**BlackPearl Spectra S3 Client Certification Test Plan**

**Version 0.1**

**11/9/2015**

1. **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** |
| 0.1 | 11/9/15 | GSJ | Initial format and content. Designed for BlackPearl Software V1.2 |

1. **Introduction**

The document describes the certification tests for a BlackPearl Client presenting a Spectra S3 compatible interface. Complete details of the Spectra S3 Certification Program can be found in the **Spectra S3 Certification Program** link in the Reference Information section below.

1. **Reference Information**

**Developer Website:** For information on the Spectra S3 interface and developing a Spectra S3 compatible interface, please refer to the Developer Website on SpectraLogic.com at: <https://developer.spectralogic.com/ds3/>

**BlackPearl:** For further detailed information refer about BlackPearl, please refer to the following documents:

**BlackPearl User Guide**: <http://support.spectralogic.com/python/documents/Spectra%20BlackPearl%20User%20Guide.pdf>

**BlackPearl Site Preparation Guide**:

<https://support.spectralogic.com/documentation/user-guides/blackpearl-site-prep-guide>

**BlackPearl Network Setup Tips**:

 <https://support.spectralogic.com/documentation/user-guides/blackpearl-network-setup-tips>

**BlackPearl DS3 API Reference**:

<https://support.spectralogic.com/documentation/user-guides/blackpearl-ds3-api-reference>

**BlackPearl Drive Installation Guide**:

 <https://support.spectralogic.com/documentation/service-and-maintenance-guides/blackpearl-install-drive.pdf>

**BlackPearl Rack Mounting Guide**: <http://support.spectralogic.com/python/documents/Spectra%20BlackPearl%20rackmount%20guide.pdf>

**Spectra S3 Client Certification Program:** For information about the BlackPearl Spectra S3 Client Certification Program refer to the Certification Program Website: <https://developer.spectralogic.com/certification/>

**Important Notice:** The remainder of this document assumes that the user is familiar with details of the Spectra S3 interface, the BlackPearl product, information contained on the Spectra Logic Developer Site, information contained on the Spectra Logic Certification Site and others as listed in the Reference Information section of this document.

1. **Test Setup, Installation of the Host Client Software, Connection to BlackPearl and Configuration**

After you have complete development and testing of your Spectra S3 compatible Interface you may decide to pursue certification with Spectra Logic – Please refer to the **Spectra S3 Client Certification Program** link shown above. If you choose to proceed, please have your answers to the Client Questionnaire (<https://developer.spectralogic.com/client-questionnaire/>) completed and then contact Spectra S3 Certification Program personnel (<https://developer.spectralogic.com/contact/> ). Spectra Certification Personnel will work with you to determine the appropriate test setup, test equipment requirements, test time, installation of host software and client, and other testing program details. Review the installation of your product and BlackPearl connection and configuration to verify that everything is working properly prior to commencing the testing.

Do we need to include some additional information about setup, configure, etc? Create a User using BP User Interface? Create a Bucket?

1. **Logging and Result Capture Requirements for Certification**
* **Log files:** Please pull a log set daily from Black Pearl. These log files are used to assist with debugging any failures that may occur. Instructions for downloading the BlackPearl logs can be found in the User Guide: Refer to the “AutoSupport” chapter and the section titled “Log Sets”.
* **Screen Captures:** Screen captures will be required for several of the tests presented in this document to enable Spectra Logic to verify test results and certify a product. Please ensure that:
	1. The person performing the test understands how to capture a screen snip of (or capture a screen shot and snip or crop) the information that is requested. In Windows, this can be accomplished by using the Windows Snipping Tool (<http://windows.microsoft.com/en-us/windows/use-snipping-tool-capture-screen-shots#1TC=windows-7> )
	2. The host system computer connected to the BlackPearl User Interface has sufficient disk storage space to capture the screen snips.
	3. The screen snips can be copied into the test results summary document that is required to be submitted to Spectra Logic to complete the Certification Process
1. **Sequential List of Tests That Must Be Performed Using the Host Application and Results That Must be Captured**
2. Attempt to access BlackPearl with invalid user credentials.
	1. Capture a screen snip that clearly shows the system status once the system reports an error, most likely when credentials are entered or when the application tries to PUT a file using the invalid credentials. Paste the screen snip into a new document (Microsoft Word is preferred) titled “Test Results” with a heading of “Invalid user credentials”.
3. PUT a single object with size less than the current chunk size. For BlackPearl V1.2, the default chunk size should be set to 100GB.
	1. Capture a screen snip (or snips) that clearly shows the status of the job: a) In Progress and b) Completed with status of completed or error. Paste the screen snips into the “Test Results” document with a heading of “PUT a single object with size less than the current chunk size”.
4. GET an object that already exists from BlackPearl (to a fresh location in application so there is no conflict with original source – can use the same object as the previous test)
	1. Capture a screen snip (or snips) that clearly shows the status of the job: a) In Progress and b) Completed with status of completed or error. Paste the screen snips into the “Test Results” document with a heading of “GET an object that already exists from BlackPearl”.
5. Attempt to access a bucket that the user does not have access to (PUTs and GETs)
	1. Using the BlackPearl User Interface:
		1. Create a new user with credentials that are different than the credentials the application is currently using.
		2. Create a bucket with the user credentials created in 4.a.i above.
	2. From the host application, attempt to PUT data into the new bucket using the applications current user credentials (not the ones created in 4.a.i above).
		1. Capture a screen snip that clearly shows the application error. Paste the screen snip into the “Test Results” document with a heading of “Attempt to PUT to a bucket that the user does not have access to”.
	3. From the host application, try to GET data from that bucket, if application can see or thinks there is data in the bucket
		1. Should report failure status – Capture a screen snip that clearly shows the application error. Paste the screen snip into the “Test Results” document with a heading of “Attempt to GET from a bucket that the user does not have access to”.
6. This item needs editing/completion from Steven/Jeff: Updating an existing object in BlackPearl (if allowed) - Either test atomic replacement/failure when PUT with same object name

