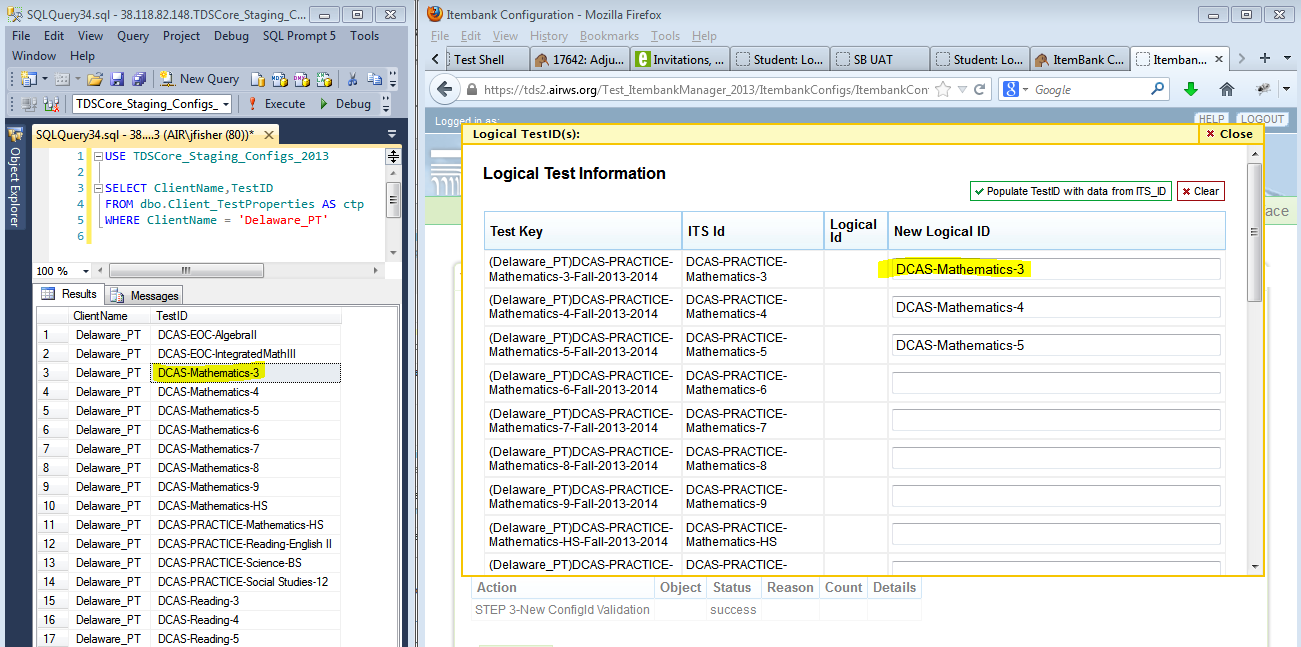
# A Primer on TDS Environments

1. **Dev –** only for the developers. Do not touch unless explicitly asked.
   1. **ItemBank Manager**
      1. <https://tds2.airws.org/Test_ItembankManager_2013/NoRTSLogin.aspx>
   2. **TDS Configs Manager**
      1. None. Must copy configs from a different DB or use SMSS
2. **Staging –** only for the project managers. The purpose of this environment is to work on new ItemBank and TDS configurations in a pristine environment without fear of a non-PM corrupting the work. There is no student or proctor application in this environment purposefully because it is not meant for testing.
   1. **ItemBank Manager**
      1. <https://tds2.airws.org/Test_ItembankManager_2013/NoRTSLogin.aspx>
   2. **TDS Configs Manager**
      1. <https://tds2.airws.org/Staging_TDSConfigsManager_2013/NoRTSLogin.aspx>
3. **Test –** for functional testing by the QA Team. All ItemBank and TDS Configs here are fair game for change at any time and they may or may not be true to production. Nearly everyone on the team has the ability to change configurations in this environment with little to no notice in order to fit their needs. For example; if a client typically has Print on Request set as a non-selectable tool the QA Team may change this to selectable so that they don’t need to find students with this accommodation in order to test.
   1. **ItemBank Manager**
      1. <https://tds2.airws.org/Test_ItembankManager_2013/NoRTSLogin.aspx>
   2. **TDS Configs Manager**
      1. <https://tds2.airws.org/Test_TDSConfigsManager_2013/NoRTSLogin.aspx>
4. **Deploy** – all configurations in the deployment environment are fair game to go to production at any time. This should be the final testing environment for the QA Team to certify the realistic client configuration.
   1. **ItemBank Manager**
      1. <https://tds2.airws.org/Test_ItembankManager_2013/NoRTSLogin.aspx>
   2. **TDS Configs Manager**
      1. None. Must copy from Staging or Deploy using SQL Data Compare
5. **Simulation** – for the tech team to run adaptive algorithm simulations, tweak the blueprint, and push changes to ITS. Typically a deployment engineer will load configurations into this environment. However, after the first config of the year is loaded the tech team has the ability to load their own subsequent configs. There is no ItemBank or TDS Configs Manager for this environment
6. **Platform Review** – allows temps to navigate a series of fixed-form tests so that they may review the items in the item pool. This has largely been done by deployment engineers or the PM in the past, but we should build an application to do this.
   1. **ItemBank Manager**
      1. <https://tds2.airws.org/Test_ItembankManager_2013/NoRTSLogin.aspx>
   2. **TDS Configs Manager**
      1. None. Must use SMSS
7. **UAT** –allows clients to test the functionality of the system and obtain client sign off before deploying to production.
   1. **ItemBank Manager**
      1. TBD
   2. **TDS Configs Manager**
      1. None. Must copy from Staging or Deploy using SQL Data Compare
8. **Load Test** – for load testing TDS. Custom environment. No TDS configs manager or itembank manager.
9. **Production** – hosts all LIVE tests and is public facing
   1. **ItemBank Manager**
      1. TBD
   2. **TDS Configs Manager**
      1. Purposefully none. All configs should be done in test first, tested, then a PCB should be written to copy them to production.

