Software Implementation of Online Assisted Platform for Computer-aided Design

Ognyan Chikov, Galia Marinova
Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd
Sofia 1000, Bulgaria
{ognyan.chikov@tu-sofia.bg}
{gim@tu-sofia.bg}

ABSTRACT: *In this paper the software implementation of Online assisted Platform for Computer-aided design in communications (Online-CADCOM) in Cloud environment is described. The platform is presented as multi-tenant Cloud application. Platform and database architecture are discussed as well as platform availability and security. Application lifecycle in Cloud is also described. The role of each aspect is widely discussed and also each potential problem is explained and solutions, applicable for Online-CADCOM and similar cases are proposed.*

1. Introduction

The developers and suppliers of Computer-aided design (CAD) tools were longtime reticent to direct cloud exposure of their platforms and security reasons were the most widely pointed for this state. Recently some attempts are made to propose online CAD tools on clouds, as the Mentor Graphics SystemVision Cloud platform [13], which proposes only typical models of different electronic and electro-mechanical components and reduced set of Spice-based analysis - frequency and time-domain. Different forms of feedback are provided as blog, chat, comments on designs. The current application of the platform is limited, some advanced options, which are necessary for some oscillator circuits analysis are missing. DesignSoft TinaCloud [14] is a commercial circuit simulator on a Cloud. OrCAD Cloud [15], proposed by Cadence is providing mainly project and design sharing, intelligent pdf files, applications to expand OrCAD Design Suit and documents and discussions on some new concepts.

During the last few years the Laboratory for ComputerAided Design in Telecommunications in Technical University-Sofia is developing an Online assisted Platform for Computer-aided design in communications (Online-CADCOM), described in [1]. The platform is developed on virtual server and new trends in the area of CAD tools motivated the transfer of Online-CADCOM in a cloud environment. This transfer raises a set of challenges which are considered in details further.

The software realization of Online-CADCOM is implementing Model-View-Controller (MVC) Framework approach for its development as explained in [2]. The platform is built of large number of different content – resources and connections to studied models and tools, knowledgebase materials, standards and how-to video materials that are used to help students for their study and work process. The most complicated part of the platform is the section with online tools which are classified based on Daniel-Gajsky Y-model of design levels as explained in [1]. Each tool has separate built in passport with accurate and consistent classification. In terms of implementing this model in software platform that would be hosted on Cloud service, consumptions and prerequisites are taken in order to implement this complicated model. Cloud based web applications have large number of architecture requirements for proper functionallity. The platform is managed by separate system that is developed using the same development approach using MVC Framework [3]. This system is called Content Management System (CMS). Both platforms are operating as one software solution which with the help of CMS represent an online application with reliable method of managing, updating and monitoring. Running software platforms that are built of large number of separate modules requires proper design and scalability of Cloud architecture. Online-CADCOM platform best fits in software as a service (SaaS) model of Cloud applications based on models explained in [4]. Online-CADCOM as a SaaS application provides users of the platform with centralized, network-based access to all platform resources with less overhead than when using an ordinary hosted web application approach. But in order to take advantage of the benefits of making Online-CADCOM as SaaS in Cloud, the platform is providing a level of control over its users' data. To earn this trust, one of the highest priorities for Online-CADCOM as SaaS model is creating the platform data architecture that is both robust and secure enough to satisfy tenants or clients who are operating in the platform, and also supply them with effective method for maintaining and administration providing best working times and low latency. The paper considers further the development of Online-CADCOM as a multi-tenant application, the Database partitioning and the availability in Online-CADCOM, the way to achieve a secure multi-tenant cloud application and the application lifestyle in cloud. The application of solutions proposed for the transfer of OnlineCADCOM from a virtual server to cloud environment will be useful for any other CAD platform to be transferred to a cloud.

2. Online-CADCOM as A Multi-tenant Cloud Application

Making the platform a multi-tenant application requires to re-design the database architecture of Online-CADCOM [4][5]. The platform is built with SQL Database, a centralizedand open-source relational database management system [1]. Furthermore using this approach gives high availability and database "sharding" to the solution. For making the platform SaaS Cloud based solution, it shold use one of the following database architecture approaches, so all functionalities that are provided initially by using SQL, to continue operating in Cloud as described in [6]:

• Table Architecture,

• Blob Architecture,

• SQL Database Architecture,

• SQL Database Architecture on Virtual Machines.