(is atomic replacement settable with Create Bucket – DS3 API, or in the BP GUI, etc.) Ability to update an existing object (overwrite) is a setting on the bucket.

* 1. PUT a file, then
	2. If testing atomic replacement,
		1. Modify file in application and then upload a new version of the file to BlackPearl using the same object name.
		2. PUT of same file again with the same file name/GUID, does it succeed
		3. Then, test a GET of the new/modified file just sent, is it the modified file
	3. If not testing atomic replacement, did the PUT fail?
	4. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “Updating an existing object in BlackPearl”.
1. GET an object from BlackPearl that doesn’t exist
	1. Open different Spectra client (DSB or CLI)
	2. Capture a screen snip of DSB or CLI that clearly shows the list of files. Paste the screen snip into the “Test Results” document with a heading of “List of objects from BlackPearl that exist”.
	3. DELETE an object(s) using the DSB or CLI
	4. Attempt to restore deleted object(s) through application
	5. Capture a screen snip that clearly shows the application status of “object does not exist” and gracefully return this error to end user. Paste the screen snip into the “Test Results” document with a heading of “GET an object from BlackPearl that doesn’t exist”.
2. PUT single object larger than current chunk size. For BlackPearl V1.2, the default chunk size should be set to 100GB.
	1. Capture a screen snip (or snips) that clearly shows the status of the job: a) In Progress and b) Completed with status of Success or Error. Paste the screen snips into the “Test Results” document with a heading of “PUT single object larger than current chunk size”.
3. GET an object larger than the current chunk size that already exists from BlackPearl (delete or restore to a fresh location so no conflict with original source – can use the same object as the previous test)
	1. Capture a screen snip (or snips) that clearly shows the status of the job: a) In Progress and b) Completed with status of Success or Error. Paste the screen snips into the “Test Results” document with a heading of “GET an object larger than the current chunk size”
4. PUT 50 objects of 1GB size to BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “PUT 50 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “PUT 50 objects of 1GB size Complete or Error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as BPPerformPUT50.CSV.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “PUT 50 objects of 1GB size Performance”.
5. GET 50 objects of 1GB size from BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “GET 50 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET 50 objects of 1GB size Complete or Error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as BPPerformGET50.CSV.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “GET 50 objects of 1GB size Performance”.
6. PUT 500 objects of 1GB size to BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “PUT 500 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “PUT 500 objects of 1GB size complete or error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as BPPerformPUT500.CSV.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “PUT 500 objects of 1GB size Performance”.
7. GET 500 objects of 1GB size from BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “GET 500 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET 500 objects of 1GB size Complete or Error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as BPPerformGET500.CSV.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “GET 500 objects of 1GB size Performance”.
8. PUT 1000 objects of 1GB size to BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “PUT 1000 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “PUT 1000 objects of 1GB size complete or error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as BPPerformPUT1000.CSV.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “PUT 1000 objects of 1GB size Performance”.
9. GET 1000 objects of 1GB size from BlackPearl
	1. Capture start time and job complete time and enter into the “Test Results” document with a heading of “GET 1000 objects of 1GB size Time”.
	2. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET 1000 objects of 1GB size complete or error”.
	3. Capture BlackPearl Performance
		1. Open the BlackPearl User Interface performance screen (see the **BlackPearl User Guide** section titled “View Performance Metrics”). Set pool to cache and resolution to “60s (25hours)” – then download CSV. Save the file as “BPPerformGET1000.csv”.
		2. Capture a screen snip that clearly shows the Performance Screen. Paste the screen snip into the “Test Results” document with a heading of “GET 1000 objects of 1GB size Performance”.
10. Test Cache full condition by writing files and reducing the size of the BlackPearl Cache:
	1. PUT tape library backend in quiescent mode How do they do this?
	2. Contact Spectra Logic to reduce cache size on BlackPearl system. Wait for OK

