PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

1: CRM Implementation Best Practices â€“ Part 2, Planning and Budgeting | ContactEase Blog

Last week we shared CRM Implementation Best Practices - Part 1, Getting Buy-In. This is our CRM five part series: 1) Getting Buy-In, 2) Planning and Budgeting, 3) Implementation, 4) Data Management, and 5) Impact and ROI.

By Jim Czuprynski Synopsis. Oracle 10g Release 2 10gR2 has improved significantly the methodology for tracking performance metrics within the database. In this article, I will spend a lot less time on theory. Instead, I will concentrate on how simple it is to set up adaptive thresholds with some actual examples of how to create metric baselines using both static metrics and rolling time periods. After I click on the Enable Metric Baselines button, Enterprise Manager requests a final confirmation; then, when I click on the OK button, metric baselines are automatically activated. Creating a Metric Baseline My next step is to create a metric baseline. Oracle 10gR2 allows me to do this using either statistics gathered from a rolling time period immediately prior to today, or from a static but specific range of dates. On the other hand, to create a static metric baseline, I instead specify the name of the new static baseline as well as a range of dates. This date range must correspond to a minimum of seven contiguous days. The time period chosen should be representative of the type of adaptive thresholds that utilize these metrics. For example, if I know that my database has been performing in a relatively stable fashion during a particular time period, the corresponding statistics would be valuable metrics for comparison against future performance. Note that Oracle marks with a warning symbol any metrics that cannot be determined; also, no data will be displayed when I click into any of the "eyeglass" icons for each faulty metric. Even in this case, all is not lost! I could also change the variables that are used to calculate the time series for the metric static baseline. On the other hand, if sufficient data do exist, then Oracle will display a checkmark to indicate this. In this case, I have created a new static metric baseline, MB1 using all available data for the time period between March 12, and March 21, I can simply click into each of the "eyeglass" icons to see how Oracle 10gR2 will calculate the values for the metric baseline. Clicking the OK button on this page completes the creation of the metric baseline, establishing it for immediate use with adaptive thresholds. Remember from the previous article that Oracle 10gR2 permits the creation and retention of multiple static baselines for use by adaptive thresholds. As shown in Figure 2. Oracle then passes control to the Manage Adaptive Thresholds page Figure 2. From this screen, I can set specific adaptive thresholds for each of the displayed metrics. In this example, I have selected just the Number of Transactions per second adaptive threshold for editing; once selected, I can then set up adaptive thresholds for the metric by clicking on the Edit button on this page. If I choose the Significance Level option, Oracle will use percentiles to determine when an adaptive threshold has been breached. The other option, Percentage of Maximum, Oracle will determine when an adaptive threshold has been breached based on the specified percentages, using the 99th percentile value for the calculation. Note that in either case, I can also specify the total concurrent number of threshold violations that must occur before an alert is raised. Note that there are two distinct sets of adaptive thresholds: For each of the three categories of baselined metrics Performance Metrics, Workload Volume Metrics, and Workload Type Metrics , I can choose and position three different metrics for display in the deflection graph. For this example, I have selected the nine metrics as shown in Figure 2. I accessed this page by clicking on the Baseline Normalized Metrics breadcrumb from the Metrics Baseline page see Figure 2. Since all of the metrics have been selected, Oracle 10gR2 provides a series of graphs that show the deflection from the established adaptive thresholds for the metrics. The most interesting feature of these graphs is that they have no X-axis; indeed, they all essentially use the same axis because they really represent the common deflection as measured across all adaptive thresholds. Observing the Deflection Note that the graph that represents the Number of Transactions metric shows virtually no deflection from the established adaptive thresholds, while many of the other metrics do indeed show some apparent deflections. I first executed the SQL code in Listing 2. In both cases, this caused an obvious and observable deflection in the graph on the Baseline Normalized Metrics page for the Number of Transactions per second graph as shown in the highlighted section of the graph in Figure 2. When I executed the same SQL code with a value of six 6 at 8: This is the essence of adaptive thresholds: One final point of interest on the page that displays these