# How to Load New ItemBank and TDS Configurations

1. When a new ItemBank config is ready a configID and a config validation spreadsheet should be delivered by the Tech Team, Content Team or the Project Director. There are going to be many many ItemBank configs over the course of the year and many of them will actively be worked on simultaneously so we need to track it somewhere. A FogBugz wiki page has been prepared to track the ItemBank configs throughout the year.
   1. <https://bugz.airast.org/default.asp?W577>
2. Look at the config validation log that was supplied with the configID. If there are any FATAL errors in this log send it back. Anything else review closely and make sure that there are no serious issues.
3. At this stage it is also a good idea to pull the content and validate it. Even though the content is not needed until much later in the process there are often problems with the content for a new config and it is ideal to have the Content Team start working on fixing those issues right away rather than hold up the process later.
4. Pull down the config to WS using SQL Red Gate Data Compare
   1. .41.ITS\_TDS\_Configuration to .148.ITS\_TDS\_Configuration
   2. Change the where clause to “where configID = ‘XXXX’”
5. Load everything into **STAGING** first so that it can be prepared for testing
   1. Before loading anything in the ItemBank it is advisable to open SQL Data Compare and compare TDSCore\_Staging\_Configs\_2013 with TDSCore\_Test\_Configs\_2013. Sync as much data from staging to test as possible and then write down the number of records that are still different between the two databases. This is important because when TDSCore\_Staging\_Configs\_2013 is updated with new TDS Configs as a result of this ItemBank load it will be very easy to us this application to take a diff and see exactly what the ItemBank changed when it inserted / updated records into TDS Configs.
   2. Use step 3 in the ItemBank Manager app (urls above) to load the new config ID
      1. Enter the ConfigID
      2. Enter the content path
         1. **D:\DataFiles\BB\_Files\tds2\_airws\_org\TDSCore\_2013-2014\**
         2. This path must be entered exactly like this. Do not add or remove anything or the student app will throw errors saying it can’t find the content.
      3. Check the box that says “Update TDS Configs”
      4. Click the [GO] button
   3. The application will require confirmation (c/o a checkbox) that the config validation was read and this config is ready to be loaded. It will also show the user the tests that are coming in the new config and the tests that are already in the ItemBank just as an aid to make sure the right thing is being loaded into the right place.
   4. After confirming this application is going to load a set of staging tables with all the new data that is to be entered into the ItemBank. As it is doing that it will run additional validation checks. If a fatal error is found it will not allow the user to continue.
   5. If everything validates successfully the user will be asked to configure the logicalIDs for all the new physical tests coming in this config. If this is not the first configuration for the year then these logicalIDs will likely be populated already. The existing testIDs in TDS Configs Test Properties for this client should be used in this mapping because everything in TDS Configs from the previous year is already configured against the LogicalID. If this is not done it’ll become necessary to reconfigure TDS Configs all over again and there will be redundant data.
      1. Example mapping showing an online and paper test that share a logicalID so that max opportunities can be managed between them:

