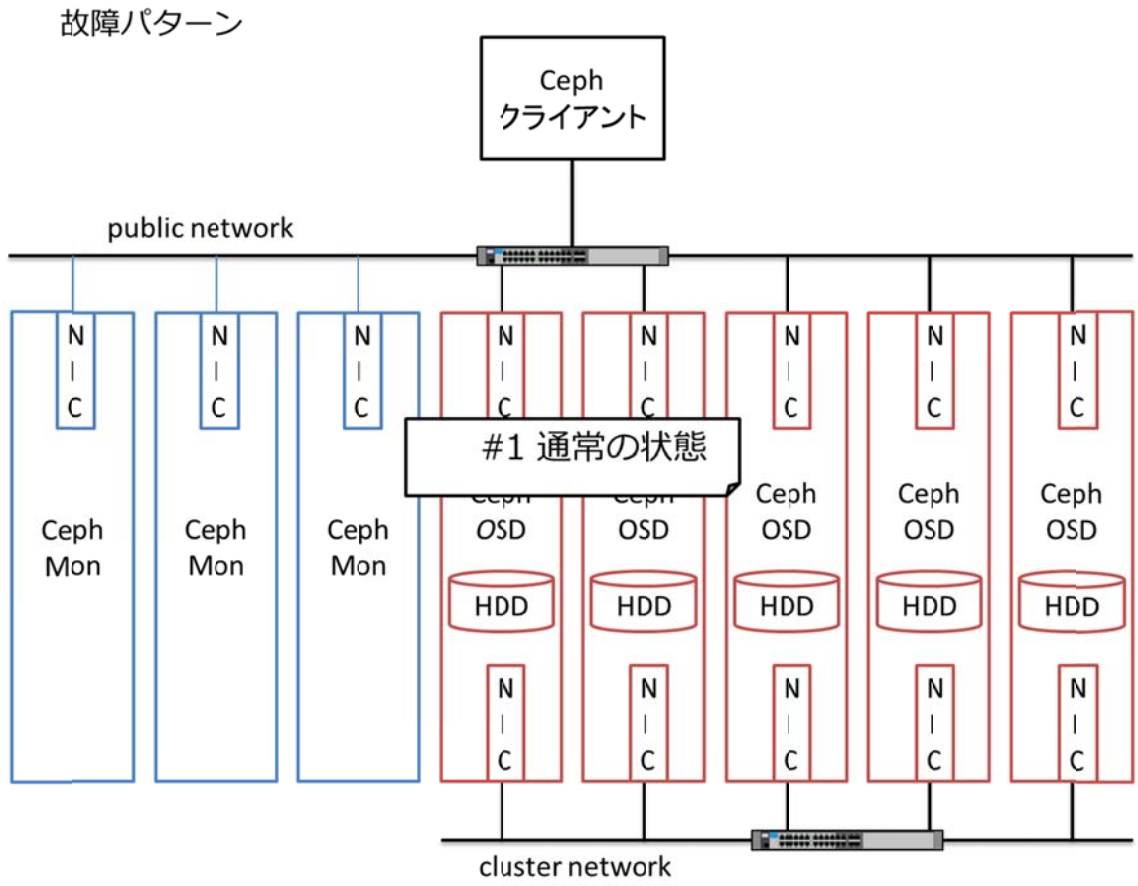
3. OpenStack/Ceph 異常系テスト (1)

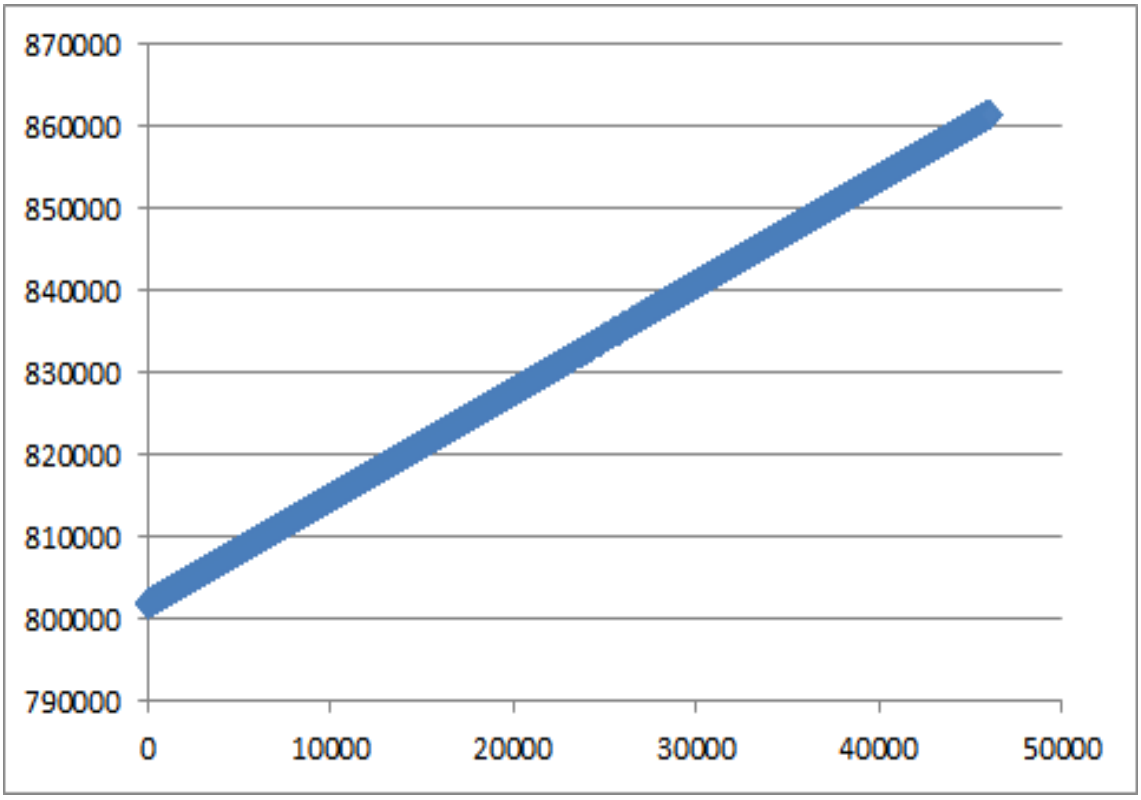
3.4. テスト結果

具体的なテスト項目とその結果について説明します。
始めに、各テスト項目を表現した図の見方を説明します。テスト項目は、故障するハードウェアのドメインの状態とシステム構成のパラメータの値のパターンです。



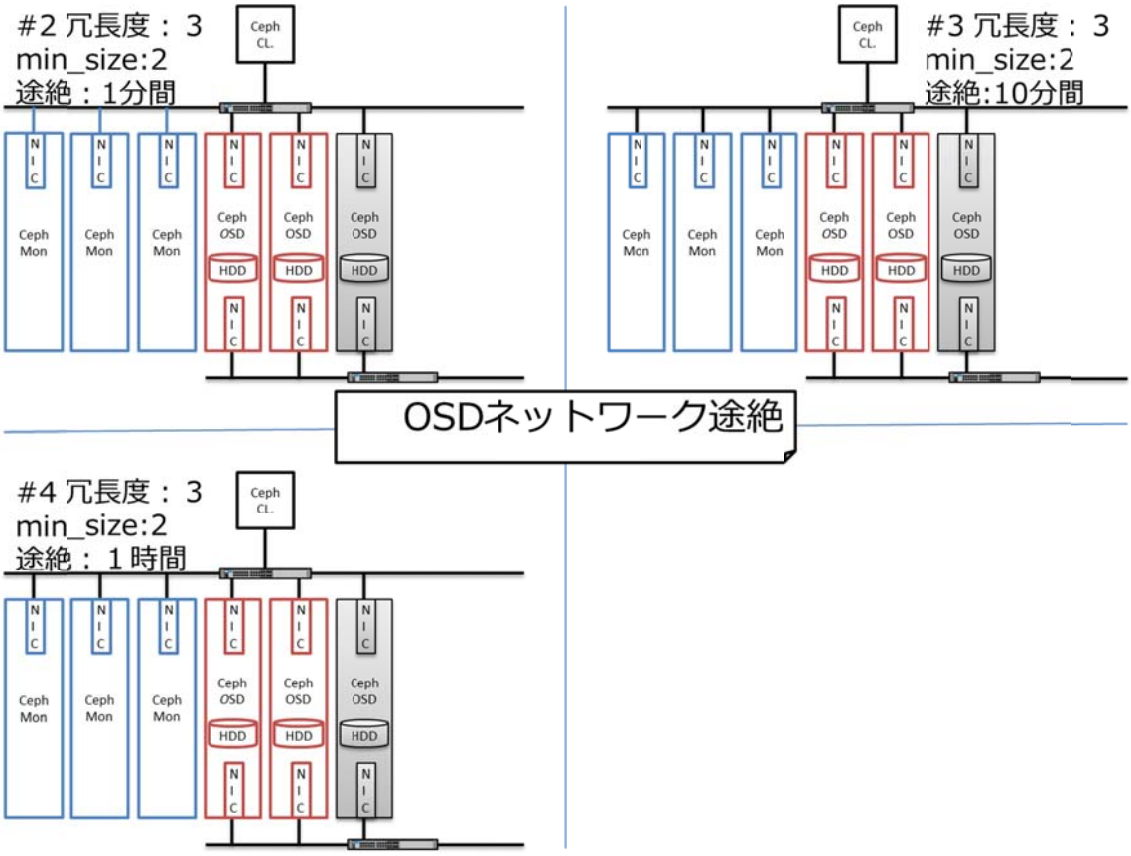
この図はテスト項目 #1 を表しており、全てのテスト項目の基準となります。
この図では 3 台の Ceph MON サーバーと 5 台の Ceph OSD サーバーがあり、全てのサーバーが public network に接続されており、全ての Ceph OSD サーバーが cluster network に接続されている状態を表しています。
故障するハードウェアのドメインは灰色で示されます。この図では灰色のドメインは存在せず、すなわち、通常の状態を表します。
システム構成のパラメータについては、注目すべきものについて余白に記載します。この図では何も記載されていません。

つぎに下記の図の見方を説明します。



OpenStack/Ceph 異常系テスト (1) では、障害が及ぼす様々な影響のうち、アプリケーションの I/O への影響に着目します。
上記の図は、障害 (＝テスト項目) 毎に示されるアプリケーション I/O の進捗を表します。縦軸は単位時間を表し、横軸はファイル I/O 用のボリュームに作成されたファイルの数を表します。データは、RBD クライアント上で DirectIO を使用してワークロードを生成している I/O プロセスでの結果を使用しています。
テスト項目 #1 は通常の状態での結果です。時間の経過におよそ比例して作成されるファイル数が増加しています。このグラフは、ファイルを 1 つ作成する毎に点をプロットしています。 (複数の点を結んだ直線でないことに注意してください。)

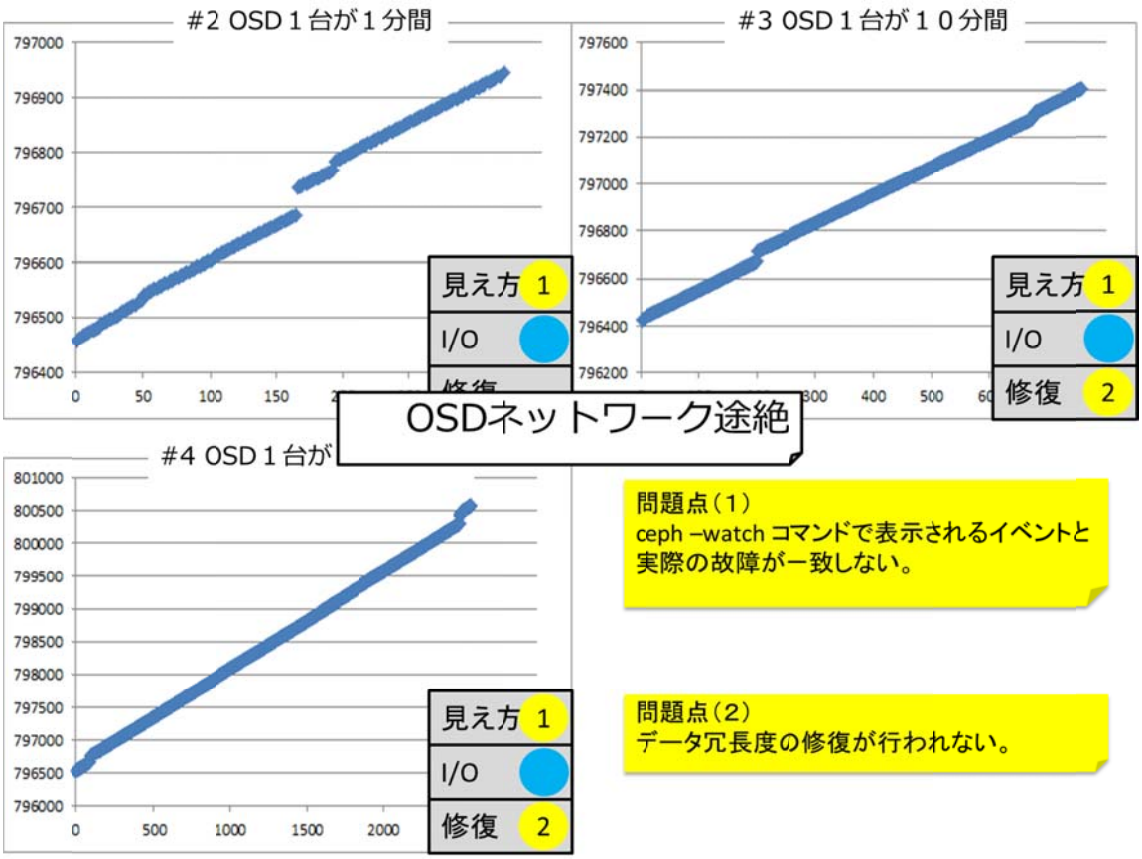
もし、ファイル I/O でエラーが発生すると I/O プロセスは終了するのでその時点以降、ファイル数はプロットされません。もし、ファイル I/O でハングが発生すると、I/O プロセスもハングするので、ハングしている間はファイル数がプロットされません。ハングが解けた時点で再びファイル数がプロットされますので、点の集まりはハングが持続した長さだけ上下に分断されるはずです。また、ファイル I/O の進捗が滞ると点の集まりの傾きが急になり、逆に進捗が捗ると傾きが緩やかになるはずです。



