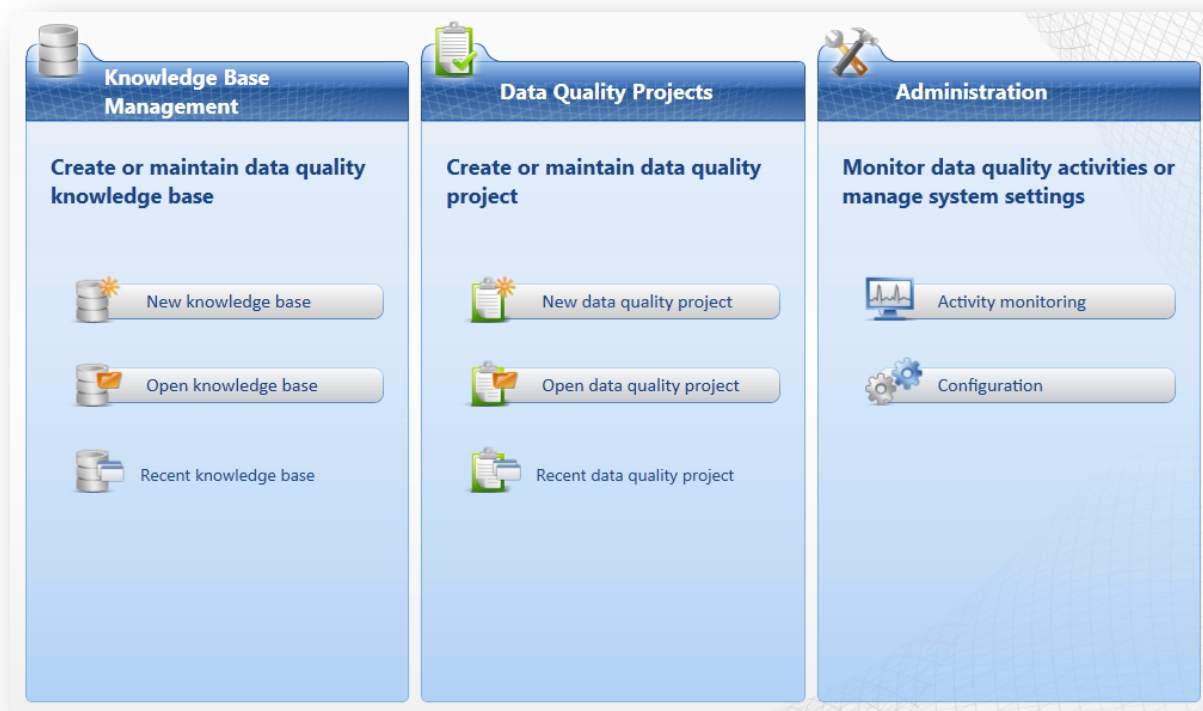


Figure : 5 – The DQS Client

B) Create the Knowledge Base

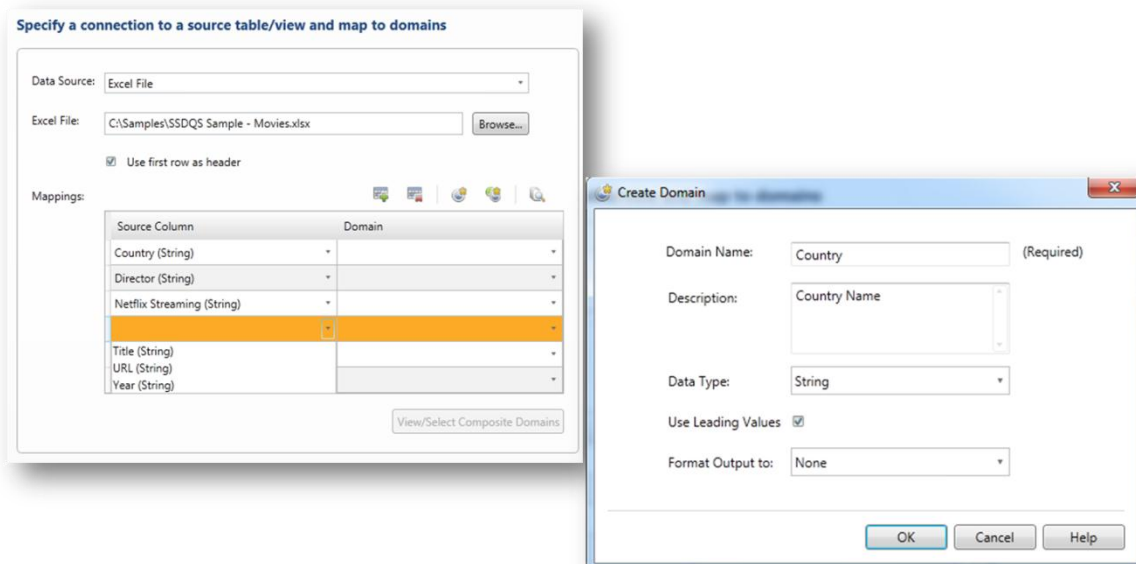


Figure:6 – Create the knowledge base

C) Analysis and managing result of first pass

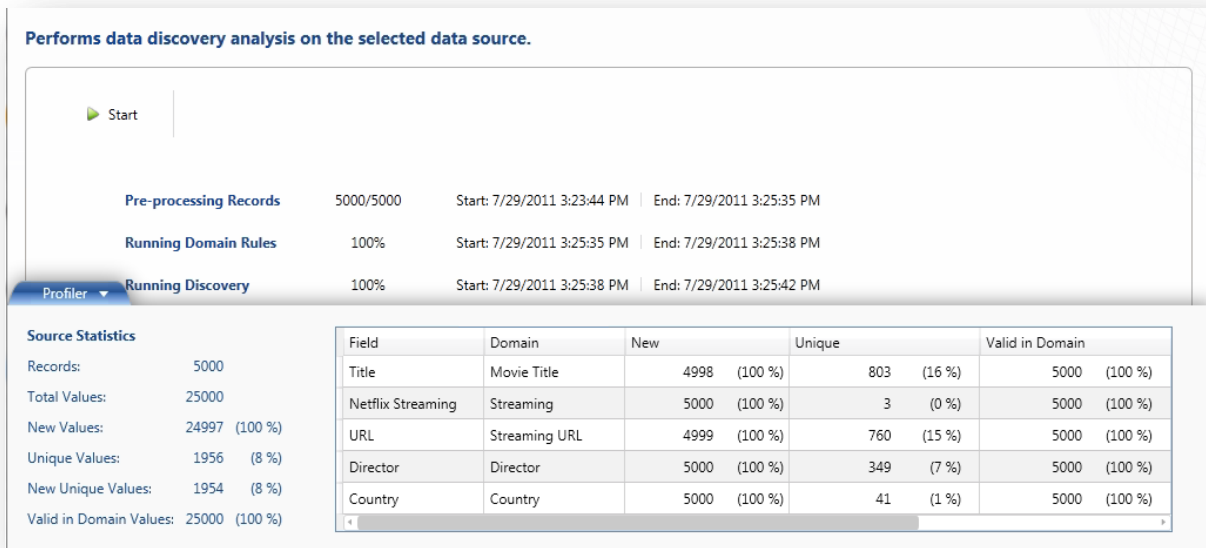


Figure:7 – Result screen

D) Domain management