The best approach that is implemented for Online-CADCOM running SQL Database is by using SQL Database Architecture running on Virtual Machines in Cloud [4]. Choosing this approach for Online-CADCOM database is dealing with managing multi-tenant data as explained in [5] which is specific for the platform. There are a lot of Cloud solutions for multi-tenant databases partitioning including shared and isolated approach of tenants databases and containing tables, but choosing this partitioning scheme is based on a couple of considerations affecting the platform and its architecture [4]:

• **Security Considerations -** As Online-CADCOM store sensitive tenant data, users have high expectations about security, so the platform needs to provide strong data safety guarantees. A common misconception holds that only physical isolation can provide an appropriate level of security in the platform. In fact, data stored using a shared database approach can also provide strong data safety, but requires the use of more sophisticated software algorithms.

• **Tenant Considerations -** The number, nature and needs of the tenants that affect content in Online-CADCOM are many and varied. They are using big storage space in the platform and access to a wide software instances, so separate scheme for each tenant is a good approach, but at the same time data storage requirements may force to move to a shared-database model.

• **Economical Considerations** - Applications optimized for a shared approach tend to require a larger development effort than applications designed using an isolated approach (because of the relative complexity of developing a shared architecture), resulting in higher initial costs. Since they can support more tenants per server, their ongoing operational costs seems to be lower.

Taking into account the above considerations and also due to Online-CADCOM platform specification which is complicated and built of different categories, content and userlevel access, the best one is to use shared databases on Virtual Machine (VM) with separate schemas for every tenant. This approach involves housing multiple tenants in the same database, with each tenant having its own set of tables which are grouped into a schema created specifically for the tenant. Choosing this approach assumes that everything is virtuallized within the Cloud.

When a tenant profile is initially created in the OnlineCADCOM platform, the provisioning subsystem algorithm creates a discrete set of tables for the tenant and associates it with the tenant's own virtualized database schema, based on the type of access user level. Below are showed commands that are automatically sent to the virtualized SQL Database when new tenant is created in the platform:

CREATE SCHEMA UserNameSchema AUTHORIZATION Username

Then the platform creates and accesses tables within the tenant's schema using the SchemaName.TableName convention:

CREATE TABLE UserNameSchema.Resumes (UserID int identity primary key, Resume nvarchar(MAX))

After the schema is created, it is set as the default schema for the tenant account:

ALTER USER Username WITH DEFAULT_SCHEMA = UserNameSchema

A tenant account can access tables within its default schema by specifying just the table name, instead of using the SchemaName.TableName convention. This way, a single set of SQL statements can be created for all tenants in platform, which each tenant can use to access its own data:

*SELECT * FROM PersonalData*

3. Online-cadcom Database Partitioning

Online-CADCOM uses a tenant-base horizontal partitioning as explained in [4] due to the reason that many users would need access in same specific time. This approach is used because sometimes a database can no longer meet baseline performance metrics. This can happen when too many users are trying to access the database concurrently or the size of the database is causing queries and updates to take too long to execute. The simplest way to scale-out a shared database is through horizontal (row-based) partitioning based on tenant IDs. Online-CADCOM as a SaaS with shared database is good for horizontal partitioning because each tenant has its own set of data, so an individual tenant could easily ask to move his data [4]. Different tenants can place radically different demands to the platform so it is important to be planned carefully to avoid simply creating smaller, but still overtaxed, partitions while other partitions are not used.