テスト項目 #2 から #4 です。

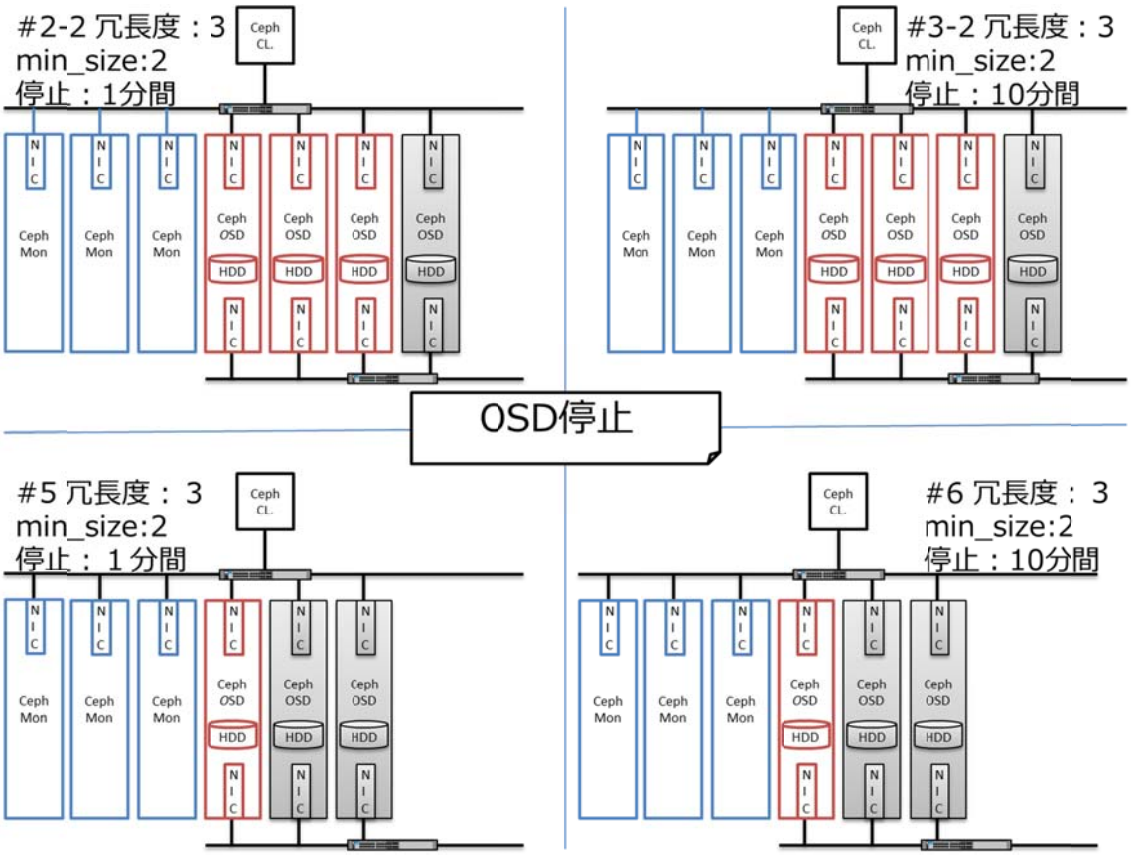
ここでは Ceph OSD サーバーの 1 台でネットワーク途絶が発生しています。各サーバーは 2 つの NIC が両方とも灰色になっていますので、public network と cluster network の両方のネットワークから途絶しています。

システム構成のパラメータについては、注目すべきものについて余白に記載します。ネットワーク途絶の状態が持続した時間が余白に記載されています。また、冗長度と min_size の値がそれぞれ記載されています。冗長度は pool size の値で、プールのデータ冗長度を指します。Ceph OSD デーモンは 3 台で、pool size が 3 なので、全てのプールはこれら 3 台の Ceph OSD でデータ冗長化されているはずです。よって、1 台がネットワーク途絶の間は全てのプールでデータ冗長度に満たない状態です。min_size は pool min_size の値で、I/O を継続するデータ冗長度の設定です。健全な Ceph OSD デーモンが 2 台残っている状況で min_size が 2 なので全てのプールで I/O は継続されるはずです。



テスト項目 #2 から #4 です。
いずれも点の集まりが不連続な箇所があるので、一時的な I/O ハングが発生していたと思われます。
各図の右下に示されている表示の見方について説明します。
「見え方」はコマンドの表示に発生したハードウェア障害を特定するような情報が含まれていたかどうかを指します。青色のマークは、コマンドがハードウェア障害を特定するような情報を表示したことを表します。黄色のマークは、コマンドが何等かの表示を行ったが、ハードウェア障害を直接示すものではなかったことを表します。赤色のマークは、コマンドの表示がなかったことを表します。
「I/O」は障害が及ぼす様々な影響のうち、アプリケーションの I/O への影響が許容できるものかどうかを示します。青色のマークは、影響なしまたは影響が軽微であったことを表します。黄色のマークは、影響があったものの I/O は 継続されたこと表します。赤色のマークは 、I/O が継続されなかったことを表します。
「修復」はデータ冗長度が維持されたかどうかを示します。マークなしは、障害の持続時間が短く、データ冗長度の修復が開始よりも前に障害が復旧したことを表します。青色のマークは、データ冗長度が維持されたことを表します。黄色のマークは、データ冗長度が下がったものの、データが失われることはなく、ハードウェア障害を取り除くことでデータ冗長度が維持されることを表します。赤色のマークは、データが失われたことを表します。
青色、黄色、赤色のマークに数値が記載されている場合は具体的な記述へのポインタです。上記の図の場合、問題点 (1) 、問題点 (2) を指しています。
問題点 (1) に関してより具体的には Ceph OSD サーバーのネットワーク途絶ではなく Ceph OSD サーバーのダウンとして表示されたため、コマンドの表示と実際の障害が一致していなかったことを示しています。
問題点 (2) に関しては、スペアの Ceph OSD デーモンがないためデータ冗長度の修復が開始されなかっ

たが、ハードウェア障害を取り除くことでデータ冗長度が維持されたことを示しています。スペアの Ceph OSD を用意しておくことで回避可能な問題です。



テスト項目 #2-2 から #6 です。

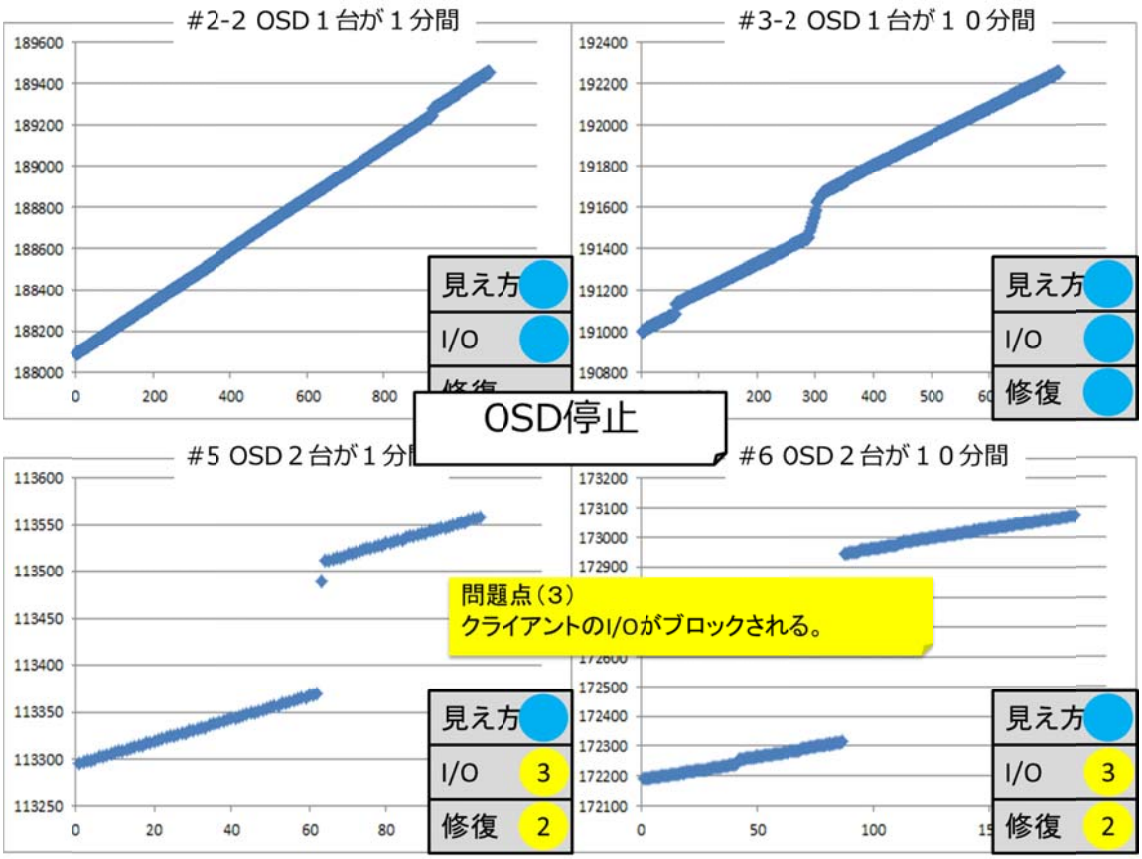
ここでは Ceph OSD サーバーの 1 台または 2 台でサーバーダウンが発生しています。

余白に記載されている「停止： 1 分間」、「停止： 10 分間」は、サーバーダウンの状態がそれぞれ 1 分間以上、10 分間以上持続したことを表します。

「冗長度」はいずれのテスト項目も 3 なので、プールはいずれか 3 台の Ceph OSD デーモンで構成されています。

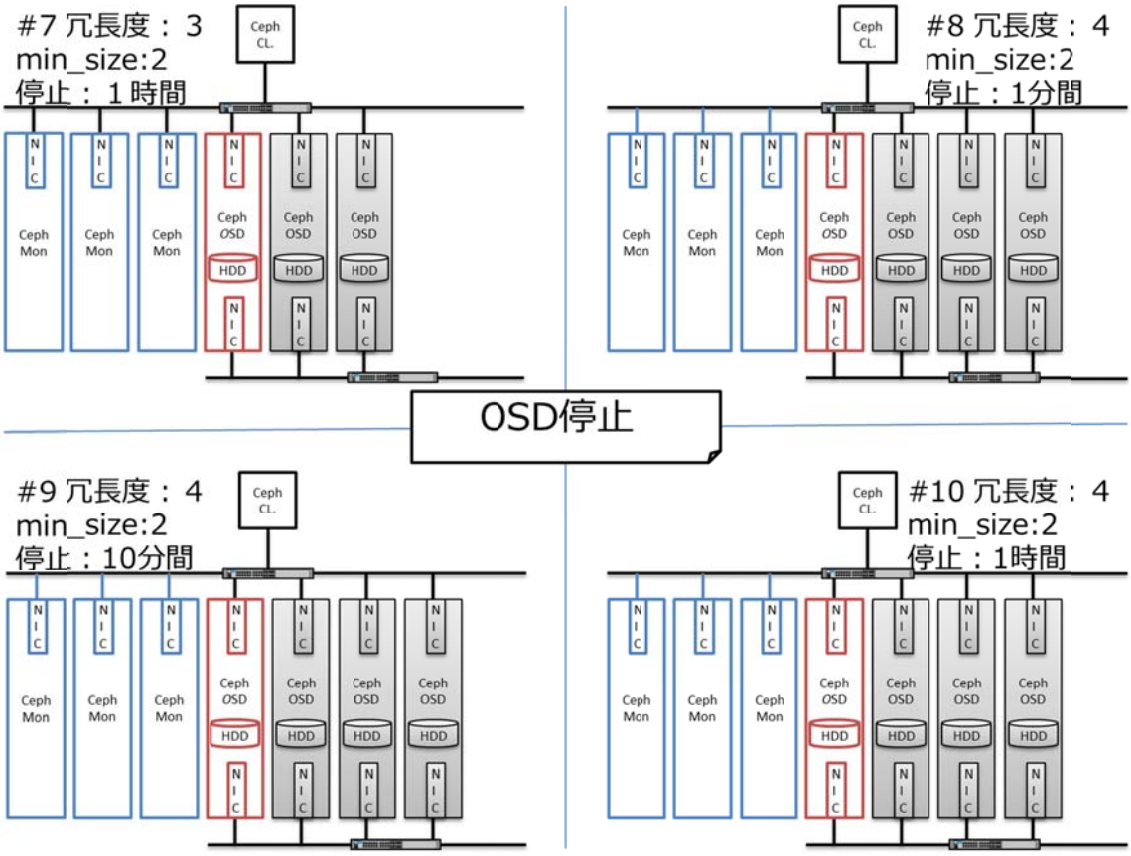
「min_size」はいずれのテスト項目も 2 なので、プールを構成する 3 台の Ceph OSD デーモンのうち 1 台がダウンしても I/O を継続します。