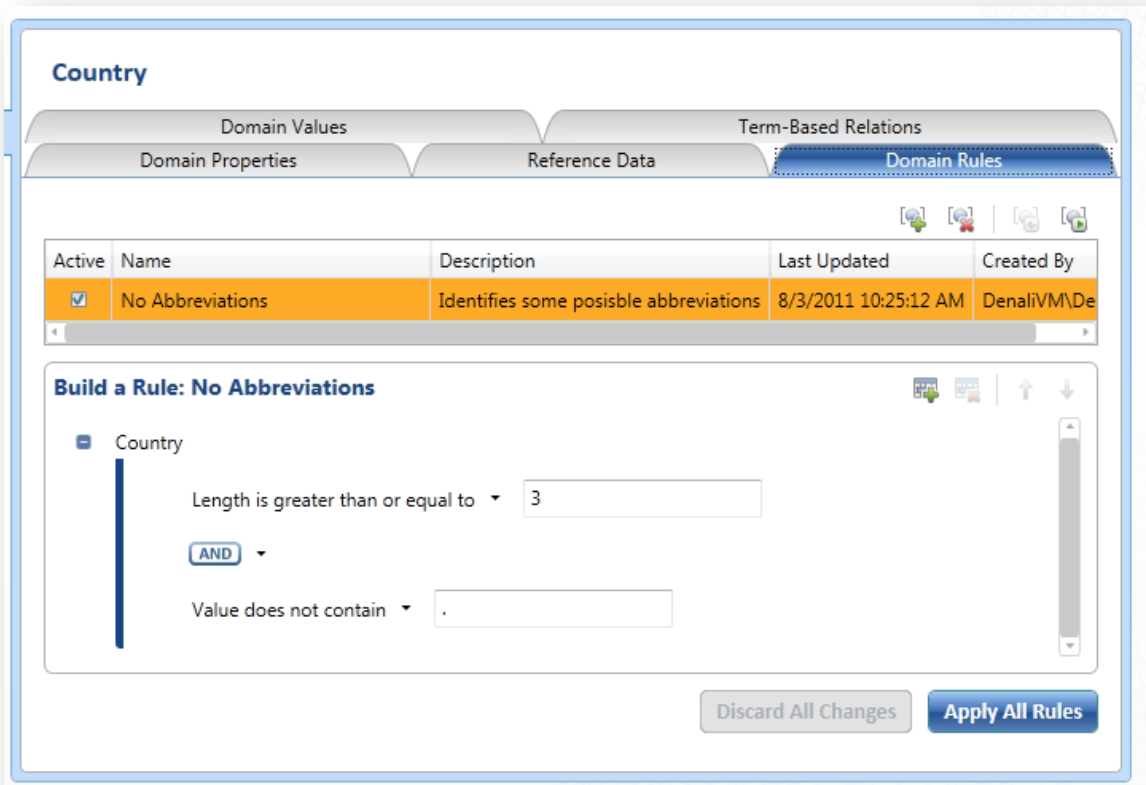


Figure : 8 – Domain management

Step-2 Data Quality Project – Cleansing and Matching

A data quality project in DQS is a means of using a knowledge base to improve the quality of your source data by performing *data cleansing* and *data matching* activities, and then exporting the resultant data to a SQL Server database or a .csv file. You can create a data quality project as a cleansing project or a matching project to perform respective activities. Cleansing and matching projects can be run using the same knowledge base, because knowledge for data cleansing and matching can be built into the same knowledge base