deflection graphs: I can also "tune out" even more of the "white noise" for the adaptive thresholds in force by selecting the High setting instead of the Medium or Low setting for Noise Reduction as shown in Figure 2. These settings tend to filter out even more of the less-noticeable deflections, thus yielding a more obvious view of the true "spikes" that have been detected so far. Caveats Finally, a warning: Oracle strongly recommends gaining experience with adaptive thresholds in a test environment before implementing them in a production environment. Since these thresholds are more complex to understand than regular arithmetic thresholds, and since it is possible to tune out "noise" accidentally, it is not impossible to configure them incorrectly the first time out. I also heartily recommend running a representative workload against the test environment to insure that threshold violations are being trapped as expected! Conclusion With the addition of Adaptive Thresholds to my arsenal of DBA tools, Oracle 10gR2 lets me monitor several critical performance metrics for my database based on their significance as of a specific point in time. With a few simple mouse clicks in Oracle 10gR2 Enterprise Manager, I can easily construct the necessary baseline metrics for either a specified historical time frame or based on a rolling time period for the last several days, weeks, or even months. Once the baselines are established, a few more mouse clicks set up the adaptive thresholds that Oracle 10gR2 then monitors for violations. These adaptive thresholds help to eliminate those irritating "false positives" that plain arithmetic thresholds tend to detect erroneously, but they also tend to detect "near-miss" violations that plain arithmetic thresholds most often ignore. Actual implementation of these features should commence only after a crystal-clear understanding exists.

PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

2: The Oracle 10g Scheduler, Part 2: Implementation - Page 2 â€" www.enganchecubano.com

Part 2: process In this part, we will describe a few ideas about test data management plan implementation. Please refer to the previous part of the article for detailed information about the plan.

The project follows the architecture depicted on a previous article , but reading that article before this one is not necessary. For easy referencing, here are the articles in the series: Implementing Redux on Angular this article Case Study pt. The project is complete on my repository and these articles focus on the implementation of the business logic using Redux on an Angular web app. A snapshot of the web app implemented for this series of articles. Source code on my GitHub repository. On this article I will explain how the project logic is structured and implemented. Some Redux knowledge is assumed and expect no Angular details here. Besides this module is lazy loaded to test its full modularity. Declares all the UI components and make all services available. Normal angular module stuff. Defines the URLs that load the tree different pages: Not important to this article, so see more on that repository. This file is the only place where this module know about its exterior. Besides this file, there is no dependencies to other modules in the app. All the exam module code access the Redux store by injecting with this token, this way it has its own isolated state provided by featureSelector: The top level includes only two properties: Created by the app outside the exam module for storing the navigation state. The top level for the exam module state. That object has two properties: These two properties are defined by reducersMap: It could have its own file, but for now that seems an overhead. Note the import statements. Each state portion defines itself as State leaving to the code that imports it, to rename it matching the property that will be used to store that state. Speeding up eventual changes to the State interface. Notice that the payload in the case above seems to be an overhead, as it could be of type ExamStatusdirectly. I always use an object for the payload. This option is justified by: The other reducers are similar and can be seen on the repository. Effects As we have seen in the former article, the effects implement much of the business logic of our app. The procedure is about detecting the action that triggers the effect and emitting new actions in a kind of response. This effects code can make calls to asynchronous services, including server access. The effect for the exam start is more complex and its logic would get in the way of explaining the role of effects in the app logic. Recalling the planning notes, we have: In response to ExamEndAction, the code emits an ExamStatusAction immediately and then takes the state for the exam and questions info and filters for their availability. The evalQuestions function asynchronously computes the score. It is a bit more complex, but it really is just more RxJS , fetching questions info from server and producing a timer that can be interrupted if the exam status change in the Redux state. I will not go through that code, since my purpose here is not to show how to use ReactiveX to solve problems, but to show how to apply Redux effects to implement some logic around state and services. The only main difference from the exam start and end effects above, is that besides triggering on an action, they inspect the action payload to match the effect. They need to know if the navigation is to a specific page. See more on that repository. Summary In this article, we just saw how to implement business logic in a Redux with effects architecture. I think the only complex part is the use of ReactiveX to kind of declaratively implement what normally is thought out in an imperative way.

PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

3: A Roadmap to Pre-Implementation of Electronic Health Record: the Key Step to Success

The CPM Implementation Plan Template (Plan) is designed to guide your county through the process of decision-making around specific action items to reach your unique implementation priorities. Based on priorities identified in the Snapshot process, counties choose a planning horizon and match priorities with implementation objectives, action.

Maryam Ahmadi, Associate professor. This article has been cited by other articles in PMC. Initial attempts are made to implement electronic health record EHR in Iran, the present study aim is developing a proper roadmap to EHR in pre-implementation phase by expert views on the matter. An initial framework was developed for preimplementation based on reviewing literature on EHR implementation and descriptive-comparative with qualitative approaches in five countries. A questionnaire was designed in several sections using 47 main topics associated with pre-implementation. Expert views were used to score each topic based on Likert scale and opinions were collected through Delphi. Then results analyzed using descriptive statistical analysis method. Framework of EHR pre-implementation roadmap was presented in four phases: Priorities in each part were determined and reflected in the roadmap using expert views, analyses, and requirements of each phase. It seems strategic planning at different levels; assessment of needs; providing and managing financial resources, setting standards; examining the existing condition was determined as highest priorities in above phases. For successful implementation, developing national carefully-designed and well-documented EHR pre-implementing roadmap, Based on country situation, from strategic to operation level is necessary. As such, a roadmap should be developed for proper nationwide implementation of EHR based on such considerations as dynamicity and temporal and spatial requirements 2. Success in e-health depends on successful implementation of electronic health record 3. The pre-implementation is an important phase which focuses on activities that can facilitate success in the project 4. EHR roadmap is generally developed at national level. Different views propose that once the need for EHR is established, a minimum of essential elements must be included: Assessment, in particular readiness assessment 5 , 6 , 7 , 8 , proper planning for implementation 6 , 9 , 10 , determining the need for EHR and selecting a vendor 10 , implementation and considerations for related support procedures, final assessment of the system, optimization, maintenance, and EHR support There are three operational phases to EHR implementation: Because of the importance structural, processes, performance aspects and uniformity systems of health care center for the health of the network connection 13 , pre-implementation assessment for decreasing failure rate is necessary. The present study aim is developing a proper roadmap to EHR in pre-implementation phase by expert views on the matter in Iran. METHOD An initial framework was developed for pre-implementation EHR based on reviewing literature in numerous library resources, relevant databases, interviews with Iranian experts and scholars, and descriptive-comparative with qualitative approaches in five selected countries, namely England, Denmark, Singapore, Canada, and the United States Countries with substantial experience and systematic actions in the field of information technology and EHR systems, or with significant progress and changes in recent years. Then elements of the initial roadmap for pre-implementation of EHR were incorporated into a questionnaire. Content validity, literature review, and advices provided by experts were used to confirm the validity of the data obtained in the comparative study. The components were designed along four dimensions preliminaries, assessment, planning, and selection of the system and evaluated by Delphi method. They were asked to identify appropriate components in each category and also to determine priorities and score these components for proper model of a roadmap in Iran, using a Likert scale 1, highest value, and 7, lowest. Thirty questionnaires were confirmed through Delphi and using the average scores for values provided by experts, priorities were determined for designing a proper pre-implementation roadmap. Delphi test based on identified elements indicates that all experts and scholars approved the above mentioned components. In preliminaries of the roadmap to EHR implementation, the average value of scores assigned by the experts to the main components 1, highest value, and 7, lowest ranges from 1. The highest percentage Average scores ranging in Preliminaries elements are from 1. Table 1 Open in a separate window In