よって、サーバーダウン中、テスト項目 #2-2 と #3-2 では I/O が継続され、#5 と #6 では I/O がブロックされるはずです。



グラフを確認すると、サーバーダウン中、テスト項目 #2-2 と #3-2 では I/O が継続されていたこと、テスト項目 #5 と #6 では I/O がブロックされていたことが分かります。テスト項目 #5 と #6 の I/O プロセスはいずれも終了していないので、ダウンしたサーバーが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。サーバーダウン中、テスト項目 #3-2 では I/O の進捗が遅くなっていたようです。これは、サーバーダウン中にデータ冗長度の修復処理が動作した影響の I/O 性能劣化と思われます。テスト項目 #2-2 で I/O 性能劣化が見られないのは、サーバーダウンの状態が持続する時間 (1 分間以上) が短く、データ冗長度の修復処理が開始される前にダウンしたサーバーが復旧したためと思われます。

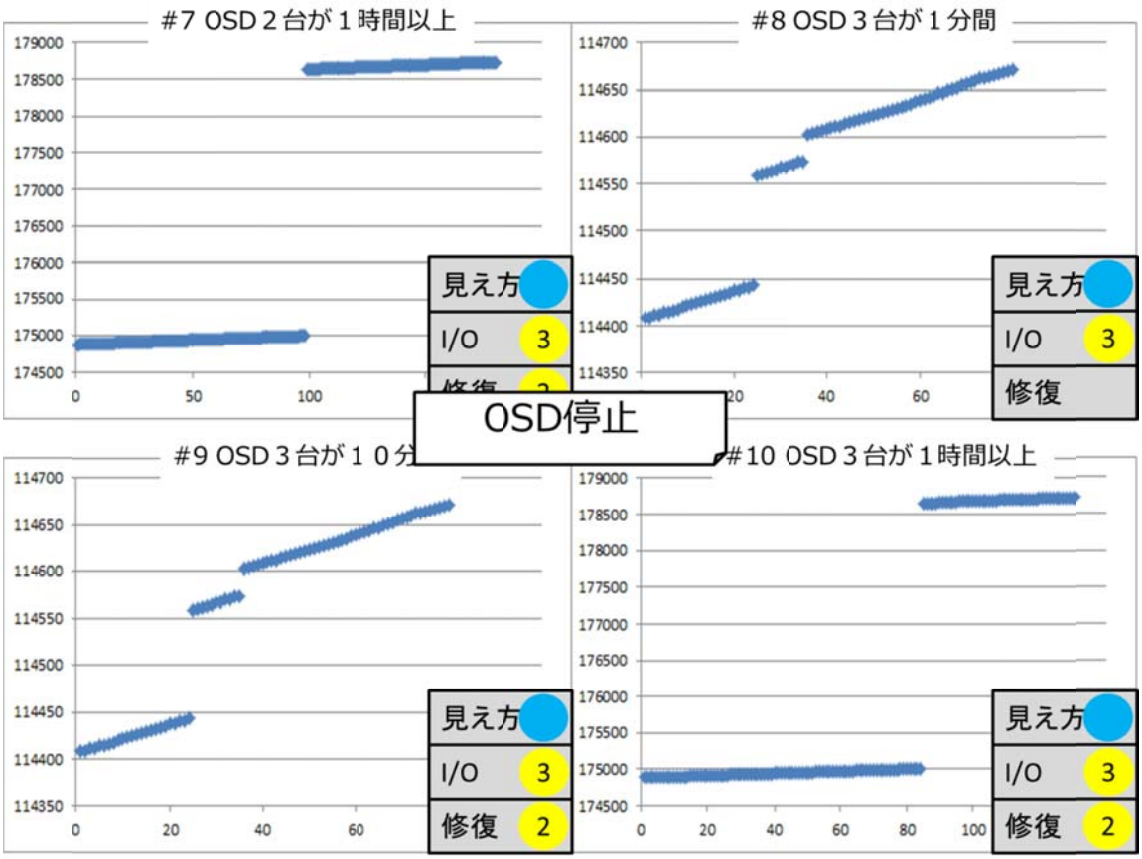
各図の右下に示されている表示については最後にまとめることにして、テスト項目毎の紹介は省略します。



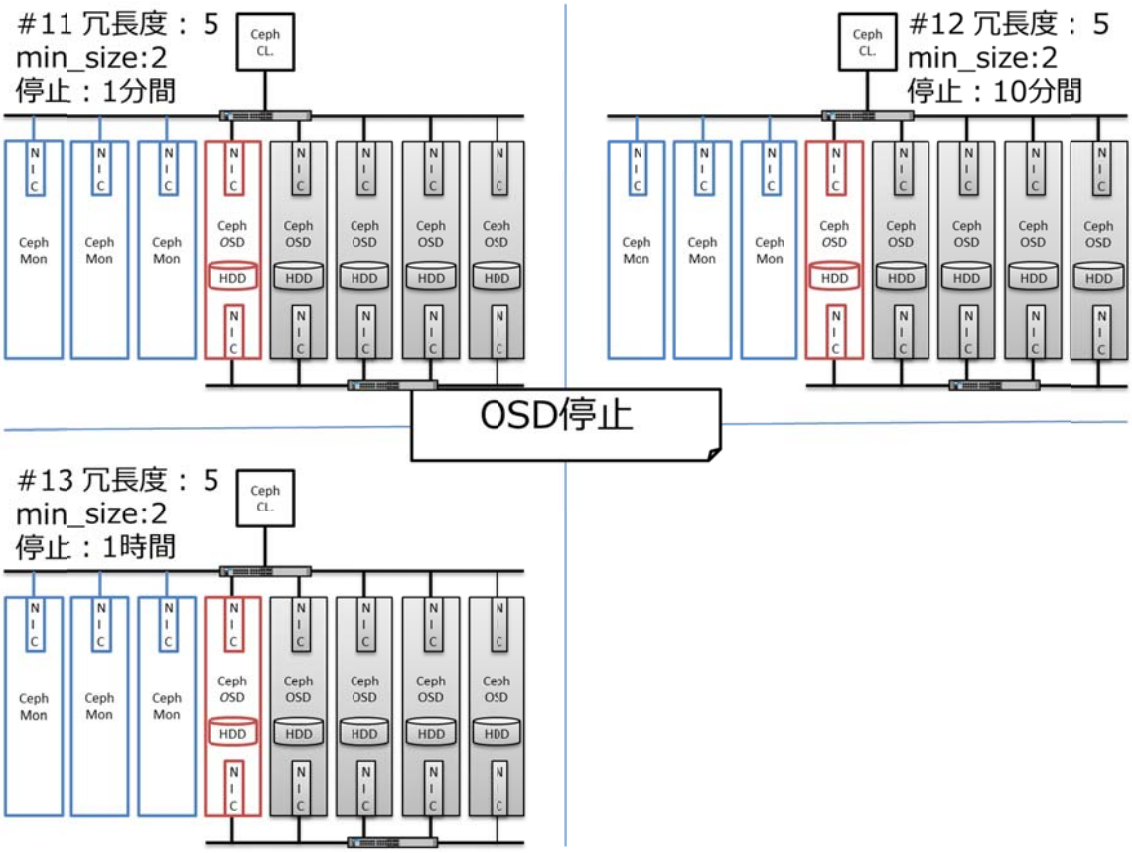
Ceph OSD サーバーダウンの項目が続きます。テスト項目 #7 から #10 です。

ここでは Ceph OSD サーバーの 2 台または 3 台でサーバーダウンが発生しています。先程のテスト項目に比べて、データ冗長度とダウンする Ceph OSD サーバーの数が増えています。

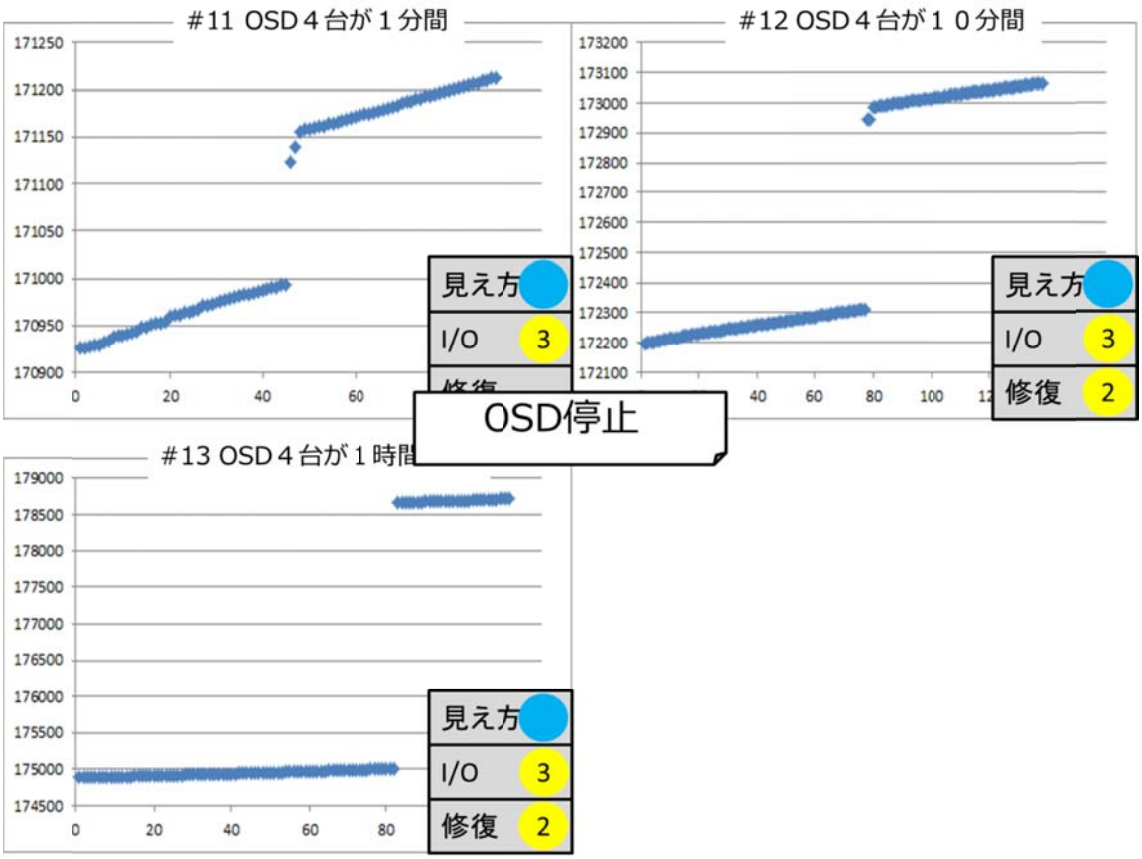
「min_size」はいずれのテスト項目も 2 に対し、サーバーダウン中の健全な Ceph OSD デーモンが 1 台しか残らない状況です。よって、サーバーダウン中、いずれの項目も I/O がブロックされるはずです。



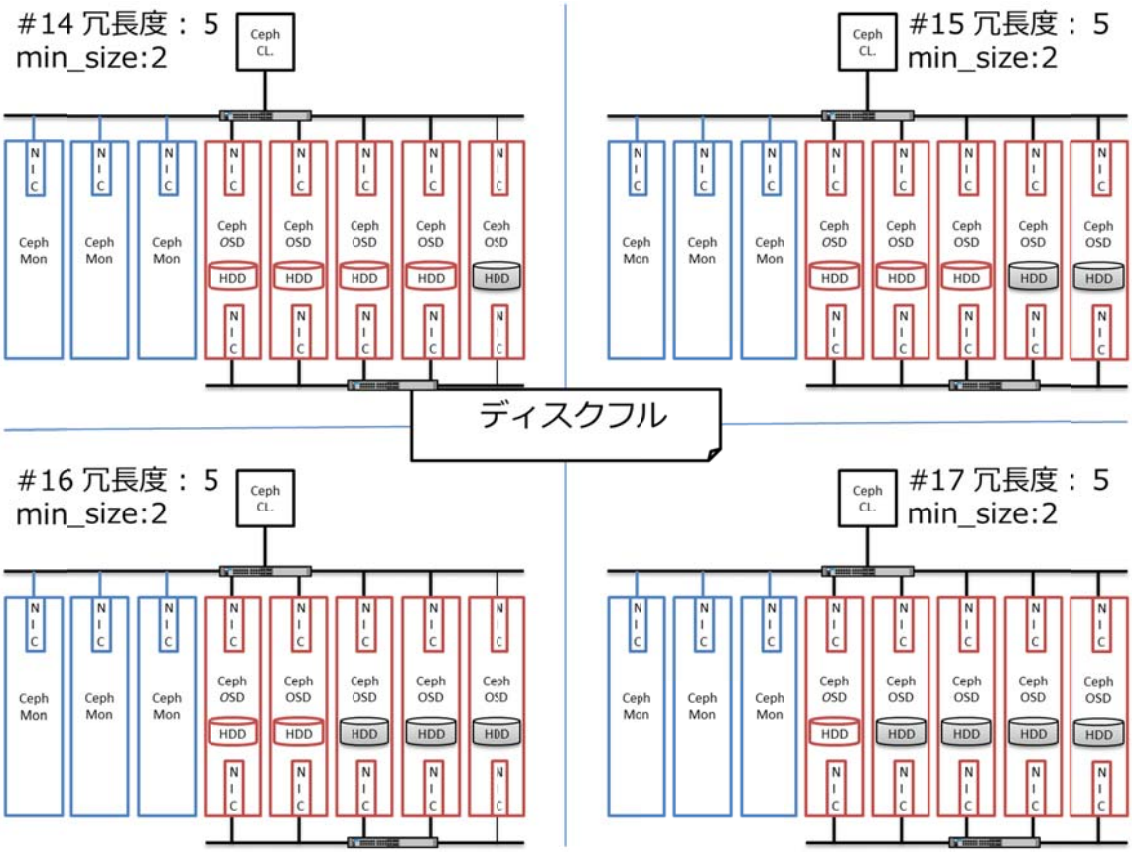
テスト項目 #7 から #10 ではサーバーダウン中 I/O がブロックされていたことが分かります。
I/O プロセスはいずれも終了していないので、ダウンしたサーバーが復旧した時点で I/O のブロックが解け、
I/O エラーは発生していないことが分かります。