Is the Cache Reduction API/Internal call complete? I believe so. Mark said they would be available via commands issued directly on the command line of the BlackPearl. We are also working on adding this to the SDKs, but it will not be available until a few weeks after the 2.4 BP release.

* 1. Transfer x files of 1GB (>larger than cache size set) Can we be specific based on minimum cache size?
	2. Capture screen snips that clearly show the application response:
		1. Waiting for the condition to clear. The application should wait patiently for the condition to clear and not hang. Paste the screen snip into the “Test Results” document with a heading of “Cache Full Waiting”
		2. After the condition has cleared. How do we tell them to clear this condition? It is acceptable to have timeout at some point. Paste the screen snip into the “Test Results” document with a heading of “Cache Full cleared or timeout”
1. GET data from tape
	1. PUT 1000 objects of 1GB Size to BlackPearl
	2. Wait for all jobs to ‘complete’ (see the **BlackPearl User Guide** section titled “View DS3 Jobs Information”)
	3. Clear cache :Do we have the Cache Reclamation command? In the Java CLI or other tool? See above
	4. GET first 1/3 X objects
		1. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET 1/3 Data from Tape”.
	5. GET first ¼ X objects
		1. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET ¼ Objects from Tape”.
	6. GET first 1/6 X objects
		1. Capture a screen snip that clearly shows the application completed or error. Paste the screen snip into the “Test Results” document with a heading of “GET 1/6 Objects from Tape”.
2. Request data from BlackPearl that is on a tape that has been ejected from the library (or simply the storage pool?)
	1. From BlackPearl User Interface, eject the bucket that was used for the previous test.
	2. Do they have to actually have to physically eject the tape, just put it in the eject pool?
	3. GET the objects in that bucket
	4. Capture screen snips that clearly show the application response: Waiting for the condition to clear. The application should wait patiently for the condition to clear and not hang. Paste the screen snip into the “Test Results” document with a heading of “Request data on a tape that has been ejected”
	5. Re-import the tape into the storage pool using the “Import Tapes” instructions in the BlackPearl User Guide.
	6. Capture screen snips that clearly show the error condition (Time out or waiting for condition to clear) has cleared. Paste the screen snip into the “Test Results” document with a heading of “Request data on a tape that has been ejected cleared”
3. Resuming a job where a connection has failed when doing an object PUT
	1. Can we do this? I think we need to. Can Ryan or Jeff build a Proxy to do this? How is it controlled? Document the procedure here and collecting test results.
	2. Because of the nature of the HTTP interaction, the client should be able to continue where it left off as long as it knows which files in the job it has successfully sent so far. We have a couple ways that this can be handled. They can save the Job ID and resume the job, they can look and see what files are outstanding by listing all the objects, or in 2.4 they can choose to ignore name conflicts when creating the job in which case BP will only request the files that are new.