If the database has problems related to the size of the database, such as the time it takes to perform queries, a tenant-base horizontal partitioning is used to target database size instead, assigning tenants to database servers in such a way as to roughly equalize the amount of data on each one. The partitioning method has significant impact on the platform development. Building support for monitoring into the platform will help to get an accurate view of tenants' usagepatterns and needs. Also, it's likely that repartition the data will be periodically, as tenants evolve and change the way they use the platform. Horizontal Partitioning pattern is appropriate for use with shared-schema applications, which impose some unusual constraints on the familiar task of scaling a database. It provides a way to scale a shared database while avoiding actions that will break the platform or harm performance (for example - splitting a tenant's data across two or more servers inadvertently or unnecessarily). Splitting method is showed on Figure 1.
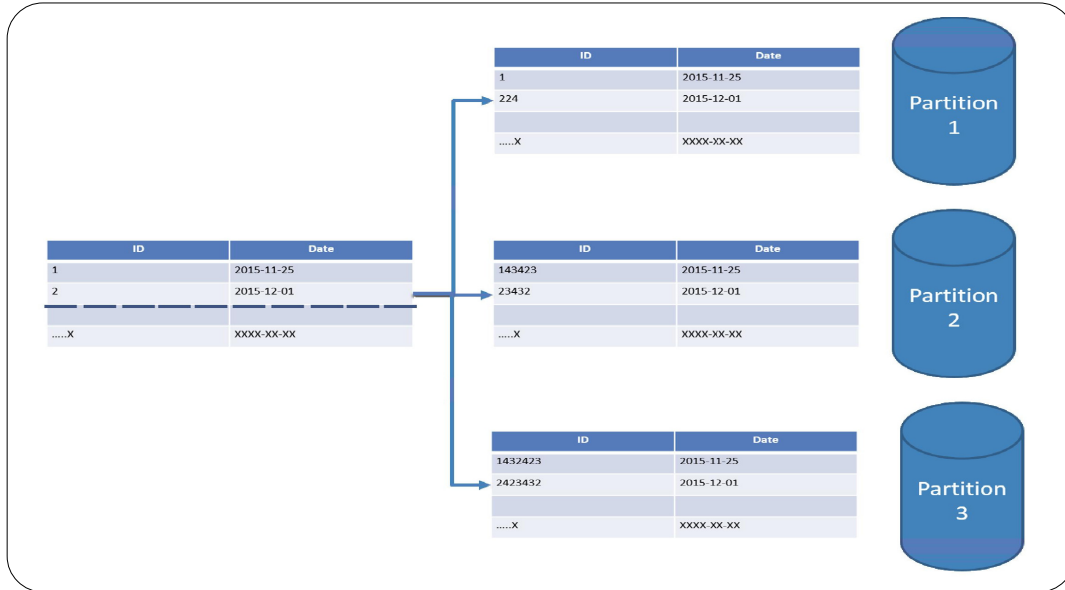
Figure 1. Splitting method of horizontal partitioned database

4. Availability in Online-CADCOM

Hosting shared database with multiple sets of tables requires specific settings that have to make due to IO (InputOutput) and overview availability of the platform [6]. As with physical machines hosting SQL Databases, IO is fast and there is no high level of latency, in Cloud there will be a noticeable IO hit from virtualization (how much of a hit depends from the technology that is used), so some prerequisites must be taken not to harm IO performance of platform and also not to cause latency. Online-CADCOM in Cloud must operate so that extensions to the application, added by a tenant, will not introduce errors that could affect the availability of the role in the platform as describe in [7]. Due to this reason, some specific configurations must be done in Cloud provisioning system as explained in [8] to meet Online-CADCOM platform requirements:

• Fixed size of virtual disks is used for hosting the platform, not dynamic virtual disks. Dynamic virtual disks are less performant than fixed size ones.

• Data in Online-CADCOM is mission-critical so the VM solutions ability to cache writes for extra performance is turned off. This is useful for the development and testing machines where the data is easily replaced, or read-only replicas local to the master that can be recreated if experiences any trouble, but it is recommended that such options stay off for production and backup systems as they increase the chance of dataloss (or worse, corruption) if the machine powers off unexpectedly. So this option is turned off in order to prevent data loss.

• Each VM has to be with enough RAM dedicated to it, so normal working set of each tables of OnlineCADCOM database can fit into memory with a fair amount of spare RAM to avoid unnecessary IO. Also, hypervisor is turned off not to page out the VMs RAM at any time (for example most VMWare products can do this, which allows users to run a larger set of VMs than it could on the same hardware, but the performance degradation can be massive - there is an option to tune or turn off this behavior in most of the Cloud Virtualized Environments).

• If the Cloud uses RAID5 arrays for securing the virtual machines, it would be better to move to RAID10. This will reduce the space available but it will remove the write performance problems that can be apparent with RAID5 or other RAID configurations.