Ceph OSD サーバーダウンの項目が続きます。テスト項目#11 から#13 です。
ここでは Ceph OSD サーバーの 4 台でサーバーダウンが発生しています。先程のテスト項目に比べて、さらにデータ冗長度とダウンする Ceph OSD サ ーバーの数が増えています。
「min_size」はいずれのテスト項目も 2 に対し、サーバーダウン中の健全な Ceph OSD デーモンが 1 台しか残らない点は同じです。

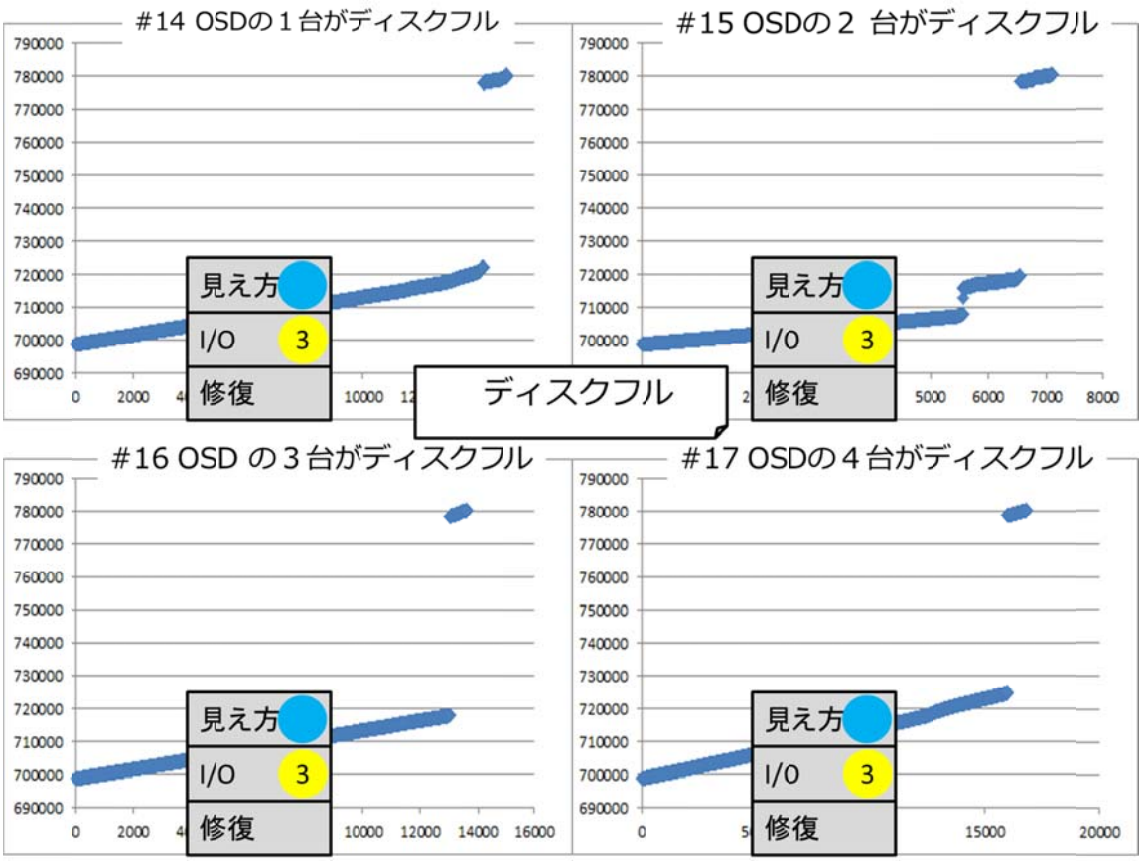


テスト項目 #5 から #10 と同様に、サーバーダウン中 I/O がブロックされていたことが分かります。ここまでの項目が Ceph OSD サーバーダウンに関連するものです。

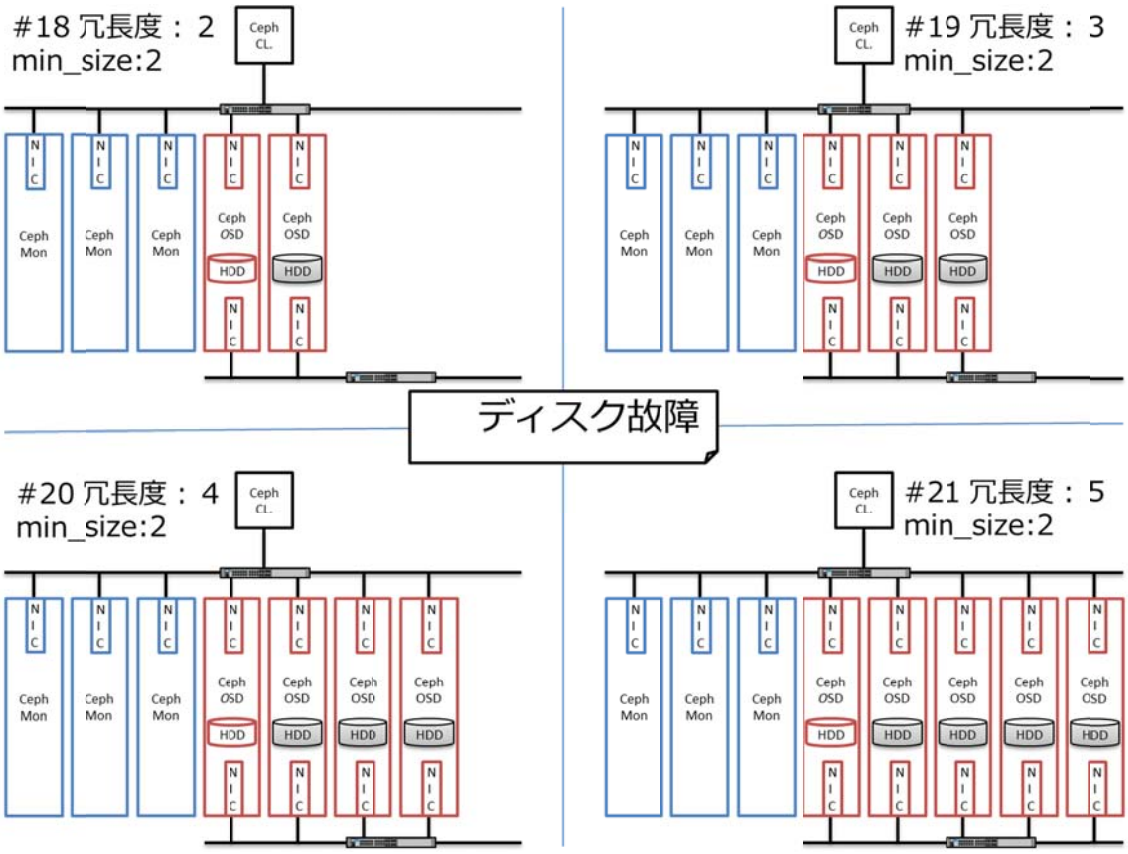


ここからの項目は Ceph OSD のディスクフルに関するものです。テスト項目 #14 から #17 です。
ここでは、データ冗長度の設定が 5 の環境で、1 台から 4 台の Ceph OSD デーモンでディスクフルが発生しています。ディスクフルの状態は、Ceph OSD デーモン毎のディスク使用率の値が full ratio パラメータの値に達した状態を指します。full ratio パラメータの値は設定で変更可能です。full ratio パラメータの他に near full ratio パラメータがあり、full ratio よりも低い値を設定します。
ある Ceph OSD デーモンのディスク使用率が near full ratio に達した時点でワーニングメッセージが出力されるようになり、full ratio に達した時点で I/O がブロックされます。

なお、1 台の Ceph OSD がディスクフルの状態となった時点で既存オブジェクトの伸長や新規オブジェクトの作成ができなくなるため、2 台以上の Ceph OSD がディスクフルの状態は full ratio の設定を変更することで作成します。すなわち、まずデータ冗長度の設定が 1 のダミープールを複数作成します。データ冗長度の設定が 1 の場合、プールを構成する PG は 1 台の Ceph OSD で構成されます。次に、Ceph OSD を選択し、当該 Ceph OSD で構成された PG で構成されたダミープールを選択し、当該ダミープールにダミーオブジェクトを配置することで各 Ceph OSD のディスク使用率が段階的に異なる状態を作成します。
次に full ratio の値の設定を変更することで意図した台数の Ceph OSD のみがディスクフルの状態を作成します。



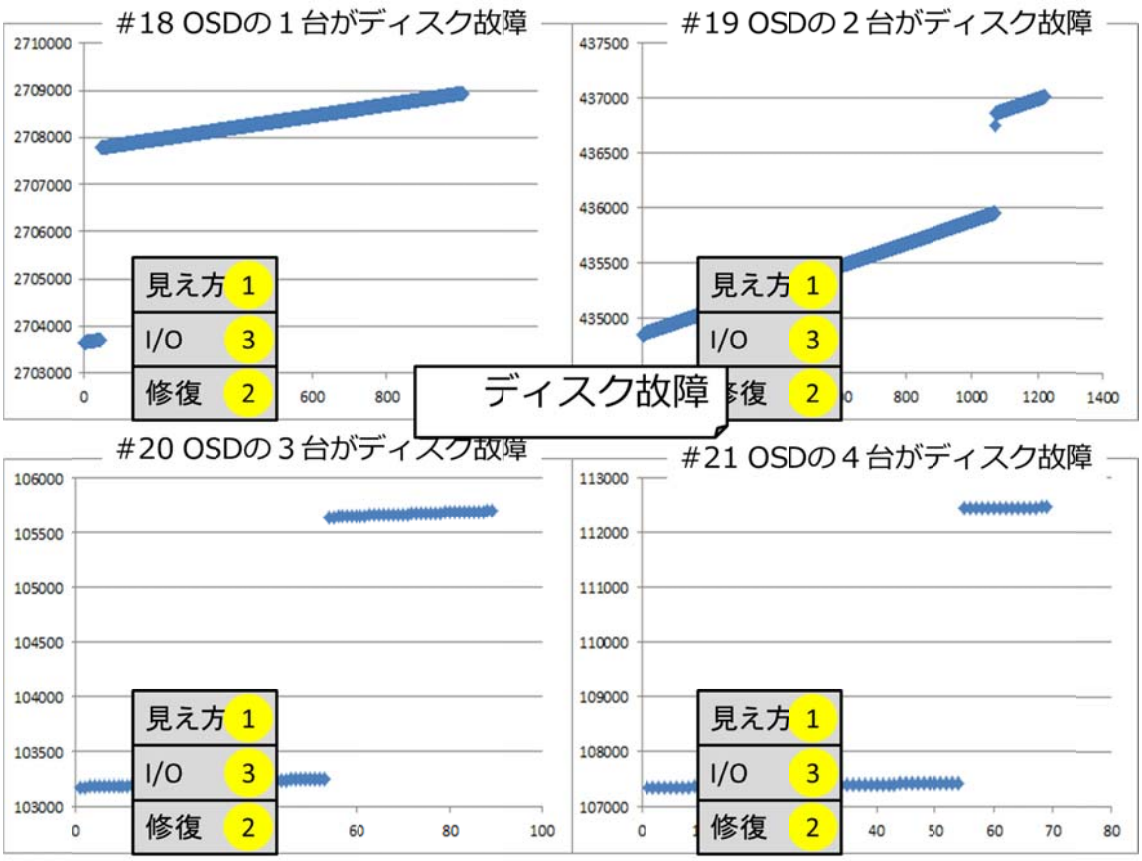
グラフを確認すると、テスト項目 #14 から #17 は I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、ディスクフルが解消した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。



(図)

