Journal of Information Systems Engineering & Management, 2:1 (2017), 1
ISSN: 2468-4376

Managing Database Services: An Approach Based in Information Technology Services Availabilty and Continuity Management

Leonardo Bastos Pontes[1]*, Adriano Bessa Albuquerque[1]

[1] *Universidade de Fortaleza*, BRAZIL

***Corresponding Author:** leonardo.bastos.pontes@gmail.com

Citation: Pontes, L.B. and Albuquerque, A.B. (2017) Managing Database Services: An Approach Based in Information Technology Services Availabilty and Continuity Management, *Journal of Information Systems Engineering & Management*, 2:1 (2017), 1.

doi: http://dx.doi.org/10.20897/jisem.201701
Published: January 5, 2017

ABSTRACT

This paper is held in the information technology services management environment, with a few ideas of information technology governance, and purposes to implement a hybrid model to manage the services of a database, based on the principles of information technology services management in a supplementary health operator. This approach utilizes fundamental nuances of services management guides, such as CMMI for Services, COBIT, ISO 20000, ITIL and MPS.BR for Services; it studies harmonically Availability and Continuity Management, as most part of the guides also do. This work has its importance because it keeps a good flow in the database and improves the agility of the systems in the accredited clinics in the health plan.

Keywords: availability, continuity, health plan, information technology services, management guides, service management

Pontes, Leonardo B. & Albuquerque, Adriano B. (2016). Continuity and Availability Management. Case Study: A Hybrid Model Applied in Database Services of a Supplementary Health Operator. Published in Information Systems and Technologies (CISTI), 2016 11th Iberian Conference on (pp. 657-660). IEEE. Gran Canaria – Spain.

INTRODUCTION

The contemporary business environment has a great dependency on the resources provided by de Information and Communication Technology (ITC) area, and, in this way, it is necessary to create/maintain a management of such ITC (ITSM – Information Technology Services Management) services, so those requirements are analyzed effectively. It is said (Bianchi, 2012) that the ITSM has as mains objective to allocate the available resources and manage them integrally; making possible for other areas in the organization to see the quality of the services, and, also, making the customers/users to see the quality of the acts of the ITC department.

Inside the subject of ITSM (SOFTEX, 2014), two major managements stand out harmoniously: Availability Management (AM) and Continuity Management (CM). This process of management seeks to ensure the quality of both management areas: availability and continuity are two valuable services characteristics. In any way, the continuity and availability may be confused among each other; it happens because availability refers to the feature that makes the service accessible and consumable in any time (the level is described in the agreement between the services provider and the consumer); the continuity refers to the ability of the services provider to maintain the agreed services online (Svata, 2013).