|  |  |  |
| --- | --- | --- |
| **Logical ID** (needs to be supplied by the PM loading the ItemBank if the physical test key is new) | **ITS ID** (supplied by Content and Tech) | **Physical Test Key** (ItemBank loader generated) |
| DCAS-Reading-4 | DCAS-Reading-4 | (Delaware)DCAS-Reading-4-Fall-2012-2013 |
| DCAS-Reading-4 | DCAS-PAPER-Reading-4 | (Delaware)DCAS-PAPER-Reading-4-Fall-2012-2013 |



* + 1. **Note:** there are a handful of minor changes the Content or Tech Teams can do which will make the system think that these are new tests when they’re not. These changes are anything that modifies the Physical test key from what we already have in the ItemBank and TDS Configs TestMode table.
       1. Example Physical Test Key: (Delaware)DCAS-Reading-4-Fall-2012-2013
       2. If the clientname, ITSID, Season or School Year changes in the config the ItemBank will think this is a new test. Sometimes that is desirable and sometimes it is not.
  1. After clicking on [SUBMIT] there could possibly be some more errors if SQL Server died when trying to insert some new data. Take these errors very seriously because it could possibly mean there is some corrupted data in the itembank now OR that TDS Configs was not updated like it was supposed to be.
     1. An error like the following means that UpdateTDSConfigs did not run properly:
        1. UpdateTDSConfigs: Violation of PRIMARY KEY constraint 'PK\_Testform'. Cannot insert duplicate key in object 'dbo.Client\_TestformProperties'.
        2. This could happen if there is a form being reused from one year to the next. An easy way to deal with this is the following:

|  |
| --- |
| /\* Remove old forms from TDS Configs where they're going to be replace in the incoming ItemBank config \*/  DELETE FROM TDSCore\_Staging\_Configs\_2013.dbo.Client\_TestformProperties  WHERE [\_efk\_TestForm] IN (SELECT \_key FROM TDSCore\_Staging\_Itembank\_2013.dbo.Loader\_Forms AS lf) |

* 1. At this point if everything was successful then the Staging ItemBank should have the latest config and TDS Configs should have the new stuff seeded into it.
  2. Re-run SQL Data Compare to compare TDSCore\_Staging\_Configs\_2013 to TDSCore\_Test\_Configs\_2013. Analyze the results and confirm that everything here looks right. For example; if there is a change to client\_testgrades that would be very abnormal look at the difference between Staging and Test and see if there is anything serious that needs to be corrected. If there is it’s likely an issue with the ItemBank config that was just loaded.
  3. The vast majority of the time for new configs there will be new entries into client\_testmode that have a sessiontype = 99. This means these are new tests that are not yet active. A sessiontype needs to be 0 or 1 in this table to be a legit test. Set all the OLD tests to 98 if they’re not going to be used anymore and set all the 99 tests to 0 or 1 as appropriate. Note; there is no interface for this.
  4. Go through all other TDS Configs tables and configure everything for these new tests per the client specs. A checklist of all tables is below for convenience. Some of these will be done by the ItemBank automatically.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Client | Client\_ItemScoringConfig | Client\_SystemFlags | Client\_TestGrades | Client\_TestToolType |
| Client\_AccommodationFamily | Client\_Language | Client\_TDS\_RTSAttribute | Client\_Testkey | Client\_TestWindow |
| Client\_Accommodations | Client\_MessageTranslation | Client\_TDS\_RTSAttributeValues | Client\_TestMode | Client\_TimeLimits |
| Client\_AllowIPs | Client\_MessageTranslationAudit | Client\_Test\_ItemConstraint | Client\_TestPrerequisite | Client\_TimeWindow |
| Client\_Externs | Client\_PilotSchools | Client\_Test\_Itemtypes | Client\_TestProperties | Client\_ToolDependencies |
| Client\_FieldtestPriority | Client\_RTSRoles | Client\_TesteeAttribute | Client\_TestRTSSpecs | Client\_ToolUsage |
| Client\_ForbiddenAppsExcludeSchools | Client\_SegmentProperties | Client\_TesteeRelationshipAttribute | Client\_TestscoreFeatures | Client\_VoicePack |
| Client\_ForbiddenAppsList | Client\_Server | Client\_TestEligibility | Client\_TestTool |  |
| Client\_Grade | Client\_Subject | Client\_TestformProperties | Client\_TestToolRule |  |