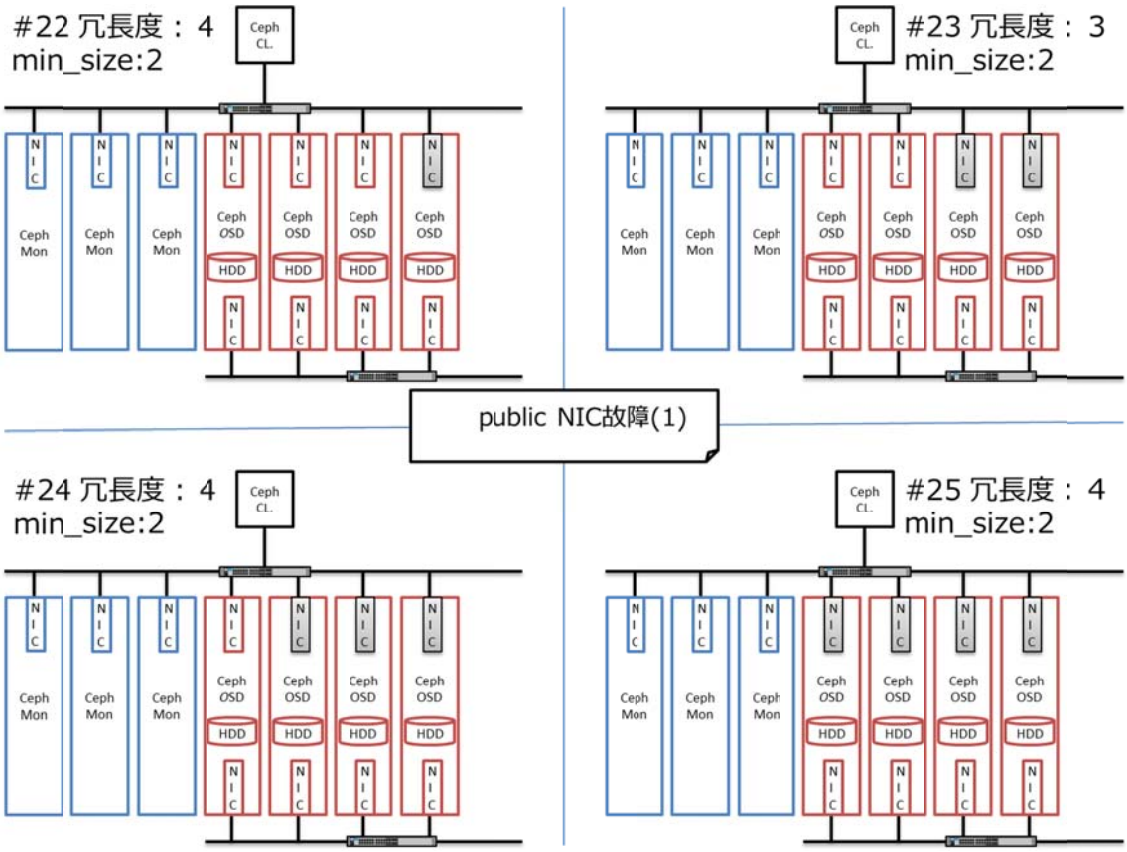
ここからの項目は Ceph OSD のディスク故障に関するものです。テスト項目#18 から#21 です。

ここでは、データ冗長度の設定が 2 から 5 の環境で、1 台から 4 台の Ceph OSD デーモンでディスク故障が発生しています。「min_size」はいずれのテスト項目も 2 に対し、ディスク故障中の健全な Ceph OSD デーモンが 1 台しか残らない状況です。よって、ディスク故障中、いずれの項目も I/O がブロックされるはずです。



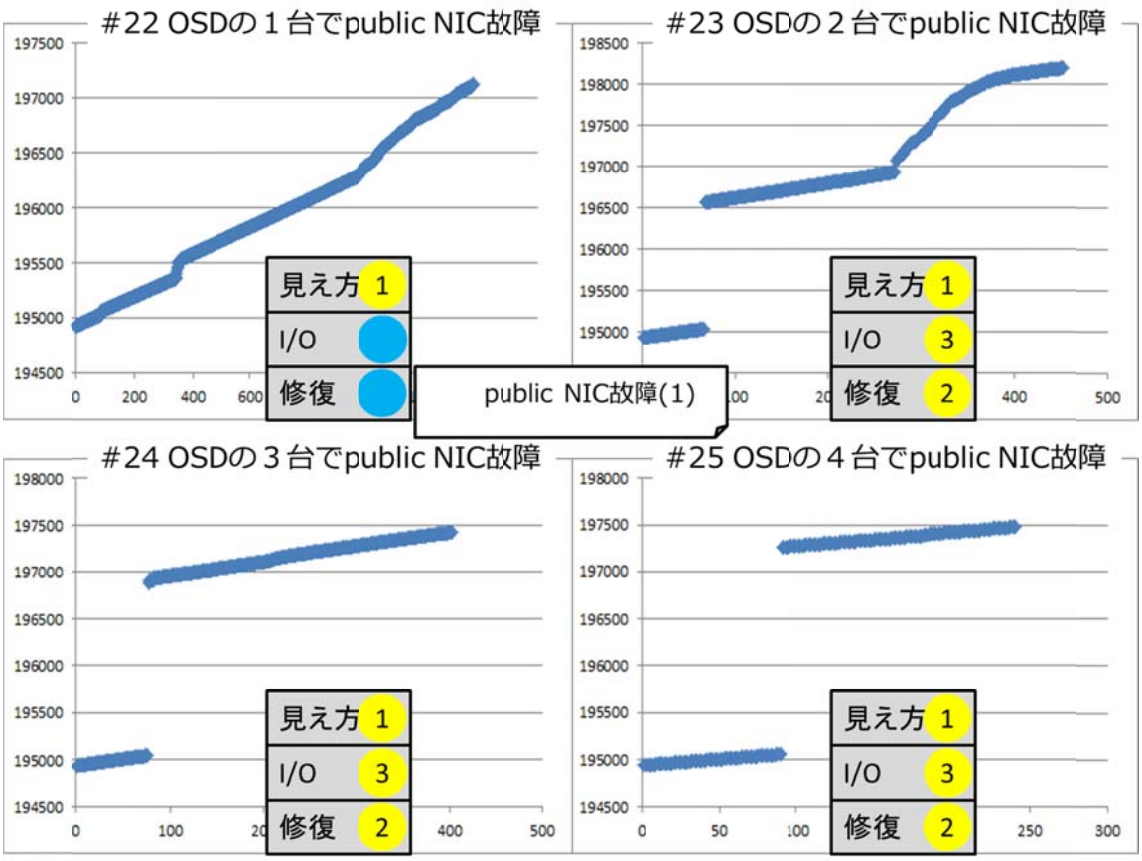
テスト項目 #18 から #21 ではディスク故障中 I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、ディスクが故障したサーバーが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。

ディスクが故障したサーバーの復旧は、当該 Ceph OSD を一旦削除した後、故障したディスクを交換し、新規 Ceph OSD として再度追加することで行っています。



ここからの項目は Ceph OSD の public network 側の NIC 故障に関するものです。テスト項目 #22 から #25 です。

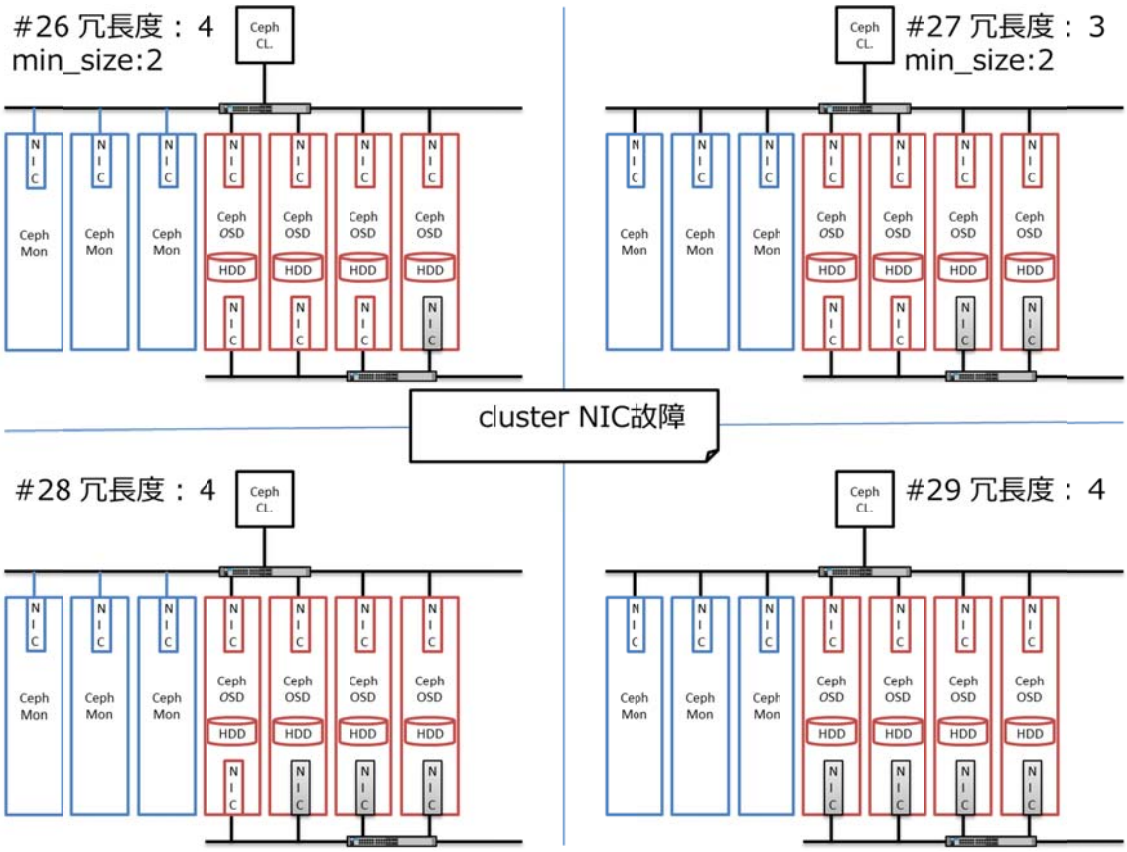
ここでは、データ冗長度の設定が 3 または 4 の環境で、1 台から 4 台の Ceph OSD デーモンで NIC 故障が発生しています。「min_size」はいずれのテスト項目も 2 に対し、テスト項目 #23 から #25 では健全な Ceph OSD が 2 台を下回る状況です。よって、NIC 故障中、いずれの項目も I/O がブロックされるはずです。



テスト項目 #23 から #25 では NIC 故障中 I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、NIC が故障したサーバーが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。

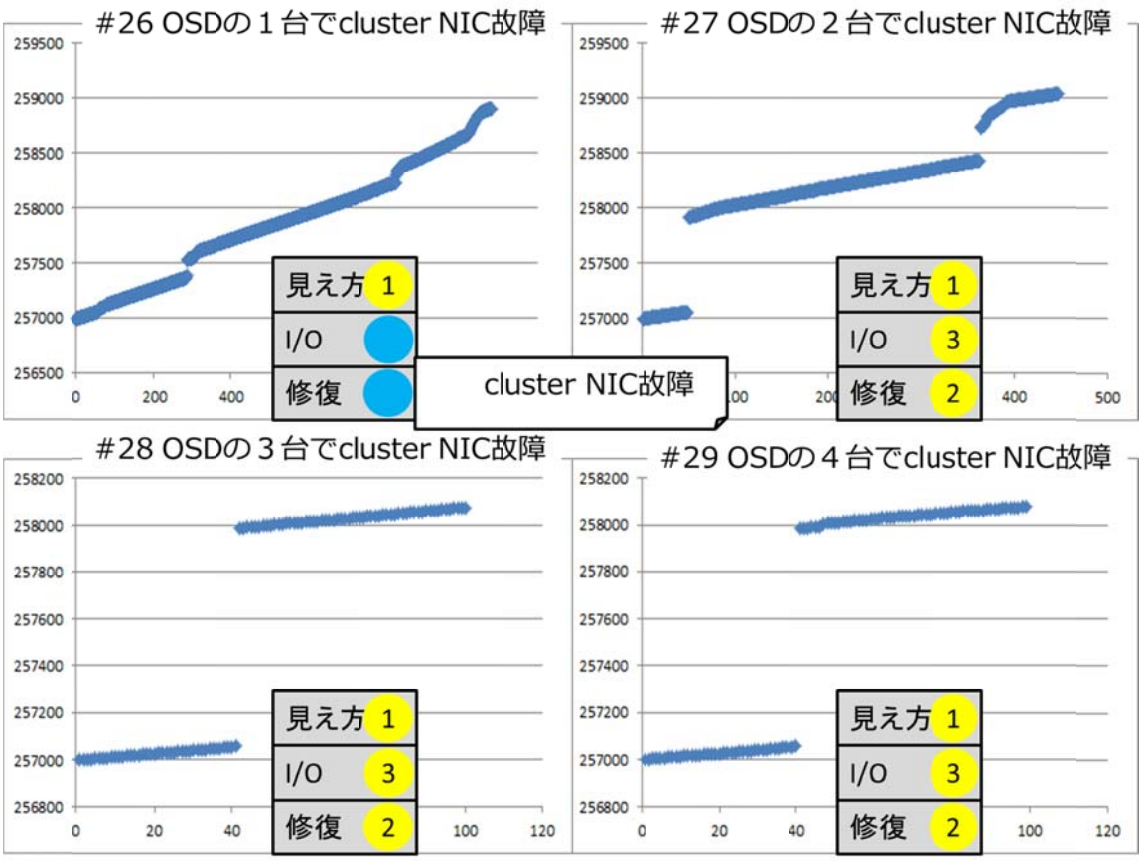
NIC が故障したサーバーの復旧は、テスト項目 #22 と #23 は当該 Ceph OSD を一旦削除した後、故障した NIC を交換し、新規 Ceph OSD として再度追加することで行っています。

テスト項目 #22 と #23 については当該 Ceph OSD をシャットダウンした後、故障した NIC を交換し、再起動することでも復旧可能です。テスト項目#24 と#25 は当該 Ceph OSD をシャットダウンした後、故障した NIC を交換し、再起動することで行っています。テスト項目 #24 については当該 Ceph OSD を一旦削除した後、故障した NIC を交換し、新規 Ceph OSD として再度追加することでも復旧可能です。



ここからの項目は Ceph OSD の cluster network 側の NIC 故障に関するものです。テスト項目 #26 から #29 です。

ここでは、データ冗長度の設定が 3 または 4 の環境で、1 台から 4 台の Ceph OSD デーモンで NIC 故障が発生しています。「min_size」はいずれのテスト項目も 2 に対し、テスト項目 #27 から #29 では健全な Ceph OSD が 2 台を下回る状況です。よって、NIC 故障中、いずれの項目も I/O がブロックされるはずです。