A) Data Quality project – Create a project and specify the domains

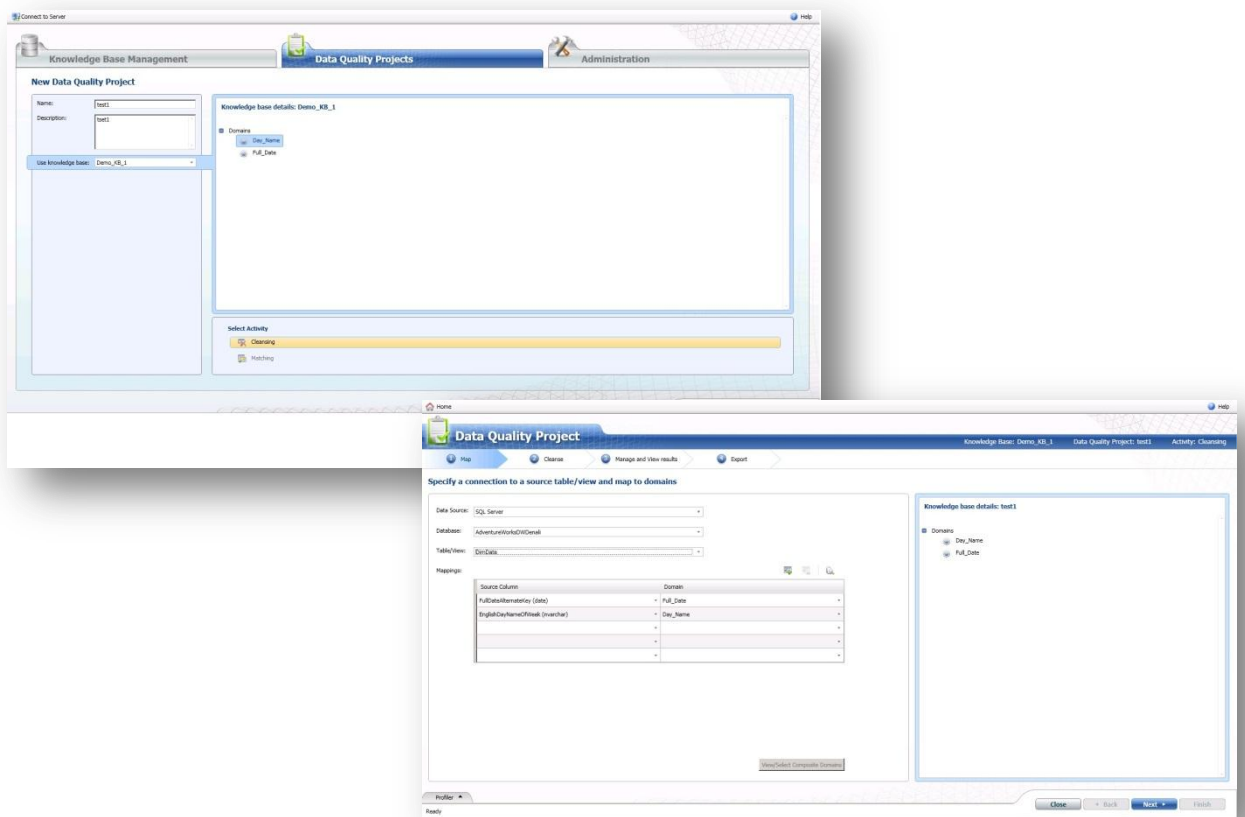


Figure : 9 – Create a project

B) Cleansing Result

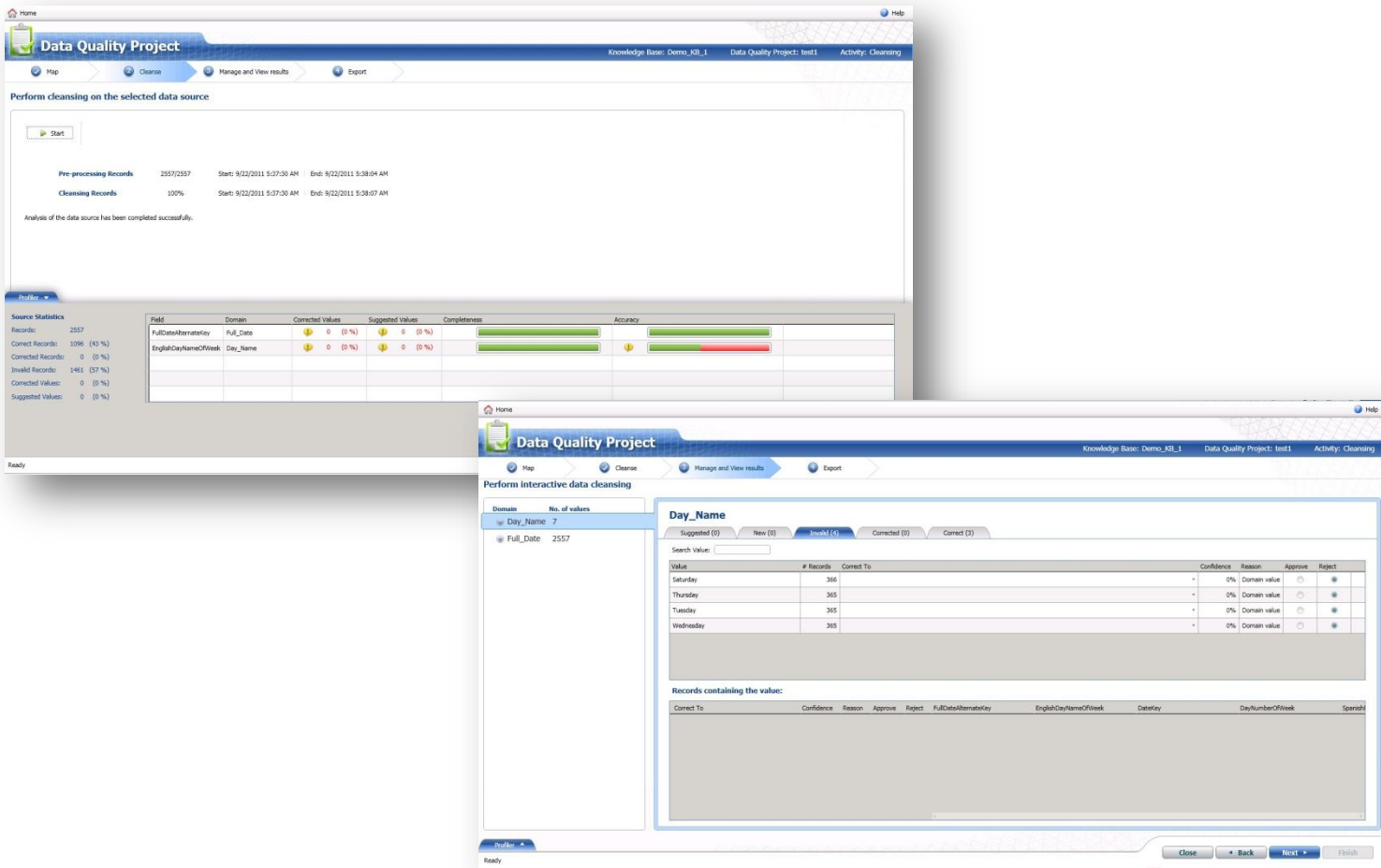


Figure 10 – Cleansing Result

C) View and Export Results

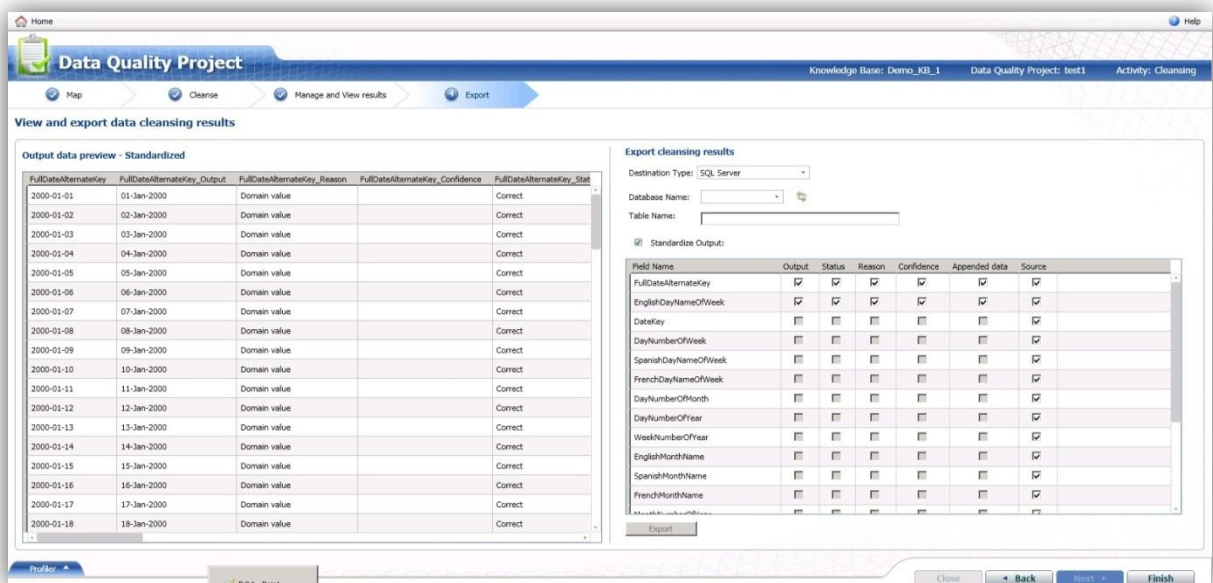


Figure : 11 – View and export results

Evangelism & Adoption

System

- ✚ It is advisable that to recommend the usage of DQS Denali as a best practice whenever processing data and it should find a mention in the best practices template. By doing so, we will be ensuring that it is followed at a global level in terms of test strategy implementation.
- ✚ At an individual level, Program Managers and Test Leads should include the DQ process using Denali at the time of planning the test cycle. Proper estimation and time should be assigned for this. Being part of test planning, Customers will understand the importance of the same and it will also awaken them to the value offered through this. It should be planned early in design and testing phase. Individual should be proactive when it comes to data.