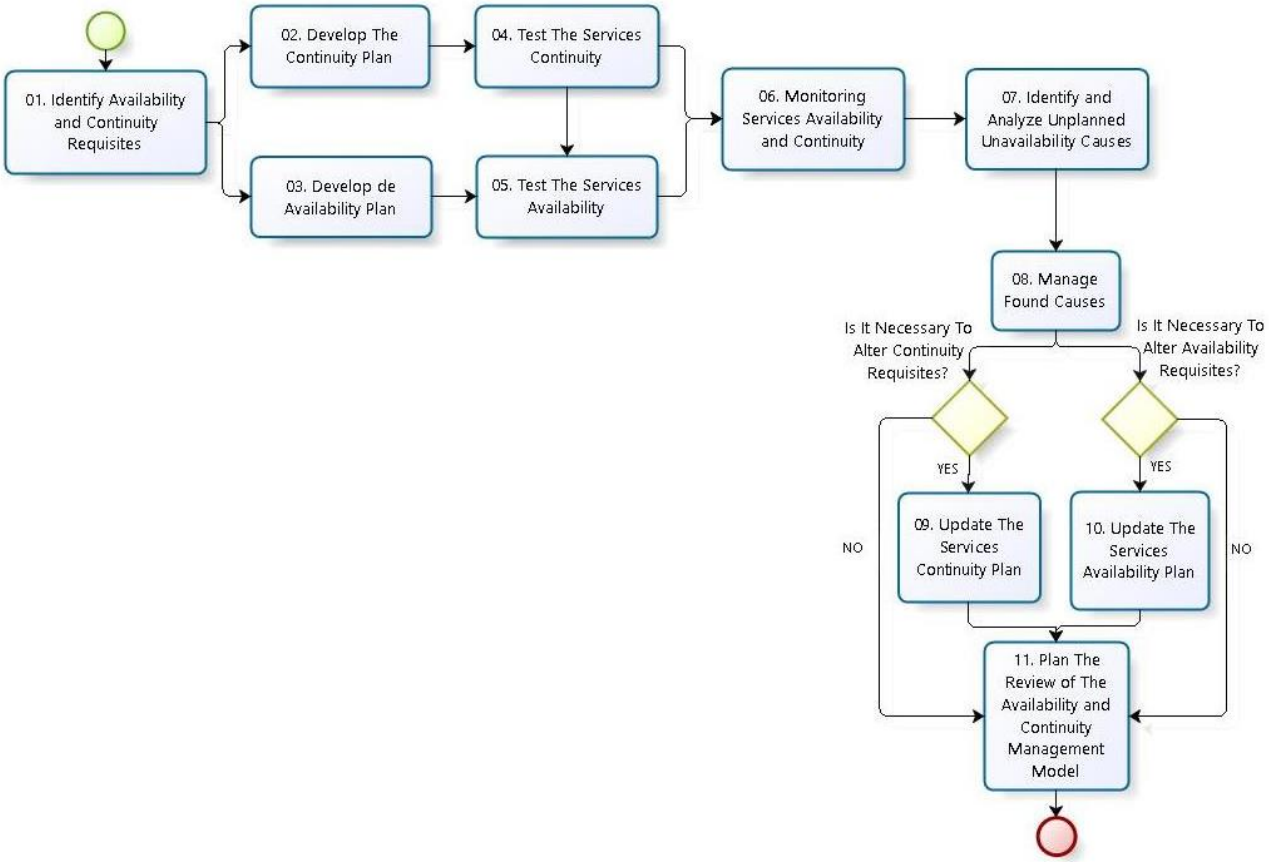

Figure 1. Hybrid model of continuity and availability management

There is no similar study in the area of health insurance/plan in Brazil, supporting the fact that this study is of major importance, that is, when the availability and continuity management is brought to the academic environment. Those sections are adjuncts because they seek to maintain the agreed SLA, in all thought scenarios; these processes are also responsible to ensure the customers and other stakeholders interests. This includes defining, analyzing, planning, measuring and improving all aspects of services availability and continuity; this involves de risk reduction to an acceptable and planned level to the recuperation of the service in the case of an existing interruption. Because of such importance and functionality, this study is also valid to business niches that utilize database technologies.

PROPOSED PROCESS

The model made and exposed in this paper seeks to deal with possible interruption in the services and systems in the medic clinics linked to the company that provides the health plan; focused in services availability and continuity.

Issues and the Process

The problems related to availability and continuity may be defined, synthetically as: multiple access and requests blocking the queries in the database. The link of the issues and services management is described as: referring to availability, the time of the existing block is associated to the unavailable time of the services, and the time to release the block is associated to the continuity of the services.