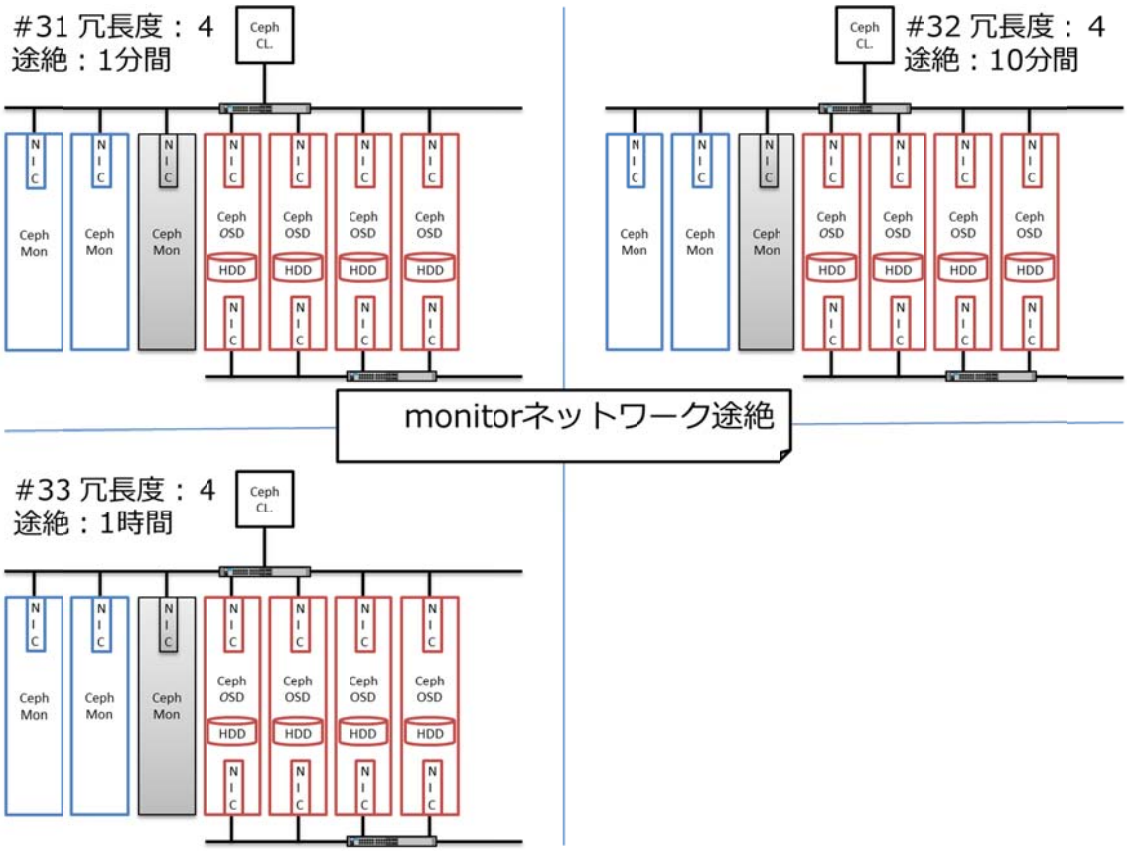


テスト項目 #27 から #29 では NIC 故障中 I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、NIC が故障したサーバーが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。

NIC が故障したサーバーの復旧は、テスト項目 #26 と# 27 は当該 Ceph OSD を一旦削除した後、故障した NIC を交換し、新規 Ceph OSD として再度追加することで行っています。

テスト項目 #26 と #27 はついては当該 Ceph OSD をシャットダウンした後、故障した NIC を交換し、再起動することでも復旧可能です。テスト項目 #28 と #29 については当該 Ceph OSD をシャットダウンした後、故障した NIC を交換し、再起動することで行っています。

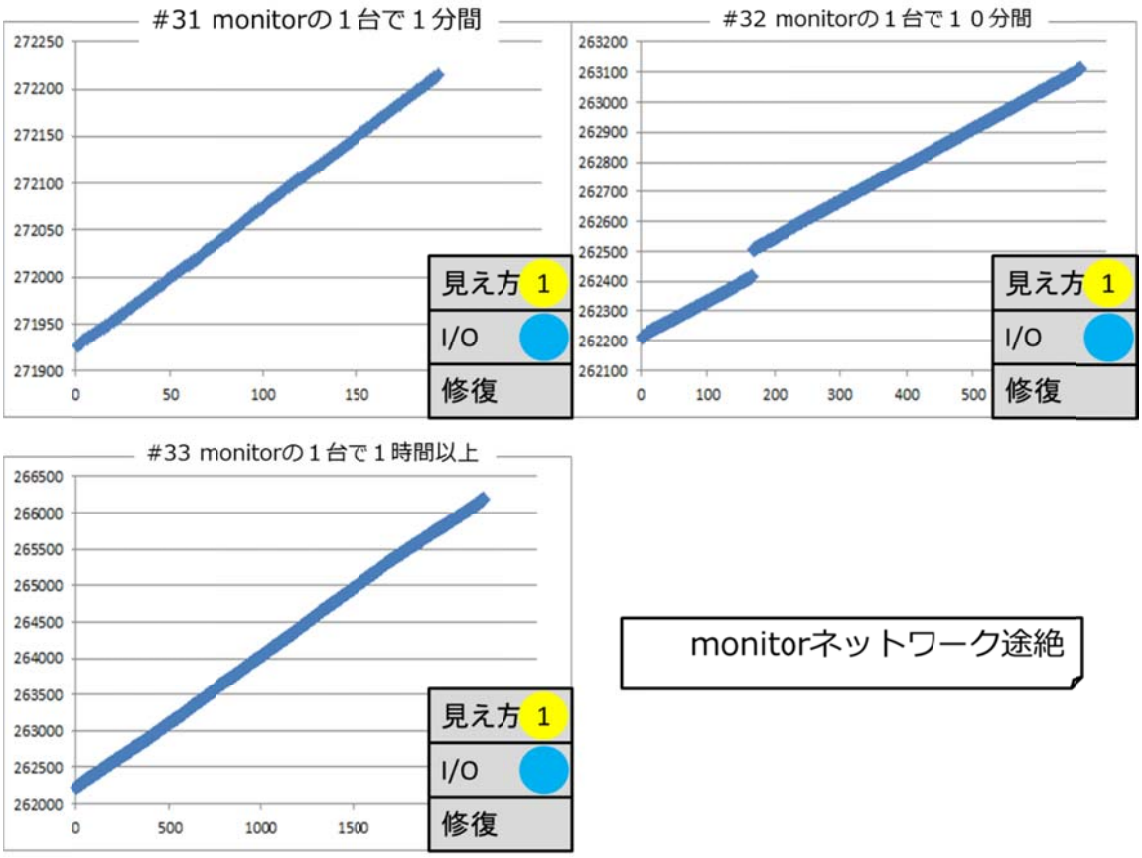
テスト項目 #2 8 については当該 Ceph OSD を一旦削除した後、故障し た NIC を交換し、新規 Ceph OSD として再度追加することでも復旧可能です。



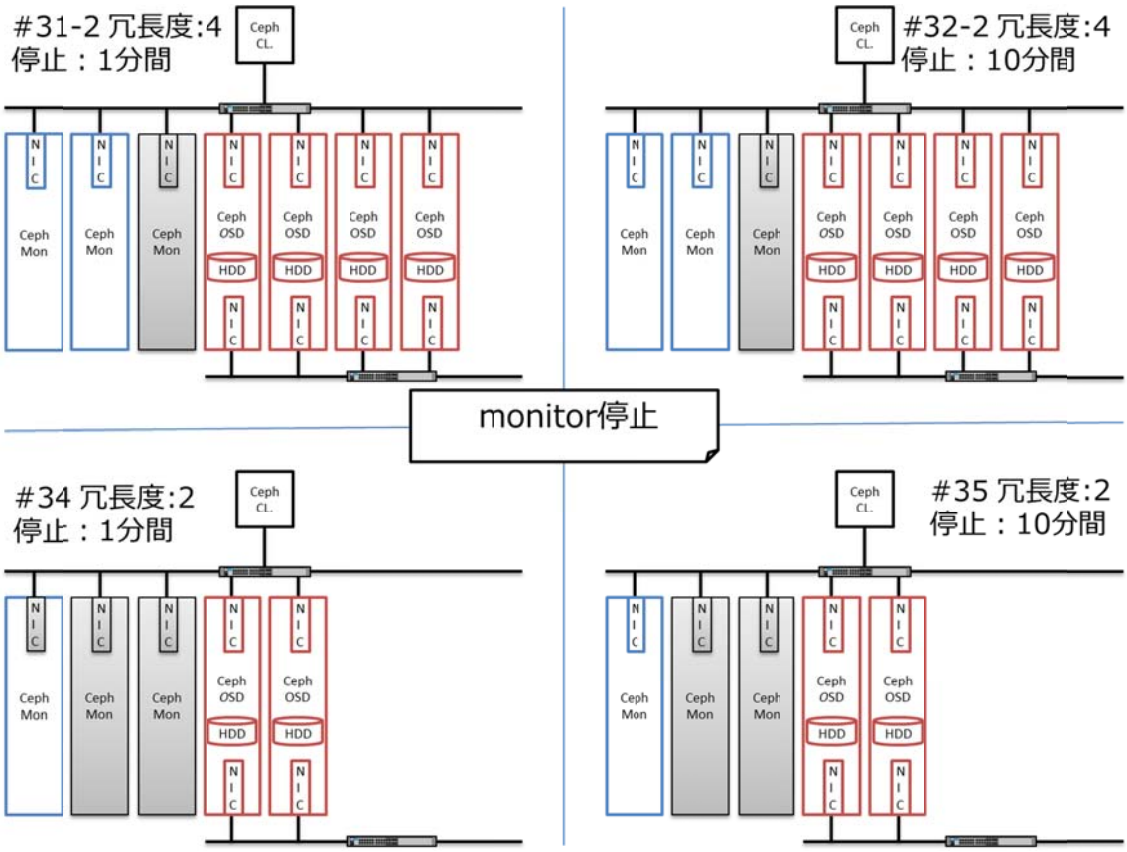
テスト項目 #31 から #33 です。

ここでは Ceph MON サーバーの 1 台でネットワーク途絶が発生しています。各サーバーは、public network のネットワークから途絶しています。cluster network のネットワークには接続されていません。ネットワーク途絶の状態が持続した時間が余白に記載されています。

Ceph は中央のサーバーがなく、各 Ceph クライアントが Ceph OSD とオブジェクトを直接にやりとりします。また、各 Ceph OSD はオブジェクトの複製を別のノード上に作成し、高可用性を確保します。このときの各 Ceph OSD 間も直接にやりとりします。Ceph クライアントと Ceph OSD はオブジェクトの所在を計算するのに CRUSH アルゴリズムを使用します。CRUSH アルゴリズムは、各 Ceph クライアント、各 Ceph OSD が Ceph ストレージ・クラスタのトポロジーを知っていることに依存します。Ceph ストレージ・クラスタのトポロジーはクラスタマップで管理されます。Ceph MON はクラスタマップを管理します。Ceph クライアント、Ceph OSD はクラスタマップのコピーを持ち、Ceph MON に問い合わせることなくデータにアクセスすることができます。

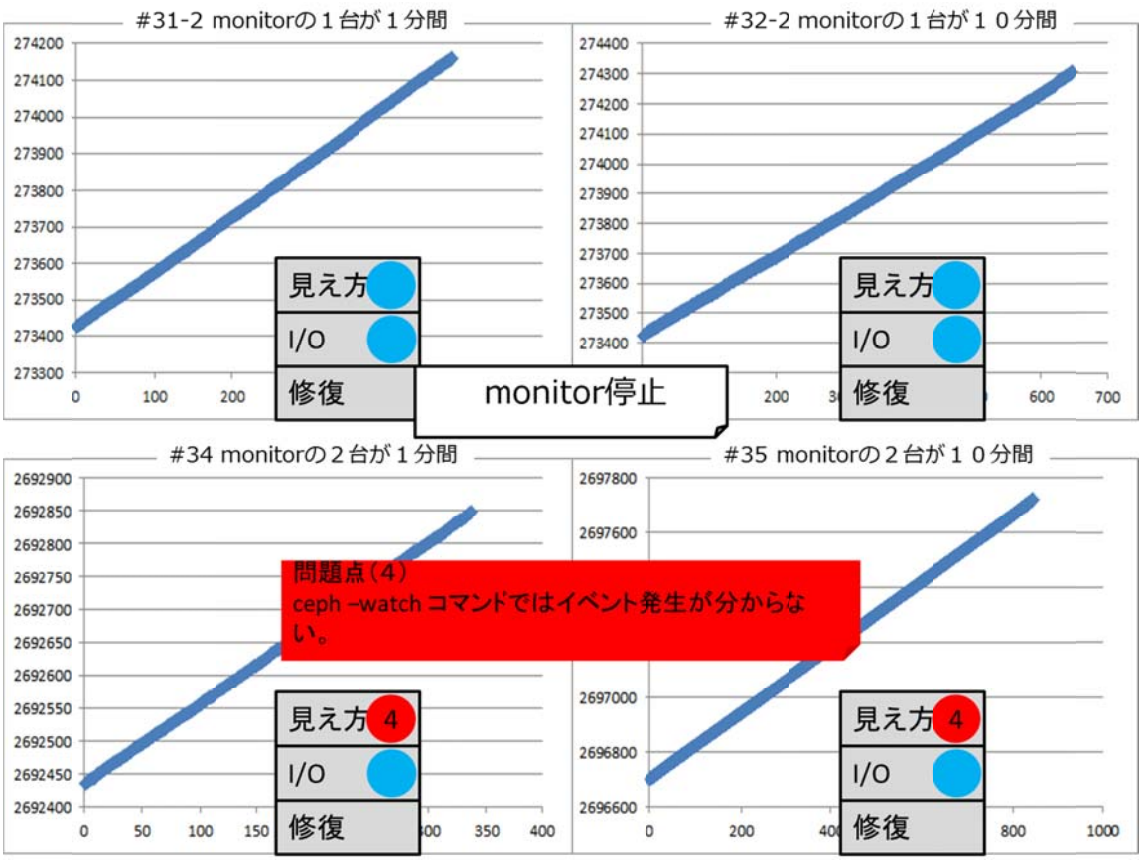


テスト項目 #31 から #33 では Ceph MON の 1 台でネットワーク途絶が発生している間も I/O が継続されていたことが分かります。ただし、テスト項目 #32 では、複数の RBD クライアントのうちの 1 台で約 90 秒間 I/O がブロックされていました。この時の Ceph ログには、4 台の Ceph OSD のうちの 3 台が、残り 1 台に対する監視の通信の応答を待っているログが出力されていました。Ceph MON がクラスタマップを最新に維持するため、各 Ceph OSD はランダムな Ceph MON1 台に対して定期的に通信を行いますが、当該 Ceph MON でネットワーク途絶が発生した影響でこの通信が滞り、連鎖的に、当該 Ceph OSD に対するその他の Ceph OSD からの監視の通信の応答に時間が掛かっていた可能性が考えられます。Ceph クライアントからのオブジェクト書き出しの I/O 要求は、CRUSH アルゴリズムで決まるプライマリ OSD が受信し、プライマリ OSD が CRUSH アルゴリズムで決まるレプリカ OSD に I/O 要求を配布し、レプリカ OSD からの応答を回収してから Ceph クライアントに応答します。よって、プライマリ OSD かレプリカ OSD のいずれかで通信の応答に時間が掛かる状況では、 Ceph クライアントの I/O 要求の応答にも時間が掛かるものと思われます。