assessment phase Table 2 , average scores assigned to total components range from 1. In the first quartile of scores, the highest percentage Table 2 Open in a separate window While no respondent specifically indicated disagreement to the components incorporated in planning phase, the results presented in Table 3 showed that among the twenty five main components of planning, the range of average scores assigned by respondents 1. Table 3 Open in a separate window In initial framework for selecting the system and vendor Table 4 the average score for the main components range from 1. The components of this phase also have the highest priorities based on their scores. Availability of an approved list of vendors in national level has lower average score, and therefore higher priority, compared to non-listed vendor. A huge majority of respondents placed assessment of vendors and the systems in the first and the second scoring level Table 4 Open in a separate window 4. Priorities were identified by scholars and experts. In preliminaries phase of EHR pre-implementation, the components were ranked from the highest to the lowest priorities P1P9 in the last column in the Table 1. In addition to the above mentioned items, based on views determined by the experts with regard to assessment this assessment was divided into two main parts: Although identified as the fifth priority, implementation governance, were considered of great importance by experts. Providing financial and human resources 20 ; requiring adequate participation from stakeholders 12 ; information governance including frameworks for privacy policy, access to information, and defining frameworks for confidentiality 26 , 28 and best practice in EHR implementation; requiring involvement from clinical and managerial sectors in developing and identifying success factors 12 ; making clear decisions and commitment to strategic goals 1 were identified as the five top priorities in implementation governance. A study carried out in on successful implementation of EHR, showed that after careful analysis, it is necessary to design a path to move forward, estimate project expenses, and identify IT solutions to clinical businesses Esterle and Kourobali believe government support for implementation, participation by stakeholders, emphasis on leadership, governance, vision and information governance e. Kotter stresses need for establishment of a system by identifying main opportunities, building a powerful team to direct changes, developing an vision and a strategic plan, linking different strategies, new prospects, removing obstacles, planning for planning, making required changes for improvement including policies that have not been anticipated in the prospect, and creating new approaches in the process McGowan and Cusack emphasized the need for developing standards for technology and formative assessment of EHR These findings are all in line with the findings of the present study. As far as assessment is concerned, priorities P1-P4 in this regard can be proposed in Table 2. With regard to the first priority, scholars emphasize identification of needs and priorities for personnel and stakeholders, and patients as well as technical and non-technical requirements. Ajami et al stated that assessment of readiness, as a part of assessment of EHR implementation, should be considered in the early stages of assessment. Main areas of readiness for EHR implementation include organizational culture, management and leadership, and operational and technical readiness All these findings are consistent with this research. In planning phase of the roadmap to pre-implementation EHR, priorities were identified and proposed by experts P1-P9 in Table 3. Once implementation plan is approved, an executive team is required to operationalize the implementation process. According to the experts in the field of support programs, the highest priority should be given to assessment pre and post implementation followed by creating a integrity framework e. An important point noted by the experts was focus on standards. With regard to standard setting, the components can be ranked in terms of priorities assigned by the experts: Lorenz notes that in EHR implementation, an organization needs to analyze the state of readiness by identifying core values, understanding overall organizational area of activity, identifying concerns of stakeholders and needs of end users, creating an vision, and identifying basic requirements in order to be able to accept the new system Esterle and Kourobali stressed planning for time, personnel status, stakeholder awareness, implementation approaches, change management, focus on required processes, commitment, complete realization of advantages, and involvement by researchers 19 , the latter being consistent with the findings of the present study. With regard to Selecting the system and vendor, priorities P1-P5 were given by experts to proposed in the last column Table 4. Different types of users should also be considered in determining the type of products and clinical requirements. Available hardware and what is operationally needed by clinical staff should be carefully examined These suggestions are in line with the