• For all VMs that Online-CADCOM is hosting on, is better that they have own drive or array, so that OnlineCADCOM databases are sharing the drive(s)/array(s) only at each other.

• Online-CADCOM's entire table structure (and everything else the VM is running) fits into at about 6GB allocated to the VM and it sees medium write activity so IO performance is not an issue.

• For using virtual disks with Online-CADCOM, the best controller type must be used for better performance. Some virtualization solutions offer several virtual controller types for better guest compatibility but they are less efficient than others due to protocol limitations in the design they are emulating and limitations in the guest OS's drivers.

• All Cloud providers offer specific drivers for the virtual drive controller and Online-CADCOM's virtual platform must use them otherwise it can speed up bulk IO by reducing the number of memory-to-memory copies and host<->hypervisor<->guest context switches involved in some specific operations.

```xml
XML
<ServiceConfiguration serviceName="Online-CADCOM.Cloud" ... >
<Role name="Online-CADCOM.Web">
<Instances count="1" />
<ConfigurationSettings>
...
<Setting
        name="Cloud.Plugins
        .Caching.NamedCaches"
        value="{&quot;caches&quot;:[{&quot;name&quot;
        :&quot;default&quot;,&quot;policy&quot;
        :{&quot;eviction&quot;
        :{&quot;type&quot;:0},&quot;expiration&quot;
        :{&quot;defaultTTL&quot;
        :10,&quot;isExpirable&quot;
        :true,&quot;type&quot;:1},&quot;
        serverNotification&quot;
        :{&quot;isEnabled&quot;
        :false}},&quot;secondaries&quot;:0}]}" />
<Setting
name="Cloud.Plugins
.Caching.DiagnosticLevel"
value="1" />
<Setting name="Cloud.Plugins
.Caching.Loglevel"
value="" />
<Setting name="Cloud.Plugins
.Caching.CacheSizePercentage"
value="30" />
<Setting name="Cloud.Plugins
.Caching.ConfigStoreConnectionString"
value="UseDevelopmentStorage=true" />
</ConfigurationSettings>
...
</Role>
<Role name="Online-CADCOM.Web.Survey.Public">
...
</Role>
<Role name="Online-CADCOM.Workers.Surveys">
...
</Role>
</ServiceConfiguration>
```

Figure 2. Part of configuration file that shows cache options

Caching is another parameter that can cause problems to tenants in the platform [6]. Online-CADCOM in Cloud uses ASP.NET 4 Caching Session State Provider in the tenant web role. It requires caching setting to be configured in the Cloud configuration file of the platform. Also the platform is using a co-located cache that uses a proportion of the web role's memory. The code bellow shows the part of the service configuration file where the cache configuration is stored. The value for NamedCaches is the default set by the SDK and cache settings can be changed while the application is running simply by editing the configuration file in case of need. Part of configuration file is showed on Figure 2.