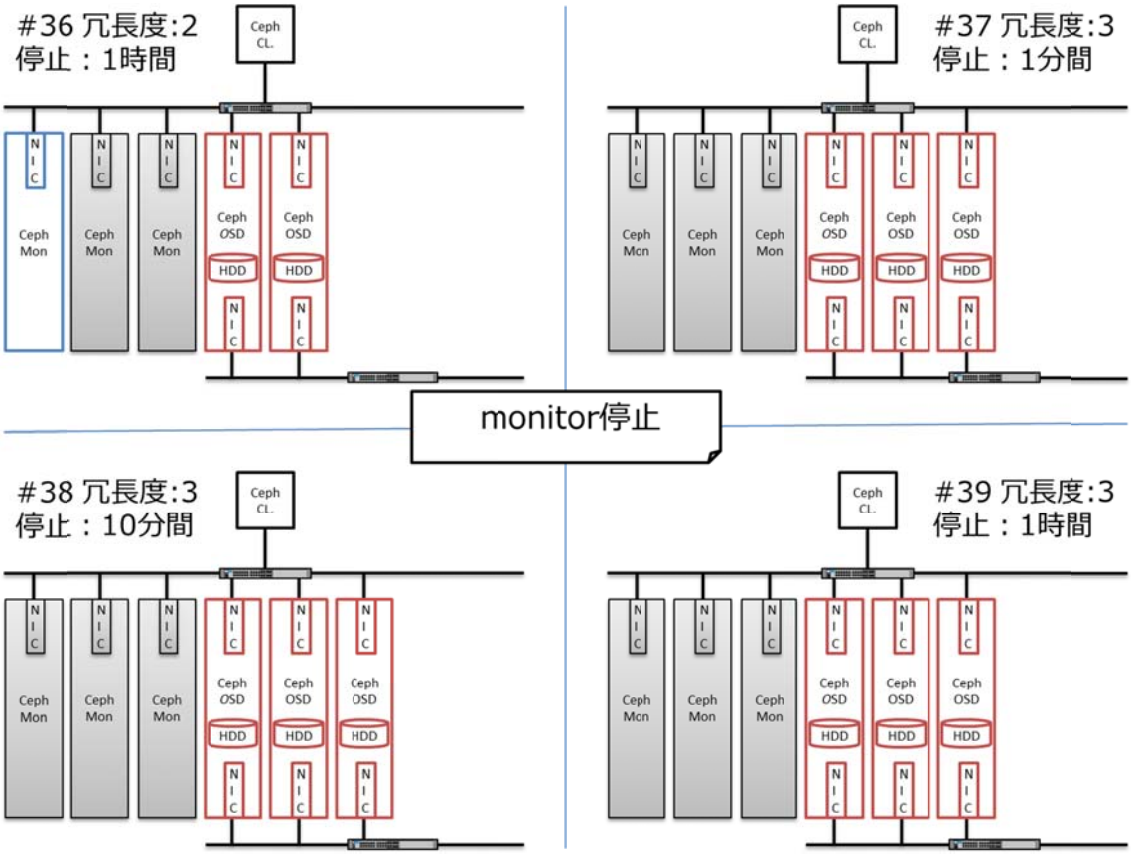
テスト項目# 31-2 から #35 です。

ここでは Ceph MON サーバーの 1 台または 2 台でダウンが発生しています。Ceph MON サーバーがダウンした状態が持続した時間が余白に記載されています。

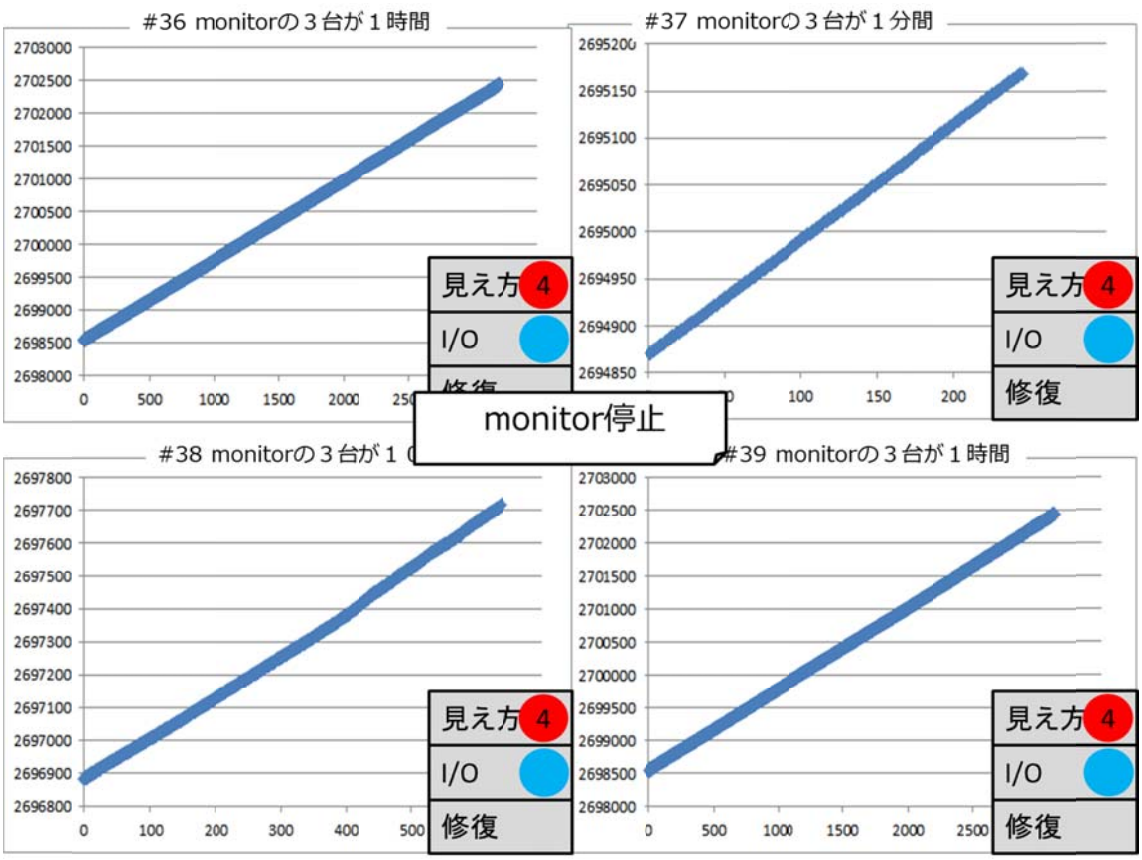


テスト項目 #32-2 から #35 では Ceph MON の 1 台または 2 台でダウンが発生している間も I/O が継続されていたことが分かります。ただし、テスト項目 #34 と #35 では、コマンド (ceph –watch) の出力から Ceph MON がダウンしていることが判別できませんでした。具体的には、2 台の Ceph MON がダウンしたにも関わらず、コマンド (ceph –watch) の出力上では、1 台の Ceph MON が停止し HELTH_WARN に状態遷移し、当該 Ceph MON 再起動後に HEALTH_OK に状態したように見え、もう 1 台の Ceph MON が停止したことが分かりませんでした。

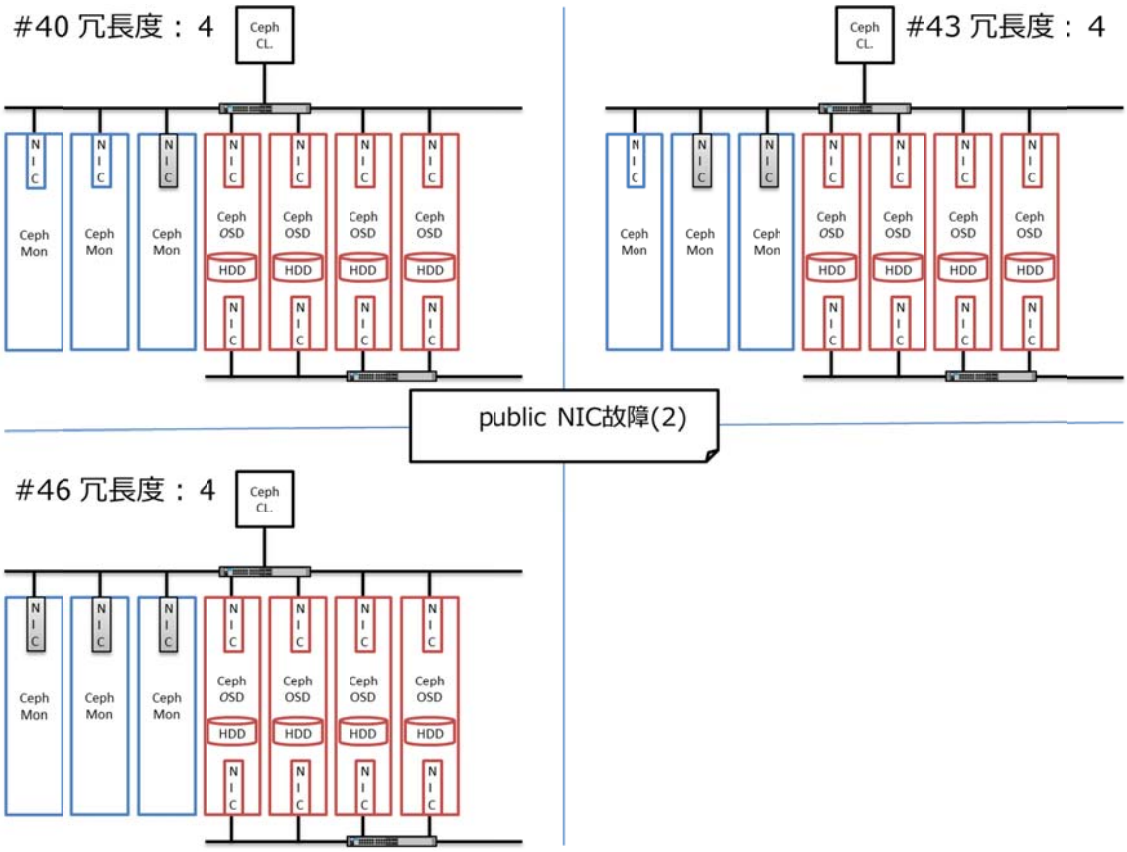
Ceph MON クラスタは奇数台数で構成され、その動作には多数派を形成できる台数 (＝定足数) が健全である必要があります。テスト項目 #34 と #35 では、3 台で構成された Ceph MON クラスタのうち 2 台がダウンしてしまうため、健全な Ceph MON の台数＝1 が定足数＝2 に満たない状況となります。また、コマンド (ceph –watch) の動作は Ceph MON クラスタが動作していることに依存します。よって、Ceph MON クラスタの定足数がない状態で発生した、もう 1 台の Ceph MON が停止、再起動の事象を、コマンド (ceph –watch) が正常に動作せず検知できなかったものと思われます。



Ceph MON サーバーダウンの項目がつづきます。テスト項目 #36 から #39 です。
ここでは Ceph MON サーバーの 2 台または 3 台全てでダウンが発生しています。いずれのテスト項目も Ceph MON クラスタが定足数に満たなくなり動作しなくなる状況です。