Management Model

The model made in this study is based in most recent studies, regarding IT services management (Barbosa, 2011), (Institute, 2014), (Cots & Casadesús, 2013), (Crisc, 2012), (SEI, 2010), and was created to manage the continuity and availability of the services (queries) running on the database, where: the least quantity and time of blocks, the more the database will be available, and the time to deal with the blocks will provide the continuity level. The model has eleven – 11– steps, described as follow, and designed in the **Figure 1**.

```
        -- BLOCKED QUERIES --
SELECT
    sess.session_id, -- blocked process
    sysp.blocked, -- blocker
    req.command,
    sess.login_time,
    sess.original_login_name,
    sess.login_name,
    sess.host_name,
    sess.program_name ,
    conn.client_net_address,
    sess.status,
    req.cpu_time
FROM sys.dm_exec_sessions sess
    INNER JOIN sys.dm_exec_connections conn
ON sess.session_id = conn.session_id
    INNER JOIN sys.dm_os_waiting_tasks tasks
ON sess.session_id = tasks.session_id
    INNER JOIN sys.dm_exec_requests req
ON sess.session_id = req.session_id
    INNER JOIN sys.sysprocesses sysp
ON sess.session_id = sysp.spid
    WHERE tasks.blocking_session_id <> 0
ORDER BY req.cpu_time desc
```

Figure 2. Command listing blocked sessions

```
    -- EXECUTING QUERIES --
SELECT sqltext.TEXT as Text,
getdate() as Data,
req.session_id as #Session_Id,
req.status as Status,
req.command as Command,
req.cpu_time/1000 as Tempo_CPU, -- time in seconds --,
req.total_elapsed_time/1000 as Total_Time, -- total time --
sess.login_time,
sess.original_login_name,
sess.login_name
FROM sys.dm_exec_requests req
INNER JOIN sys.dm_exec_sessions sess
ON req.session_id = sess.session_id
CROSS APPLY sys.dm_exec_sql_text(sql_handle) AS sqltext
where req.session_id = 3198 -- insert blocker SPID
```

Figure 3. Command listing information about session

Step 01. Identify the Continuity and Availability Requisites

In the case studied, the requisites were the number of blocked sessions (queries) and the time of each blocker. As said, the number and time of blocks refer to the availability, and the time of each blocking session and the time needed to drop them refer to the continuity level.

Step 02. Develop the Continuity Plan

The Continuity Plan has a template with the name of the responsible for the step (aka the Database Administrator, who is also responsible for every step), the version of the document, date and other informations about the company, but is focused in collecting the time of each blocking sessions, and the time needed to drop them. The **Figure 2** shows the query that lists the blocked sessions, and the **Figure 3** details a specific session.

Step 03. Develop the Availability Plan

The Availability Plan also has a template document, with the fields of the Continuity Plan. But, regarding the Availability, the number of blockers and the type of the sessions is important, because it will be able to create statistics along the time, saying, for example, how much insert queries blocked de database services.

Step 04. Test the Services Continuity

In this step the database administrator will force a moment of interruption, running queries in the database to create blocks and test de Continuity Plan. The blocks in this and next step (Step 05) will be executed in a controlled environment (development server), to not interrupt the services in the production server.

Step 05. Test the Services Availability

In this step, similarly to the last step, the database administrator will force a moment of interruption, running queries in the database to create blocks and test de Availability Plan.


```
IF (SELECT COUNT (*) from(
                -- BLOCKED QUERIES --
                SELECT
                sess.session_id, -- blocked process
                sysp.blocked -- blocker
            FROM sys.dm_exec_sessions sess
                INNER JOIN sys.dm_os_waiting_tasks tasks
            ON sess.session_id = tasks.session_id
                INNER JOIN sys.sysprocesses sysp
            ON sess.session_id = sysp.spid
                WHERE tasks.blocking_session_id <> 0
            ) as Quantidade) <> 0
    EXEC msdb.dbo.sp_send_dbmail
    @profile_name = 'DBA',
    @recipients = 'dba@domain.com.br',
    @query = 'SELECT COUNT (*) from(
                -- BLOCKED QUERIES --
                SELECT
                sess.session_id, -- blocked process
                sysp.blocked -- blocker
            FROM sys.dm_exec_sessions sess
                INNER JOIN sys.dm_os_waiting_tasks tasks
            ON sess.session_id = tasks.session_id
                INNER JOIN sys.sysprocesses sysp
            ON sess.session_id = sysp.spid
                WHERE tasks.blocking_session_id <> 0
            ) as Quantidade',
    @subject = 'Blocked Queries Count';
```