findings of the present study. A comprehensive roadmap, particularly in pre-implementation stage, is necessary. Diffused activities in this area seem to create obstacles in EHR implementation and to lead the process to failure in its very early stages in any country. Therefore, it is important to have a clear and well-defined roadmap characterized based on opinions put forward by experts in the same country. In this study with Developing framework for a pre-implementation roadmap is defined as the first step within four fundamental phases: It is suggested to implementing EHR in every country, competent authorities, like Ministry of Health, should appoint or establish an EHR agency and develop carefully-designed and well-documented strategic plans and roadmaps at provincial levels for complete successful. The evolution of the Electronic Health Record: A roadmap for realizing the vision. E-Health readiness framework from Electronic Health Records perspective. International Journal of Internet and Enterprise Management. Modelling and implementing electronic health records in Denmark. International Journal of Medical Informatics. Journal of Health Information Management. Medical Information Science Reference; Best practice for implementing EHR and information systems. Human, social, and organizational aspects of health information systems; pp. Health and Medical Education. The Ministry of ; [updated 5 Feb ; cited september]. Electronic Health Record in Iran. American Hospital Association; The Road to Meaningful Use: Critical areas of national electronic health record programs - -Is our focus correct? National Health Services; Barriers and success factors in health information technology: Esterle L, Kouroubali A. Executive agency for health and consumers. Canada Health Infoway Inc; What matters when introducing new information technology. Communications of the ACM. California Health Care Foundation; Report of the council on medical service Contract No.: CMS Report 10 - A Assess Your Practice Readiness. Primary Health Care Transition Fund. A guide for clinicians and administrators. A step by step guide for medical practice:

PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

4: Migrating Datacenter to Azure â€“ Part 2 â€“ Hybrid Cloud Best Practices

Project Implementation Plan For successful project management, there is a need to have a good project implementation plan, which, is a single document that specifies the duties, dates, and who is in charge.

Migrating IT systems and infrastructure to a private or public cloud can be a challenge even for the most seasoned IT professional. At the same time, it is the moment to revise, rethink and improve architecture and concepts. The most common strategic drivers behind moving a datacenter to the cloud include: A datacenter migration must start with very careful planning and a phased approach to execution. Although the ease of entry to the cloud invites to easily rush in without a solid plan an inadequate cloud architecture almost always ends up preventing organisations from realizing the benefits that prompted the migration in the first place. Also, a datacenter migration is a strategic undertaking that must be executed without causing a significant impact on business operations, service delivery, performance, and data protection requirements. This post is focused on the process and most essential design decisions towards the transformation of the on-premises environment into Microsoft Azure. Implementation specific topics, such as prescriptive guidance on Azure Resource Manager templates, Azure Network Security Group design or Azure storage configuration are not addressed within this post. This post consists of multiple sections that do not need to be read in sequence and which may be referred to as required to assist in planning and migration activities: The Landscape and Feasibility. This chapter summarises the key areas to address when evaluating the business feasibility of moving the managed services provider platform into Microsoft Azure. Transformations or migrations are not accomplished Customer functional and non-functional requirements, platform requirements, and other dependencies force typically demand a phased approach to migration. This means that both the source and target environments will be active concurrently for at least the duration of the migration. A thorough understanding of the source environment is required when moving a workload, application or service from the source environment into the target environment. Dependencies within and between workloads must be identified to assess the impact of migration and plan an appropriate approach. Designing the Target Environment. This section will address considerations for planning the Microsoft Azure environment and provide examples of logical and physical architectures and approaches to leverage the agility introduced when embracing public cloud services. This section focusses on different approaches to moving services into Azure subscriptions, the approach to project execution as well as roles and responsibilities within the migration project team. This post has been prepared for Managed Service Providers seeking to move their shared services and tenant-specific workloads into Microsoft Azure. Many of the concepts and recommendations within this post may also be applicable to other business types including, but not limited to Systems Integrators and traditional datacenter hosting companies. Typically, MSPs host multiple tenants customers within their datacenters and these tenants share networking, storage and compute resources. Managed Services Providers also typically provide shared or common platform services for tenants including directory services, messaging services, publishing services, service delivery services and file services. It is the shared nature of these common services that introduces complexity, as the significant size of these environments prevents them from being transformed overnight. At any point in time during this transitioning cycle, there are two platforms that need to be operated and maintained concurrently. When evaluating these parameters for transformation or migration planning it is essential to understand the overall process and steps that should be followed. Figure 1 provides an overview of the steps involved, and further details are provided elsewhere in this post. Utilising the Microsoft Cloud Solution Provider CSP program, each Azure subscription provisioned under program the must contain resources utilised by one and only one tenant. Not only is this a requirement of the CSP program terms, but the approach also provides security, billing and operational benefits. Role-based access control may be configured at the subscription level to secure Azure resources. All Azure resource consumption within a CSP-based subscription is linked to the subscription, enabling the service provider to readily identify the tenant with which the consumption is associated. When requesting support from Microsoft, service providers can create service requests on behalf of their tenants that are linked to a specific tenant and subscription. Most