Online-CADCOM has built in mechanism for using a delay write pattern when a lot of users are signed in Content Management System (CMS) as method explained in [8] [9]. The delayed write pattern is a mechanism that allows code to hand off tasks that may take some time to complete, without need to wait for them to finish - for example upload big video files in Knowledgebase [8] [9]. The tasks can be executed asynchronously as background processes, while the code that initiated them continues to perform other work or returns control to the users, so a lot of big data files can be processed simultaneously as explained in [9]. The delayed write pattern is particularly useful in the platform because the tasks that must be carried out can run as background processes, and application's UI (User Interface) is free for other tasks as quickly as possible. It doesn't mean that the platform cannot return the result of the background process to the user within the current request on the same page without refreshing it. In Clouds background tasks are typically initiated by allowing the UI to hand off the task by sending a message to a storage queue of the platform. Because queues are the natural way to communicate between the roles in a Cloud application, it's tempting to consider using them for an operation such as saving data collected in the UI of Online-CADCOM. The UI code can write the data to a queue and then continues to serve other users in the platform without need to wait for operations on the data to be completed. Figure 3 shows the delayed write pattern that Online-CADCOM can use when the platform is hosted in Cloud.
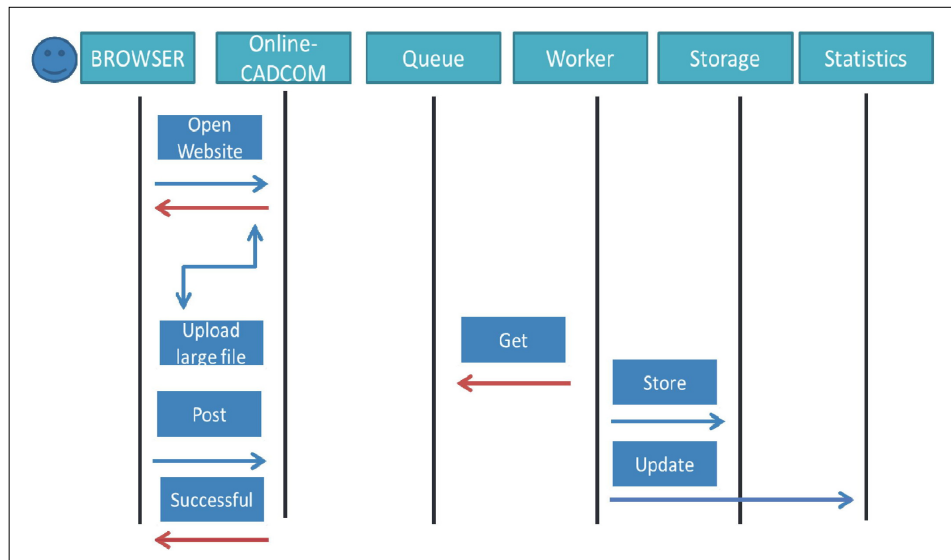

Figure 3. Delay write pattern algorithm

In this scenario a platform user inserts a file in CMS platform and then submits this file to the platform. The code running in the web role instance puts the file into a message on a queue and returns a "Successful" message to the user as quickly as possible, minimizing the value of time response. One or more tasks in the worker role instances are then responsible for reading the user message that contains the file from the queue and saves it to the cloud storage and then updates the summary statistics. This operation is implemented to avoid any possibility of conflict during uploading files and updating content information and also against latency.

Online-CADCOM is writing directly to storage from the web role process [4][10]. The worker role instances had only one task to do: updating the summary statistics. As using the delayed write pattern implemented in the platform, worker roles must accomplish two tasks: saving the files from user iterations to storage and updating the summary statistics to show information about uploaded files. For best availability it is better that these tasks are separated. New uploaded resources

from users is good to be saved to storage as quickly as possible and calculating the summary statistics is not an urgent requirement that is time dependable. The summary statistics can be recalculated at a later time, but file uploading at a later time is not possible. Online-CADCOM can differentiate the level for special and standard users by ensuring that summaries for special users are available more quickly. To scale the tasks separately, the platform uses two separate worker roles as explained in [8]:

• A worker role that just updates the statistics by polling a queue for messages containing the identifier of new action sets.

A worker role that just saves new uploaded resources to storage by polling a queue for messages that contain the resource information. The files uploaded in the platform contain headers with information about user, type of file, proper location and etc.

5. Online-cadcom as a Secure Multitenant Cloud Application

Online-CADCOM as a SaaS architect cloud based platform has multiple defense levels that complement one another to provide data protection in different ways, under different circumstances available to defense application against both internal and external hacker treats as explained in [4]. This means that Online-CADCOM has security on different levels as showed in [6]. The platform is sensitive because of its data content, so three different levels of security are implemented:

• **Filtering** – Platform uses a level between a tenant data source that acts like a sieve, making this level appear to the tenant as its data is the only data in the database.

• **Permissions** – Platform uses access control lists (ACLs) to determine who can access data in the application and what this user can do with these resources.

• **Encryption** – Every tenant data in the platform is encrypted so in case of threats it remains inaccessible to unauthorized parties even if they have a copy of it.