1. Do another compare with tdscore\_test\_configs\_2013 and confirm that all the changes are expected changes based on the configurations just made.
2. Start loading the **TEST** environment with this new data
   1. TDS Configs (more specifically the client\_testmode table) must be loaded into test before the ItemBank config can be loaded there because that is where the itembank loader receives the logicalIDs from. Now that this work has been done in staging it can simply be copied to test.
   2. Using SQL Data Compare copy over all the appropriate configs for the specific client to tdscore\_test\_configs\_2013.
   3. If this is the first config for the client for the year then it is necessary to clean out tdscore\_test\_itembank\_2013. To do this use the scripts in the box below. If this is NOT the first config for the year then proceed to the next step

|  |
| --- |
| --DELETE FROM dbo.tblClient WHERE Name = 'XXXXX' -- THIS MUST BE DONE FIRST!!!  --/\* The rest of these will clean house on the ItemBank based on what the trigger (from above) has already cleared out \*/  --DELETE FROM dbo.tblItem WHERE [\_efk\_ItemBank] NOT IN (SELECT [\_efk\_ItemBank] FROM dbo.tblItemBank AS tib)  --DELETE FROM dbo.tblStimulus WHERE [\_efk\_ItemBank] NOT IN (SELECT [\_efk\_ItemBank] FROM dbo.tblItemBank AS tib)  --DELETE FROM dbo.tblSetofItemStimuli WHERE [\_fk\_AdminSubject] NOT IN (SELECT \_key FROM dbo.tblSetofAdminSubjects AS tsas)  --DELETE FROM dbo.tblSetofItemStrands WHERE [\_fk\_AdminSubject] NOT IN (SELECT \_key FROM dbo.tblSetofAdminSubjects AS tsas)  --DELETE FROM dbo.tblItemProps WHERE [\_fk\_AdminSubject] NOT IN (SELECT \_key FROM dbo.tblSetofAdminSubjects AS tsa) |

* 1. Using the same ItemBank loader application choose TDSCore\_Test\_ItemBank\_2013 this time.
  2. Load the exact same config that was just loaded into TDSCore\_Staging\_Itembank\_2013.
     1. DO NOT select “UpdateTDSConfigs” or this will overwrite the configs that you just copied there from staging.
     2. If the TDS configs data from staging’s client\_testmode was copied over properly this time the logicalIDs will be prepopulated in the ItemBank Manager. If they’re not then something is wrong.
  3. Make sure that TDSCore\_Test\_Session\_2013.\_externs has an entry for the new client that was loaded or it won’t be available to the applications
  4. Get the content for the new config from the appropriate ItemBank in ITS. If this is the first config of the year for these items then it is best to retrieve the content from the publication module. If this is a new config that modifies existing items (that have already been in production) it is best only to extract those items which are NEW.
  5. Check the content validation log and make sure there are no issues that need to be resolved before moving forward. When the content is retrieved and QC’d place it in the directory below so it is available to everyone:
     1. **H:\Assessment\Interactive Content\Projects\Test Delivery System\ITS Content\**
  6. Deploy the content using the contentuploader application
     1. <https://tds2.airws.org/contentuploader_2013>
  7. At this point the TEST system should be ready to use. It is prudent to test it before releasing to QA.
     1. https://tds2.airws.org/test\_student\_2013/?c=XXXXXX
     2. https://tds2.airws.org/test\_testadmin\_2013/login.aspx?c=XXXXXXXX&clslogin=false
  8. Update FogBugz wiki along the way to keep track of the status of the work in case it is necessary to pause and come back.