Figure 4. Blocked sessions notification

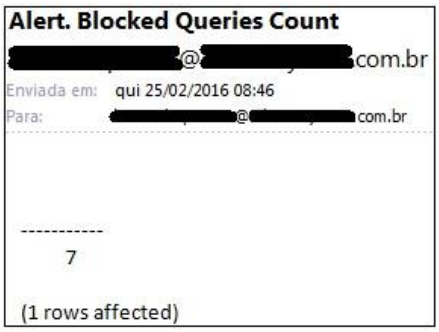

Figure 5. E-mail example

Step 06. Monitoring the Services Availability and Continuity

This approach alerts the DBA by email, detailing the existence of blocked sessions in the database, and the quantity of blocks. This feature is part of SQL Server Agent, and executes the query shown in **Figure 4** in every minute. The result of such figure is exemplified in the **Figure 5**.

Step 07. Identify and Analyze Unplanned Unavailability Causes

This step is not useful in this approach, because the database has a direct link with it, and the only way to lose connection would be a power outage; with a power outage, no customer or user would have access to the database.

Step 08. Manage Found Causes

This follows the same way of the last step.

Step 09. Update the Services Continuity Plan

If another approach is based in this one, and it could have another failure besides de blocks, this and the next step would be activated to meet the correct agreement.

Step 10. Update the Services Availability Plan

This step follow the same principles of the Step 09.

Step 11. Plan the Review of the Availability and Continuity Management Model

Having or not any kind of trouble to execute the model, it is of major importance to plan the review of the current availability and continuity management model; with this, it's possible to create new indicators and performance points to evaluate the quality of the model.

RESULTS AND CONCLUSION

There was no routine to manage de availability and continuity of database services in the company used in this study, then, it was not possible to create control indicators of success. Before implementing this approach, the

only way to discover if the blocks were interrupting the services was if the customer services reported; when it happened, the rollback time could take hours to be completed.

After the implementation of the approach, the DBA could take his time to deal with the blocker and blocked sessions. Following this line, the blocks that, before, took twenty minutes to be resolved; after it took only the maximum of two minutes. Such results validate the importance of this study, where, before there was a chaotic environment, now exists an environment capable of planned management and solutions.

The present paper proposes an availability and continuity management model, based in official guides, and applied to databases services of a company that provides health insurance plans. The model can manage any kind of database systems services, not only in health area, but also in data centers, virtual commerce, etc., as long as any features allowing the DBA to schedule the execution of the alert script, and features that makes possible the scheduler to send emails.

There is no published paper talking about information technology service management applied to database services, which is another validation of this work.

This approach was used every minute, during one week, which is a limitation, but also a point to improve; as another improvement point is that the queries can be edited, so the DBA can use the approach more freely.

REFERENCES

Barbosa, A.M. (2011). Governança em TI: COBIT; ITIL. *Revista Científica Eletrônica de Administração, XI*(19).

Bianchi, I.S. (2012). Gerenciamento de Serviços de TI: Desenvolvimento e Implantação de um Sistema de Informação para o CSE da UFSC. *Dissertação*. Florianópolis: Universidade Federal de Santa Catarina.

Cots, S. and Casadesús, M. (2013). Implementing ISO 20000: Proposals from Learned Lessons. *TMQ - Techniques, Methodologies and Quality, IV*.

Crisc, R.E. (2012). Comparing COBIT 4.1 and COBIT 5. *ISACA*. California.

Institute, C. (2014). *CMMI Institute*. (CMMI) Acesso em 2 de Junho de 2015, disponível em http://www.cmmiinstitute.com

SEI, S.E. (2010). *CMMI for Services - Improving Processes for Providing Better Services*. Pittsburgh: Carnegie Mellon University.

SOFTEX. (2014). *MPS.BR - Melhoria de Processo do Software Brasileiro: Guia de Implementação para o Nível C do MR-MPS-SV:2012*. São Paulo: Softex.

Svata, V. (2013). System View of Business Continuity Management. *Journal of Systems Integration*, pp. 19-35.