Online-CADCOM use a trusted database connection that in case of multi-tenant Cloud application has two methods for security – impersonation and trusted subsystem account [11]. With the impersonation access method, the database is set up to allow individual users to access different tables, views, queries, stored procedures, and other database objects. When an end-user performs an action that directly or indirectly requires a call to a database, the application presents itself to the database as that user, gives access to the user for the purposes of accessing the database. With the trusted subsystem access method, the platform always connects to the database using its own platform process identity, independent of the identity key of the user. The server then grants the application access to the database objects that the application can read or manipulate. This approach makes security management easier, eliminating the need to configure access to database objects on a per-user basis, but it means giving up the ability to secure database objects for individual users and tenants. In OnlineCADCOM as SaaS platform, the concept of "users" is a bit more complicated than in traditional hosted web applications as showed in [11], because of the distinction between a tenant and an end user. The tenant is an organization that uses the application to access its own data store, which is logically isolated from data stores belonging to any other tenants. Each tenant grants access to the application to one or more end users, allowing them to access part of the tenant's data using end user accounts controlled by the tenant. So in this case, the platform uses a hybrid approach to data access that combines aspects of both the impersonation and trusted subsystem access methods and gives maximum logical isolation of tenant data without creating an unworkably complex security model. This approach involves creating a database access account for each tenant, and using Access Control Lists (ACLs) to grant each of these tenant accounts access to the database objects the tenant is allowed to use. When an end user performs an action that directly or indirectly requires a call to the platform database, the application uses credentials associated with the tenant account, rather than credentials associated with the end user. The database server does not know differences between requests originating from different end users associated with the same tenant, and grants all such requests access to the tenant's data. Within the application itself, security code prevents end users from receiving and modifying any data that they are not entitled to access. To secure the database on the table level for example for Knowledgebase, the following automatic command is generated for this table [8]:

    GRANT SELECT, UPDATE, INSERT, DELETE ON Knowledgebase FOR User

This adds the user account for "User" to the ACL for the table and this only has to be done once, during the tenant provisioning process because of the hybrid approach to the database access, discussed earlier, that is used in the platform. Any end user accounts created by the tenant will be able to access the table. Online-CADCOM is using also a tenant view filter using SQL views to grant individual tenants access to some of the rows in a given table, while preventing them from accessing other rows:

CREATE VIEW User AS SELECT * FROM Knowledgebase WHERE TenantID = SUSER_SID()

In addition, Online-CADCOM uses a SAS (Shared Access Signatures) method for the users that can't be trusted with the storage account key showed in [4] and explained in details in [11]. This is an additional option when exposing the storage account key is not a good option. This approach provides a safe alternative that allows other clients to read, write, and delete data in platform storage account according to the permissions granted, and without need for the account key. This approach is useful when a client tries to access the platform from front-end proxy service. Also this method is created in a platform in order to enable automatic data collaboration with another platform for data sharing or extraction. Figure 4 shows SAS method implemented in Online-CADCOM as a SaaS Cloud platform.
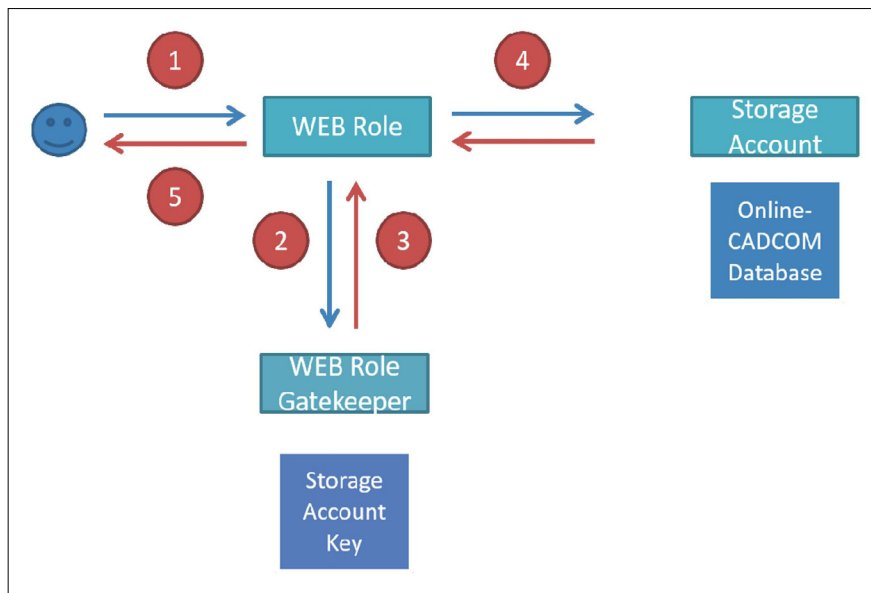

Figure 4. Shared access signature method (SAS)

First the client browser sends a request to access Online-CADCOM database. Second, the web role sends a request to the gatekeeper worker role for a SAS URL that will enable read only access to platform database. Third, the gatekeeper worker role uses the storage account key to generate the SAS URL and returns it to the worker role. Four, the web role uses 282 the SAS URL when it queries for the platform database it needs to render the web page. Then five, the web role returns the page to the browser.