Strategy

- ✚ In terms of strategy for DQS, the fact that it can benefit the ecosystem for not just clients but also for partners is an important element. Being a part of very much the mainstream processes for Test strategy execution across service engagements, DQS through Denali offers an easy-to-use and out-of-the-box feature for our partners to leverage the existing stack of SQL Server services to deliver more value to their customers.
- ✚ The second part of the strategy is then ideally to promote and exhibit the benefits of using DQS in Denali – awareness of the same alone can help partners and our consultants in not just using but also in helping promote the same in their project execution teams.

Sustenance

- ✚ Sustenance of both the strategy behind promoting DQS adoption and the integration of the same in test execution cycles can be furthered through proper knowledge management – in other words, standard support and query-resolution portals to assist users in their tasks.

Closing Thought

Denali DQS is in its first avatar and there is lot more to come in the future. We (the test community) should be ready to leverage its capabilities. The expectation from the readers is to seriously consider Denali DQS as an important tool in their armoury when they think about testing and automation. We hope this paper would have given the necessary head-start and if you like the idea then evangelize it across team and communities to change the way we think of data in testing world.

References

1. http://social.technet.microsoft.com/wiki/contents/articles/3711.aspx#Data_Quality_Services
2. <http://social.technet.microsoft.com/wiki/contents/articles/3919.aspx>
3. <http://blogs.msdn.com/b/dqs/archive/2011/07/17/installing-and-configuring-data-quality-services.aspx>

About Raj

Raj Kamal is a Senior Test consultant specializing in different types of testing techniques, test automation and testability in different domains like Manufacturing, Healthcare and Higher Education. He holds an APICS certification in Supply Chain Management. Familiar with Rational, Mercury & Microsoft testing tools, he has helped teams develop test automation strategies and architectures for such companies as Cognizant Technology Solutions, Oracle Corporation & Microsoft. He also provides training in automated testing architectures and design. He is QAI(CSTE) & ISTQB Certified. He has a master's degree in Computer Applications. He is currently working as a Test Lead at Microsoft, India, Business Intelligence domain.

His passion: <http://geektester.blogspot.com/>

Past presentations:

- Presented a webcast at Microsoft worldwide customers for Data quality Testing
- Represented Microsoft at QAI Software Testing International conference, Florida as a speaker "*Application and Script Independent Automation Framework*" (<http://www.qaiworldwide.org/qai.html>)
- Represent Oracle Corporation as Speaker on using "*Mapping Rational Unified Process with Rational Testing Tools*" at International Forum for Software Testing Professionals <http://ifstep.org/>
- Represented Microsoft as a speaker at *Test 2008 International Conference*
- Automation framework designed is accepted for Presentation and Publishing at International Conference on Information Technology: New Generations, Las Vegas & IEEE Digital Library

Publications:

- Published a paper on Resurrecting the Prodigal Son--Data Quality (Stickyminds)
- Published a paper on "Adventure of BI/DW Testing" (Stickyminds)

An article written on "Business Value of Testing" is published in Satyam Test Magazine

About Gunjan

Specializing in blue-ocean testing scenarios such as web-testing & automation testing in new terrains, Gunjan Jain is an Associate Test Consultant with Microsoft Corporation. She has worked on cutting-edge technology projects since the very start of her career – at Wipro Technologies and then with Sungard before joining Microsoft. Apart from test consulting, she also takes interest in development and deployment strategies and is a Microsoft Certified professional in both TFS and .NET platform. Web application development is also something she still tries to find time for – having done multiple corporate websites during her engineering days at Bhopal.

She blogs avidly and is a contributor to thought-leadership programs like Microsoft Thinkweek, Technovation & TechReady

Appendix

DQS : Data Quality Services