MSPs will prefer to licence Azure through the CSP program as it allows them to build a closer and higher value customer relationship. Although Microsoft offers the CSP program as the primary cloud channel program for service providers, in some cases a hybrid model that combines both EA-based subscriptions and CSP-based subscriptions may be required. From a technology perspective, there is no major difference since both environments are based on Azure Resource Management ARM deployment concepts. However, if tenant customer resources are limited in number and rely heavily on shared services, setting up dedicated subscriptions for each tenant then configuring connectivity between the tenant environment and the shared services environment may not be cost effective. There are a few considerations to be taken into account, but the most important one is if the tenant requires direct access to the resources such as a remote desktop session RDS directly hosted on a server system. This is not preferred in a shared environment since it has the potential for compromising security. If the customer is only consuming Software as a Service SaaS services through a UI, web services or a shared remote desktop session, that might warrant locating these single server systems into the shared services environment eliminating the overhead and cost of egress traffic from tenant environments into the shared services. Cost Optimization and Planning Many Managed Service Providers have adopted virtualisation technologies to optimise the usage of their compute, network and storage resources. Often the resource allocation for workloads hosted in these virtualisation environments are not optimised and more resources eg. CPU cores, memory, disk are assigned to workloads than what is necessary. In addition, virtualisation infrastructure will most commonly include redundant hardware to provide high availability for virtualised workloads. When transitioning to a public cloud service, the approach to provisioning of workload resources must change. The public cloud does not provide hardware resilience in the same way as on-premises virtualisation platforms may. Workloads should incorporate availability measures at the application layer rather than relying on highly available infrastructure. Service providers have limited control over maintenance cycles for public cloud infrastructure but are still required to meet service levels with their customers as described by their managed services agreement. There are several considerations when planning for workload migration to the public cloud to ensure cost optimisation and an ongoing ability to deliver managed services according to service level agreements: It is of utmost importance to understand the actual resource requirements of a certain workload, application or services within managed services. By understanding specific workload requirements it is possible to allocate sufficient public cloud resources without over-provisioning and incurring unnecessary additional cost. Using resource definitions equivalent to an existing on-premises environment and adopting them unchanged in the public cloud infrastructure often leads to higher cost, thereby affecting the bottom line. Not having an underlying highly available compute infrastructure for workloads means that service availability must be accomplished at a workload level. Single server configurations must transition to multiple server configurations to provide continued availability of managed services in case underlying infrastructure is affected by issues or maintenance. Microsoft Azure Availability Set resources can be used to ensure virtual machines are distributed across multiple Fault Domains. Increasing workload availability through the addition of virtual machine instances may increase operational costs and thereby impact the business case for virtual machine migration to Microsoft Azure. One approach to managing this impact is to appropriately scale virtual machines to meet workload requirements whilst adding additional instances to enable a workload to survive the loss of one or more virtual machines. Load balancing across virtual machines can be accomplished either at the application level by workload-specific mechanisms, or through the addition of infrastructure-level load balancing resources such as Azure Traffic Manager, Azure Web Application Gateway or Azure Load Balancers. Other approaches to providing workload resilience include utilising replication capabilities or cluster capabilities within a workload. Replication is unlikely to result in cost optimisation, but may address requirements for ongoing service availability. Even if a decision is made to implement a single virtual machine instance, Microsoft recommends that single virtual machine instances are deployed in their own Availability Sets. This approach facilitates the addition of additional virtual machine instances in future if required, through their addition to the same Availability Set. By including the initial virtual machine in an Availability Set from the outset, a service interruption to join the virtual machine to an Availability Set when subsequent virtual machine