**Perform the Following Tests if These Commands are Used in the Application**

1. If the application supports multiple buckets or multiple storage targets that can be mapped to a bucket
	1. Create two buckets
		1. Capture a screen snip that shows how to create the additional buckets. Paste the screen snip into the “Test Results” document with a heading of “Create Two Buckets”.
	2. PUT data to both buckets
		1. Capture a screen snip that shows the application completed with objects in both buckets. Paste the screen snip into the “Test Results” document with a heading of “Data in two buckets”.
	3. GET data from both buckets
		1. Capture a screen snip that shows the application completed with no objects in either bucket. Paste the screen snip into the “Test Results” document with a heading of “Data from two buckets”
	4. Show Screen shots of system, and how to create the profiles and setup multiple buckets?????
2. If the application is capable of creating a DS3 bucket, create a bucket and do the following
	1. set default job priorities (optional) This will be set in the data policy in 2.4 I believe, I would need to double check.
	2. select data policy (optional)
	3. show screen shot of both settings (if applicable)
3. Manually change job priorities – we might want to make this Phase 2. I don’t know if this is explicitly required since they will be able to use the java cli to perform this kind of feature. At the very least I would mark this optional.
	1. List all active jobs in BP – capture screen shot of your application
	2. In BlackPearl GUI, on S3 Jobs – capture screen shot showing the job with the priority
	3. Select and change a jobs priority – capture in screen shot of your application
	4. In BlackPearl GUI, on S3 Jobs – capture screen shot showing the job with the *new* priority
4. Eject a bucket or tape
	1. Edit Tape Eject information: If supported by the application, enter information about the eject location of a tape cartridge and assign it a label
		1. Capture a screen snip that shows how to edit the eject information. Paste the screen snip into the “Test Results” document with a heading of “Edit Tape Eject Information”.
	2. Eject a bucket or a tape
		1. Capture a screen snip Tape Management screen on BlackPearl showing the tape(s) status as offline/ejected (can this be shown on BP GUI?) I assume this is the case but I couldn't confirm it in the User Guide.
5. Host application showing a job is fully persisted:
	1. If application can show whether an object is fully persisted or not by using job polling or using notifications:
		1. PUT data into a bucket
		2. Capture a screen snip that shows the BlackPearl Job Status complete in the BlackPearl S3 Jobs Screen. Paste the screen snip into the “Test Results” document with a heading of “Fully persisted BP Job complete”.
		3. Capture a screen snip from the application showing the same job is not complete/persisted. Paste the screen snip into the “Test Results” document with a heading of “Fully persisted Application Job **not** complete”.
		4. After the job is completed/persisted to tape, capture a screen snip from the application showing the same job is complete/persisted. Paste the screen snip into the “Test Results” document with a heading of “Fully persisted Application Job complete”.
6. If application can list objects in a bucket, show that it can handle (paginate) a large list
	1. PUT 500 objects of 1GB size to BlackPearl
	2. Capture a screen snip from the application showing user interface with objects displayed in a first page. Paste the screen snip into the “Test Results” document with a heading of “Large object list page 1”.
	3. Display the second page of objects in the application.
	4. Capture a screen snip from the application showing user interface with objects displayed in a second page. Paste the screen snip into the “Test Results” document with a heading of “Large object list page 2”.

**Questions that need to be answered:**

1. Steven has proposed that we keep a LTO6 tape drive in the library we use to test/cert. with BlackPearl. And then any mention to chunk size can be kept consistently at 100GBChunk size cannot be set within BlackPearl, correct? BP2.4 is dynamic 100GB up to 1TB based on media size. Its only set outside BlackPearl, i.e. in the Clients Code. Which we will not ask them to change just for testing. We want to test production code without changes
2. Question 2, can we define this as <100GB? In 2.4 the default maximum chunk size will be dynamic depending on the media being used. Will we always have a LTO6 on so it is 100GB?
3. Is the data stored Locally, so it isn’t trying to be transferred in from out side the company? (potential for viruses, etc, BW requirements, time).
4. Question 7: do we allow other sizes than 100GB? In 2.4 the default maximum chunk size will be dynamic depending on the media being used. Will we always have a LTO6 on so it is 100GB? Do we specify that they set it to 100GB?
5. What performance are we expecting, testing for, certifying as pass or fail? Also why there needs to be some aspect of Performance testing, whether we pass or fail them on throughput isn’t as important, but from a usability perspective I personally wouldn’t certify an application that doesn’t usably function. Our products are based on throughput, which is essentially what we are selling to a customer and if an application cannot provide a usable throughput to BlackPearl I’d feel slighted if I was a customer buying that ‘certified’ solution that has very bad throughput.
6. +Do we compile the certification/testing plan into a document that can be distributed?\
7. Will we allow customers to perform their own certification/testing?
8. Do they fail certification if it is functional, but performance isn’t there? Steven: yes. Jeff: prove they are saturating. 1Gb connection is 125MB/s, 10Gb or more should get 800MB/s – max throughput of drives.
9. Do we test single 1Gb/s then a 10Gb/s? P/F criteria for both?
10. How do we handle customer test data? Verde? Customer to provide test data (covered by legal agreement?
11. Who is the technical resource to review the completed test plan to determine if the information we received is correct?
12. Who is the technical resource that will troubleshoot when the client says it is our hardware and/or software is not behaving as indicated or as it did in the simulator?
13. Who owns running this test program? Reviewing results of the questionnaire? Test results?