

Workspace SimpleDB persistent storage

From version 1.10 we support [Workspace Data Container](#) implementation for [Amazon SimpleDB service](#) . The feature gives an ability to use SimpleDB as Workspace storage (Items storage) and positioned as Content Repository for grid computing.

Amazon SimpleDB is a web service for running queries on structured data in real time. This service works in close conjunction with Amazon Simple Storage Service (Amazon S3) and Amazon Elastic Compute Cloud (Amazon EC2), collectively providing the ability to store, process and query data sets in the cloud. Amazon SimpleDB is easy to use and provides the core functionality of a database - real-time lookup and simple querying of structured data - without the operational complexity. Amazon SimpleDB requires no schema, automatically indexes your data and provides a simple API for storage and access.

Warning: Current Workspace Data Container SimpleDB implementation (v.1.10) is in beta state.

SimpleDB based Workspace Data Container *org.exoplatform.services.jcr.aws.storage.sdb.SDBWorkspaceDataContainer* is transparent for JCR API and no additional client-level programming is required. Only configuration should be applied to a Workspace.

```
<repository-service
default-repository=
"repository">
  <repositories>
    <repository name=
"repository" system-workspace=
"system"
default-workspace=
"production">
  ....
  <workspaces>
    <workspace name=
"production">
      <container class=
"org.exoplatform.services.jcr.aws.storage.sdb.SDBWorkspaceDataContainer">
        <properties>
          <property name=
"max-buffer-size" value=
"200k" />
          <property name=
"swap-directory" value=
"target/temp/swap/ws" />
          <property name=
"aws-access-key" value=
"put-your-key-here" />
          <property name=
"aws-secret-access-key" value=
"put-your-secret-key-here" />
          <property name=
"domain-name" value=
"jcr-test-ws" />
```

JCR - Workspace SimpleDB storage

```
        </properties>
        ....
    </container>
    ....
</workspace>
.....
```

where

max-buffer-size - a threshold in bytes, if a value size is greater then it will be spooled to a temporary file.

swap-directory - a location where the value will be spooled if no value storage is configured but a max-buffer-size is exceeded.

aws-access-key - Amazon Web Services access key.

aws-secret-access-key - Amazon Web Services secret access key.

domain-name - Amazon SimpleDB domain name.

But SimpleDB has few limitations which make it a good for metadata storage but not applicable for a mid and large content storage. [SimpleDB limits](#) don't allow you store more of 1024 bytes in single Attribute Value.

Within JCR storage implementation 1020bytes is a maximum threshold of Value size stored in SimpleDB. But have in mind that binary data (*PropertyType.BINARY*) will be encoded into Base64 string before processing. So, binary data size can be stored in SimpleDB is smaller.

In this way it's required to use [S3 External Value Storage](#) in conjunction with SimpleDB Items storage.

```
.....
    <value-storages>
        <value-storage id=
"production" class=
"org.exoplatform.services.jcr.aws.storage.value.s3.SimpleS3ValueStorage">
            <properties>
                <property name=
"bucket" value=
"production-1"/>
                <property name=
"aws-access-key" value=
"put-your-key-here"/>
                <property name=
"aws-secret-access-key" value=
"put-your-secret-key-here"/>
                <property name=
"s3-swap-directory" value=
"/temp/s3swap/production"/>
                </properties>
                <filters>
                    <filter property-type=
"Binary" min-value-size=
"850"/>
                    <filter property-type=
```

JCR - Workspace SimpleDB storage

```
"  
String"  
  min-value-size=  
"1020" />  
      </filters>  
    </value-storage>  
    .....
```

Here are a filters specially configured to fit suggestions about SimpleDB limits on Values size.

min-value-size for Binary *Property* type is 850 bytes, i.e. all data large of this value will be stored in S3 service. Smaller direct in SimpleDB row. *min-value-size* for String *Property* type is 1024 bytes, i.e. all Strings large of 1024bytes will be stored in S3. All other Values will be stored in SimpleDB.

Warning: There is no Item update rollback option currently. Warning will be logged but no data will be rolled back actually. Add and Remove rollback is supported. Additional retries will not be applied on modification operations timeout also. It's charge of future storage changes.

Another thing should be noted about SimpleDB (and S3) is [Eventual Consistency of the Amazon services](#) . Current JCR storage implementation is based on [Amazon SimpleDB java client API](#) which internally supports [SimpleDB API Error Retries](#) and HTTP connection reestablishment in case of timeout.

JCR Storage also has special logic for *Eventual Consistency* wait in case of read Properties lists. Contains additional logic to prevent timeout errors are not covered by SimpleDB java client.