**“Three Your Data” (Three – Free – Get it?)**

|  |  |  |
| --- | --- | --- |
| **Scene 1 – Introduction**  What is the business scenario and problem? | **Jochen :** Hi Ken, master of IT Legacy Systems!  **Ken :** Hi Jochen, the Glorious CEO of our National Kensington Bank . What can I do for you?  **Jochen** : I’ve just acquired those 'New Insurance Company' people. I think there’s a very good chance that our share price might double in the next 24 hours. That’s because, Ken, we’re going to focus on the customers who make us money. Those customers who keep having accidents or who have nothing in their bank accounts will be seeing some pretty sharp increases in costs, or ( menacingly ) maybe they’ll leave us all together. \***mwahahahaa**\*  **Ken :** Brilliant idea Sir!  **Jochen :** So I just need you to get me a list of all those customers. \****pause\**** I can wait here while you do it.  **Ken :** Errr... There's going to be a problem with that Sir  **Jochen :** Problems! You IT legacy systems people, you’re always telling me there’s a problem. What is it this time? I'm an important CEO and I need that information!  **Ken :** Well that information is in two \***completely\*** different databases. Even worse they’ll probably be organised differently, in different schema. I just can’t run a query that will compare the data in two different systems.  **Jochen :** But I need it by lunchtime! How long is it going to take?  **Ken :** Well these data unifications exercises usually are supposed to take three months, but they usually take around two years if they finish at all.  **Jochen :** That’s not exactly lunchtime, is it, Ken from Legacy Systems? Think man ,think!  **Ken :** What if I ask Adam from the Future of IT department?  **Jochen :** You mean mean the handsome devil who’s just installed MarkLogic 7?  **Ken :** Yes sir  **Jochen :** With it’s enhanced support for semantics, cloud based deployment, tiered storage, automated rebalancing and many other Enterprise NoSQL things?  **Ken :** That’s exactly who I mean … | Critical to the success of business in the 21st century is being responsive to changing business needs and costs. The quicker merged companies can save costs, the more profitable and successful an acquisition will be.  Legacy integration is very costly. You need to be able to pull in information from disparate data sources, without spending months on schema redesign. |
| **Scene 2 – Adam Helps**  Show how to import and query data.  Also show inferencing – finding joint customers. | **Ken :** So Adam, I’ve got this terrible problem. National Kensington Bank have just acquired New Insurance Co. CEO Jochen from the boardroom wants a list of all our joint customers, so he can get rid of the unprofitable ones.  **Adam :** So?  **Ken :** So that’s going to mean unifying heterogenous schema from disparate data sources. Here's the two schema look! They're nothing alike! How am I going to get this done! I'll be fired by lunchtime!!! \***sobs**\*  **Adam :** Well it’s a good job you came to me Ken! It just so happens I've been importing claims Documents from the New Insurance Company. I've also set up connections to those two databases. Lets see what we can do. There's a vibrant Open Source community with MarkLogic. Someone has developed this easily re-usable rdb2rdf thing that should do exactly what you need!  **Ken**: **\*sounding relieved\*** That sounds great!  **Adam**: It sure does, Ken. What evens better, it uses the W3C Open Standard for direct import of RDBMS data to a triple store. And you know how we're all about using Open Standards. Lets see now...  *Show the rdb2rdf import screen and list the databases*  **Adam**: So which databases do you want?  **Ken**: The NewInsuranceCo and NationalKensingtonBank ones  **Adam**: OK I'll import everything in the NewInsuranceCo one first  **Ken**: Yeah right. Like you can just import data as-is!  **Adam**: Watch me, Ken.  *Import of all tables of NewInsuranceCo runs*  **Ken**: Wow that was quick!  **Adam**: You betcha Ken! Now for the other one...  *Import of NationalKensingtonBank*  **Adam**: Ha! Done! Now lets take a quick look  *Show sparql widget of query for all 'NIC Client' entities, and click on one to view their info, and their relationships to other tables*  **Adam**: Here you go you see, all the data, ready to explore!  **Ken**: Nice! … but what about just finding the joint customers. | Creating a merged schema and moving systems over takes months, if not years. Often companies just replace systems rather than make use of existing, valuable data.  Open Standards means there are people with common, existing skills you can use, reducing vendor lock in and increasing productivity of your IT team.  Importing data as-is, to either the document database or triple store, allows you to take decisions and implement new systems quickly, reducing your bottom line and giving you competitive advantage. |
| **Scene 3 – Inferencing**  Discovering new facts from existing information. | **Adam**: Well, lets run a query that creates Joint Customer entities for people with the same DoB and names. This adds a bunch more triples in to MarkLogic. This means our future queries will be much simpler – you just search for 'Joint Customer' entities and go from there!  **Ken**: Hey that kinda looks like SQL!  **Adam**: Yes, well. If you like that sort of thing I suppose. I'm more of a Subject-Predicate-Object man myself. As you see, we're linking all National Insurance Company clients to National Kensington Bank Customers.  **Ken**: Easier than creating a whole new schema! That takes months!  **Adam**: Yeah I bet. And the beauty is we're not altering the existing data – just adding new facts.  *Inferencing query executing in the background*  **Adam**: Right we're done. Lets just find join customers.  *Executes SPARQL UI widget query for all Joint Customer entities. Clicks on one and shows links to both original customer objects*  **Adam**: You see Ken, from here we can search for all related information. What else did you need? | Rather than write spaghethi SQL you can infer simple facts from complex queries, then use those simple facts in the future. This is called Inferencing and is the basis for a lot of semantic solutions.  Storing these inferred facts rather than carrying them out on the fly increases query performance.  Being able to browse all related data in a user friendly interface is key for applications aimed at business people. |
| **Scene 4 – Dodgy customers**  Combining information from multiple data sources and formats to drive business insights. | **Ken**: I need to find dodgy customers – so those with Kensington National Bank balances less than GBP 50 and with expensive claims... The most expensive being insurance claims involving pedestrians. Oh but wait, they're in documents, not a database! That's totally different types of data! We're stuck again.  **Adam**: Calm down dear, we can totally do that!  *Edit SPARQL query so that you search for JointCustomers with NKB Accounts with Balances < 50. Executing this also shows related documents to those Customers*  **Adam**: OK so here are your poor joint customers. Now lets combine this with a content search to find the list of all customers with claims documents with the word 'pedestrian' in them.  *Adds 'pedestrian' as a word query*  **Adam**: See Ken, done. Now I click download here and you get a CSV file for the CEO's Excel viewing pleasure! Job Done!  **Ken**: Job done indeed! I've never seen software so flexible and powerful! It's a miracle.  **Adam**: No Ken, it's Enterprise NoSQL in MarkLogic 7 – Time for us to get in the fast lane! | Merging facts (triples) and documents in a single search environment is unique to MarkLogic's Enterprise NoSQL database. This enables you to drive decision making with ALL the available data, no matter the format or source, and to do so quickly as required when new data sources become available. This reduces IT landscape complexity, and cost of creating new business insights. |
| Scene 5 – credits, smily faces, etc. | **Ken :** I’ve got that data you wanted sir.  **Jochen :** Good man. I knew I could count on you.  **Ken :** At first I thought what you were asking for was really hard, but Adam helped me realize …  **Adam :** It was just a matter of semantics  **Jochen :** You guys … |  |