6. Online-CADCOM Application Lifestyle in Cloud

The most significant challenges to administrative and management tasks for Online-CADCOM as multi-tenant application when compared to a standard hosted application are concerned with the processes used to back up and restore data [12]. Each tenant's data is isolated from all others through the use of a shared database with separate tables and storage account for every tenant. It is vital that this isolation is not compromised during the backup and restore process. A tenant must not be able to access a backup that contains other tenants' data. That is ensured by the database architecture [4]. An automatic backup script is developed in case of database failure or deletion. Also backup of the entire platform can be set from Cloud service platform options as folder backup, partition backup or another user option backup. The code below shows script that is making an SQL backup of the database using PHP language:

```php
<?php
Include 'config.php';
Include 'opendb.php';
$tableName = 'online_vadcaom';
$backfile=
'backup/online cadcom)datenow)).sql';
$query ="SELECT * INTO OUTFILE
'$backupFile' FROM $tableName";
Include 'closedb.php';
?>
```

In case of script failure, or losing a connection with the database instance in the cloud, Online-CADCOM has separate method developed script for backing up using the Unix operation system environment with the following command which is scheduled for everyday work:

root@cloud_platform root mysqldump -u root - pPASSWORD —all-databases | gzip > /mnt/disk2

6. Conclusion

Online-CADCOM as a SaaS Cloud based application has to meet high requirements for quality developed process, clean source code and a couple of functions that make the platform compatible with Cloud based technologies. The platform has to be developed using the latest technologies in MVC Framework approach containing rules about proper scale and multi-tenant availability. The solutions proposed in this paper can be used for transfer of other similar platforms from virtual server to cloud. It's especially useful for CAD tools and platforms which will be transferred to cloud.

References

[1] Marinova, G., Chikov, O. (2015). Methodology for Tools Integration in the Online Assisted Platform for Computer-Aided Design in Communications, ICEST 2015, 24-26 June 2015, Sofia, p. 31-36.

[2] Chikov, O., Marinova, G. (2016). MVC Framework Approach in the Online assisted platform form Computer-aided Design in Communications – Online-CADCOM, *Computer and Communications Engineering*, TUS.

[3] Chikov, (2016). Software Development of Content Management System in Online-CADCOM Platform, ICEST 2016, Ohrid, Macedonia.

[4] Kavis, M. J. (2014). Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS), Wiley, 1 edition.

[5] Erl, T. (2013). Cloud Computing: Concepts, Technology & Architecture, Prentice Hall.

[6] Bond, J. (2015). The Enterprise Cloud: Best Practices for Transforming Legacy IT, O'Reilly Media, 1 edition.

[7] Havewala, P. H. (2012). Oracle Enterprise Manager Cloud Control 12c: Managing Data Centers Chaos, Packt Publishing.

[8] Schaefer, (2014). Cloud-Based Design and Manufacturing (CBDM), Springer.

[9] Ruparelia, N. B. (2013). Cloud Computing, The MIT Press.

[10] Rhoton, J. (2009). Cloud Computing Explained: Implementation Handbook for Enterprises, Recursive Press, 2nd edition.

[11] Rittinghouse, J. W., Ransome, J. F. (2010). Cloud Computing Implementation Management and Security, CRC Press.

[12] Trends in Cloud Computing, CompTIA Research, Fourth Annual – August 2013.

[13] Mentor Graphics, SystemVision Cloud - Free Cloud-Based Simulation Tool, https://www.systemvision.com/

[14] DesignSoft, TINACloud, 2017, https://www.tina.com/English/tinacloud/

[15] Cadence OrCAD Capture Cloud, http://www.orcad.com/orcadcapture-cloud