instances are deployed is avoided. Virtual Machines with Licenced Software: The usage costs for VMs using these images will be higher than the cost of a VM deployed with an image including only the operating system, as a component of the usage cost is attributed to the licence for the included software. When deploying virtual machines it is important to consider whether to utilise images with pre-installed software, or whether to install application software after VM deployment and leverage other licencing models for the application. The cost optimisations outlined above represent the "low hanging" fruits when planning a migration of an on-premises solution to Microsoft Azure. Additional complexities arise during migration planning when current platform capabilities are mapped to resources available in Microsoft Azure. These complexities are not address in post, but are instead discussed in some detail in Microsoft Azure online documentation and corresponding blogs. Planning For Coexistence Migrating managed services into Microsoft Azure is often unlikely to be accomplished overnight or over a weekend due to the size of the environment being migrated or the number and duration of verification steps used in the migration process. To accommodate a phased migration it is necessary to plan for a period of coexistence in which some services continue to run on-premises whilst others are running in Microsoft Azure. When planning for coexistence the following topics should be considered: During a period of coexistence the bandwidth needs of the on-premises environment will fluctuate. Initially, additional bandwidth may be required to accommodate communication between services hosted on-premises and those running in Microsoft Azure. Over time, the bandwidth requirements for on-premises systems will decrease as they are progressively migrated to Microsoft Azure and client applications access these through endpoints published in Microsoft Azure. Estimating bandwidth requirements and how these will change over the period of coexistence is important but can be a challenging task, particularly when the on-premises network is not well segregated or network components do no provide sufficient information for analysis. When planning for coexistence, Microsoft recommends deploying core services in Microsoft Azure prior to migrating services that are dependent on these. For example, consider deploying or migrating, when appropriate the following services in Microsoft Azure early in the period of coexistence: Active Directory Domain Controllers to provide authentication and authorisation services for subsequent application workloads. During the period of coexistence after core services are deployed in Microsoft Azure, consumption charges will accrue for these resources that are not yet in use. For this reason Microsoft recommends that the duration of this period is minimised. During the period of coexistence, core services will be running both on-premises and in Microsoft Azure. Once workload migration is complete, core services that are running on-premises and that are required only for workloads that have been migrated to Azure may be decommissioned. Once a hybrid scenario is established and core services are deployed and extended into Microsoft Azure, platform activation is the next step. Although there are several approaches that may be used, two of these approaches are described below. Reconfiguration of endpoints or clients should be avoided due to the effort, timeframes and costs involved in making these changes. Two possible approaches include: Introducing Azure Traffic Manager to provide global load balancing services will enable load balancing of traffic across services running on-premises and in Microsoft Azure. Azure Traffic Manager operates as a mediation layer between the end points consuming services and the platform publishing the services. If running two environments concurrently is not an option, Traffic Manager may also be used to switch between environments without requiring a change to public DNS records. Using Azure Traffic Manager in this manner to support migration is typically temporary and provides a controlled approach to direct traffic to migrated workloads. Initially, these DNS records resolved to the IP address of the workload on-premises, however after migration the DNS records are updated to reflect the addresses of the Microsoft Azure endpoints for the migrated workload. As this approach may result in a delay until all clients utilise the new IP address, performing a DNS switch is seen as the least controlled approach to connecting to migrated workloads. From the time the DNS record is updated, traffic may be directed to both the prior on-premises address and the address of the workload in Microsoft Azure for some time. Building the Inventory Platforms operated by Managed Services Providers are often large and complex, composed of shared services and dedicated or multitenant customer environments. As discussed earlier, these platforms are not typically migrated overnight or even over a weekend into public cloud platforms such as Microsoft Azure. These

transformations typically include a coexistence period during which workloads are migrated from on-premises systems into Microsoft Azure. To make efficient use of cloud economics and to be able to migrate a workload from on-premises to Microsoft Azure without affecting the services availability requires a comprehensive understanding of the workload and source environment. In smaller environments, a manual approach to compiling an inventory may be used, in which details such as server names, their current resource definitions and dependencies may be documented. In more complex or larger environments an automated approach to data collection is most often required. In these circumstances additional information including such as intra-application communication patterns and networking protocols and port information is also captured as this can be critical to planning a migration of workloads to Microsoft Azure.

PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

5: Capacity Planning Done Right for SQL Server - Database Trends and Applications

This second part of of Temporal tables in SQL Server

Customer Area Test Data Management. Please refer to the previous part of the article for detailed information about the plan. The first idea is about a number of required data sets or databases. How many test data sets do you need? Our answer is "one goal - one data set". Instead of, you can create small complex database populated by various data and large database with simple dependencies or without dependencies. However, we should explain "dependency" term in this case. The second idea describes data complexity that you really need. We recommend analyzing existing test cases and test packages. Your test database or test sets must contain at least different data rows for each test case. For example, if test case checks that account number is a positive integer value, add a few rows with 0, negative values and NULL to support this test case. Of course, the data complexity is not so critical for hardware testing except you analyze CPU loading for complex script execution. The next usual problem is how to get test data: In most cases, the answer depends on available resources. If the company has engineers and developers who can create test data generation scripts without third party tools and the total cost seems acceptable - it is a perfect way. This way guarantees flexibility but requires a lot of time for development. The alternate way is to create test data using "test data generation" software. It is a fast way but requires some kind of learning or adaptation. Also, the company must have enough budget to order the software. We do not recommend to do that without data masking or scrambling. However, modern test data generators offer data scrambling feature that replaces critical data names, account numbers, etc to randomly generated values. A few words about test data lifecycle. The company should use only actual sets of test data. That means the QA team has to rebuild test data if the goals, test cases or something else changed. It is a good idea to review test data arrays or databases after each modification of the test data management plan. It guarantees that all interested parties have right correct, actual, etc test data. The next part of the article will discuss personal roles in the test data management.

6: Oracle 10gR2 Adaptive Thresholds, Part 2: Implementation â€” www.enganchecubano.com

9 Great Ocean Road Region 2 Great Ocean Road Region 10 An implementation committee will be established to facilitate the implementation of the Great Ocean.

7: News, Tips, and Advice for Technology Professionals - TechRepublic

PROJECT IMPLEMENTATION PLAN: PART 2 3 documented. The project manager will also use the telephone in discussing different elements of the project. Interactive communication will be widely used between the team and the project manager.

8: Introduction to the National Youth in Transition Database Video Series | Children's Bureau | ACF

This is the second part of a series of articles about a case study for a web app implemented using Redux and Angular. The project follows the architecture depicted on a previous article, but reading that article before this one is not necessary. Case Study pt Planning business logic using Redux.

9: Test Data Management. Part 2: data management process

The first part of the article provides a common application settings information and described how to meet the COBIT 's PO - Integrity management control objective requirements.

PT. 2. DATABASE IMPLEMENTATION. PLANNING pdf

9780321643698 12th edition Cognitive functions of the cerebellum Spying for the Revolution Qbasic tutorial for beginners Jlg 600a parts manual The Black studies paradigm : the making of scholar activists Terry Kershaw The english language history First Middle English primer CHAPTER SIX: The Goddams withfeatures on Rouen, Bourges, Orleans, Domremy, Chinon and Reims 135 Funding Sources for Community and Economic Development 2002 Awaken Children, Volume 9 Journey of no return Synergetics of measurement, prediction, and control 21 day fix cookbook American history 2 matthew downy The Devils disciple, by George Bernard Shaw. Part IV: Wisdom in society V. 1. A phological introduction Human security and the UN : a critique. Cpr callblocker v5000 white list Kinds of variants in the manuscript tradition of the Greek New Testament Klaus Wachtel Advanced net debugging The american pageant volume 2 15th edition Antisemitism: The Generic Hatred Dom struggle of india quiz book Contemporary nursing issues trends management 7th ed The Mongol conquests, 1206-79 Titus Andronicus (The New Cambridge Shakespeare) Is your business making you money? Inefficient lobbying, populism, and oligarchy An Unexpected Family Product allocation is the new asset allocation. Compilation of laws relating to the Congressional budget process Language as an ethic Historical dictionary of Paraguay Return To Space:Flight Of Glen Basic finance and accounting concepts Measurement: a way to capture observation in mathematics and science Introduction a la psychanalyse Review of Marshall McLuhans Understanding Media