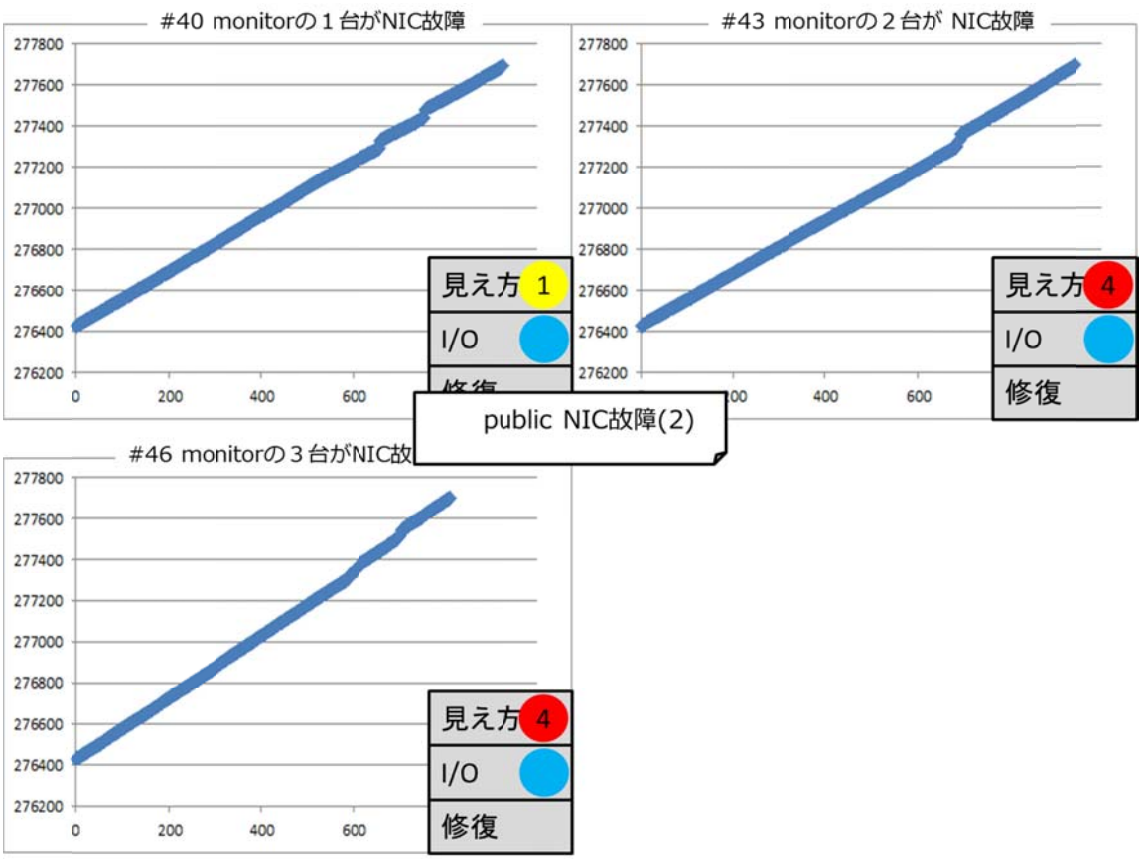


テスト項目 #32-2 から #35 と同様に #36 から #39 についても Ceph MON の 2 台または 3 台全てでダウンが発生している間も I/O が継続されていたことが分かります。いずれのテスト項目も、Ceph MON クラスタが定足数に満たない状況のため、コマンド (ceph –watch) の出力から Ceph MON がダウンしていることが判別できません。



テスト項目 #40 から #46 です。

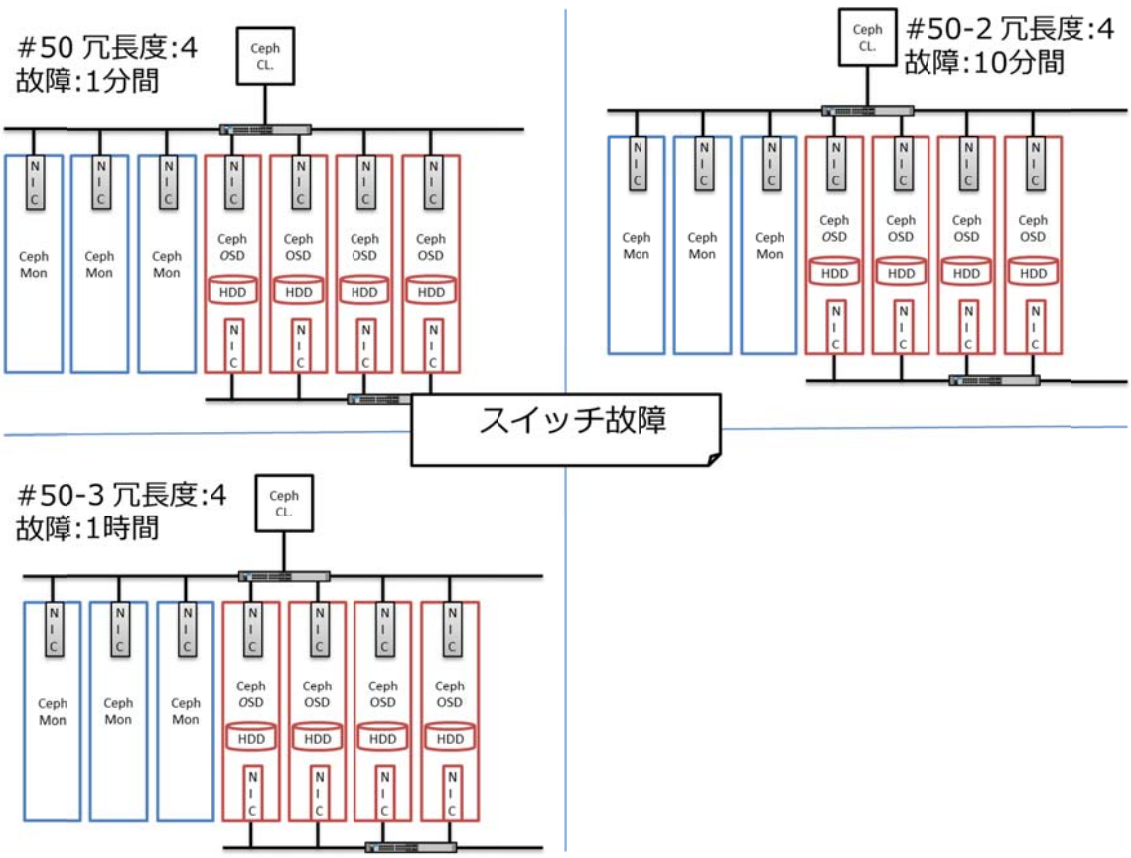
ここでは Ceph MON サーバーの 1 台から 3 台全てで NIC 故障が発生しています。各サーバーは、public network のネットワークにのみ接続されており、cluster network のネットワークには接続されていません。テスト項目 #43 と #44 は Ceph MON クラスタが定足数に満たなくなり動作しなくなる状況です。



Ceph MON がダウンする項目と同様に、Ceph MON の 2 台または 3 台全てで NIC 故障が発生している間も I/O が継続されていたことが分かります。テスト項目 #40 はコマンド (ceph –watch) 出力上、Ceph MON の NIC 故障は Ceph MON のダウンとして見えています。また、テスト項目 #43 と #46 は Ceph MON クラスタが定足数に満たない状況のため、コマンド (ceph –watch) の出力から Ceph MON の NIC が故障していることが判別できません。

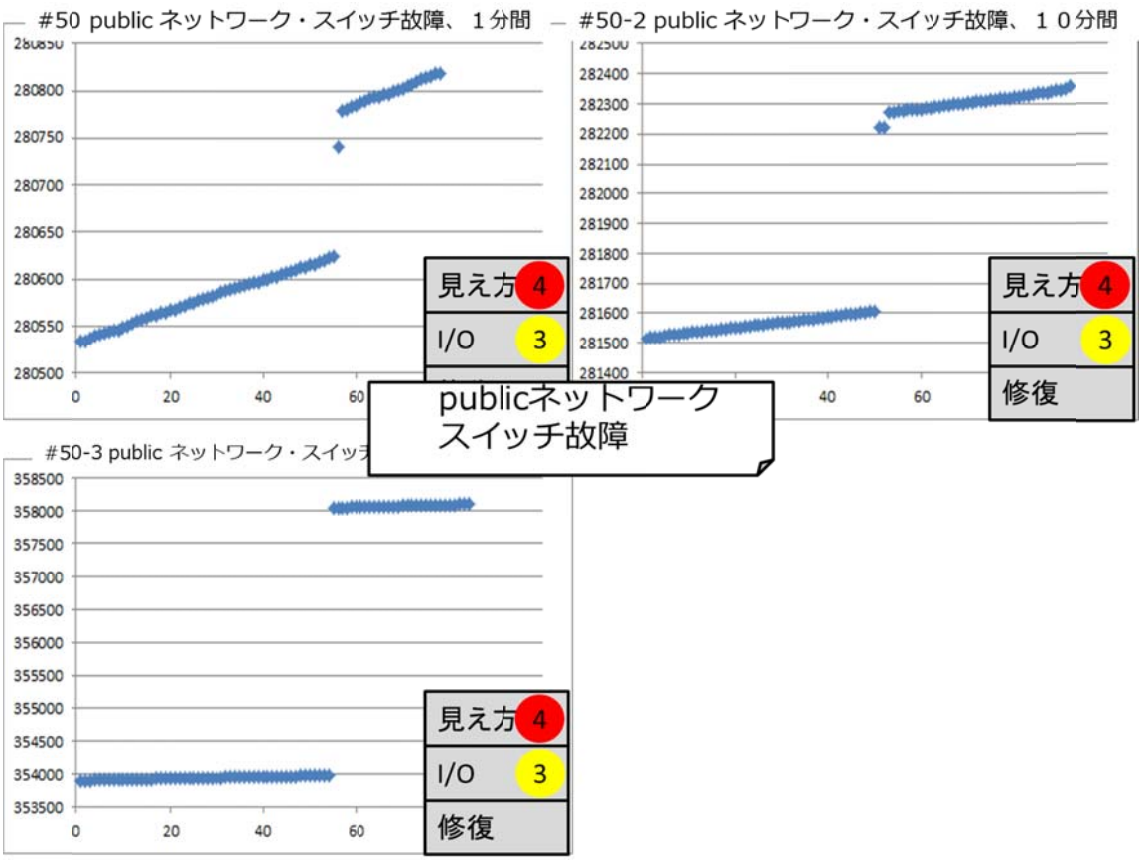
Ceph MON の NIC 故障の復旧方法としては Ceph MON を交換する (当該 Ceph MON を削除し、新規 Ceph MON を追加する) 方法と、故障した NIC のみ交換する (当該 Ceph MON をシャットダウンし、NIC を交換後再起動する) 方法がありますが、前者については Ceph MON を削除するオペレーションが Ceph MON クラスタが動作している、すなわち健全な Ceph MON 台数が定足数残っていることに依存する点に注意が必要です。

本稿では Ceph MON クラスタが動作しているテスト項目 #40 でのみ Ceph MON の交換による方法で対処し、Ceph MON クラスタが定足数に満たないテスト項目 #43 と #46 については NIC のみ交換する方法で対処しました。

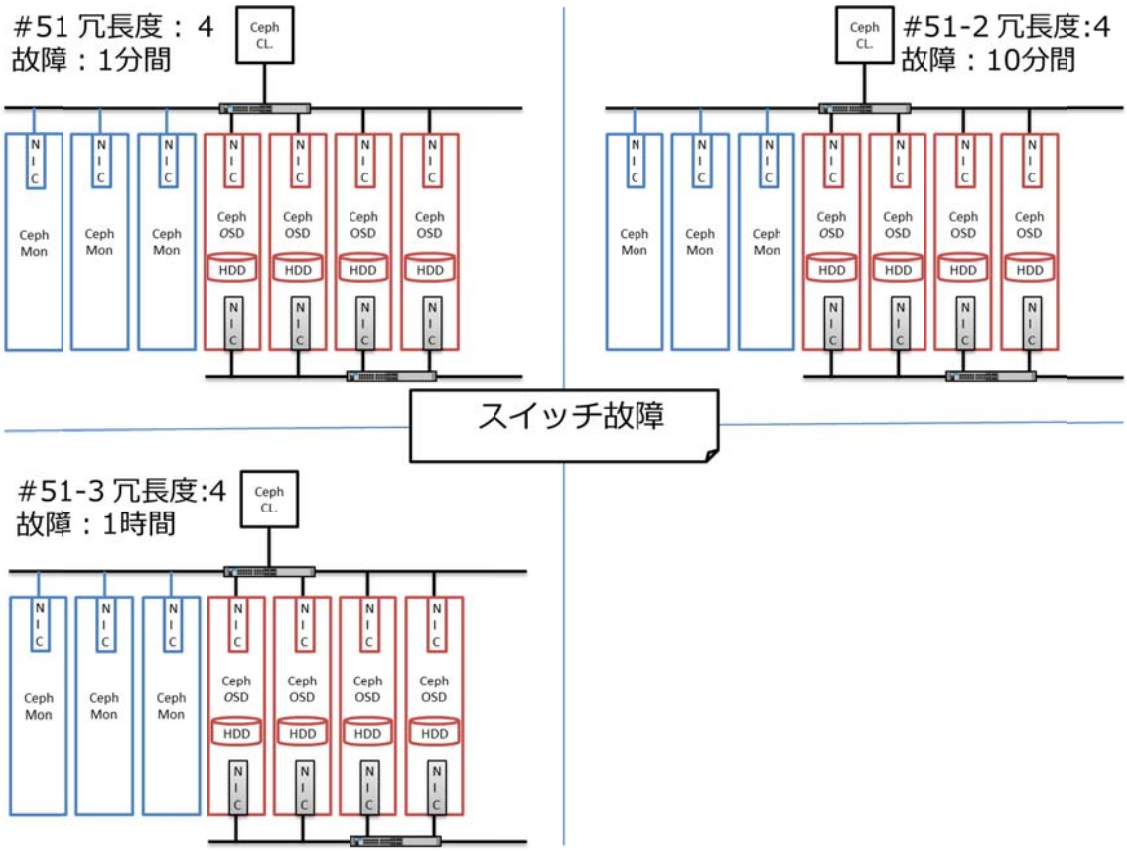


テスト項目 #50 から #50-3 です。

ここではネットワーク 2 系統のうち、public network のスイッチのダウンが発生しています。スイッチがダウンした状態が持続した時間が余白に記載されています。

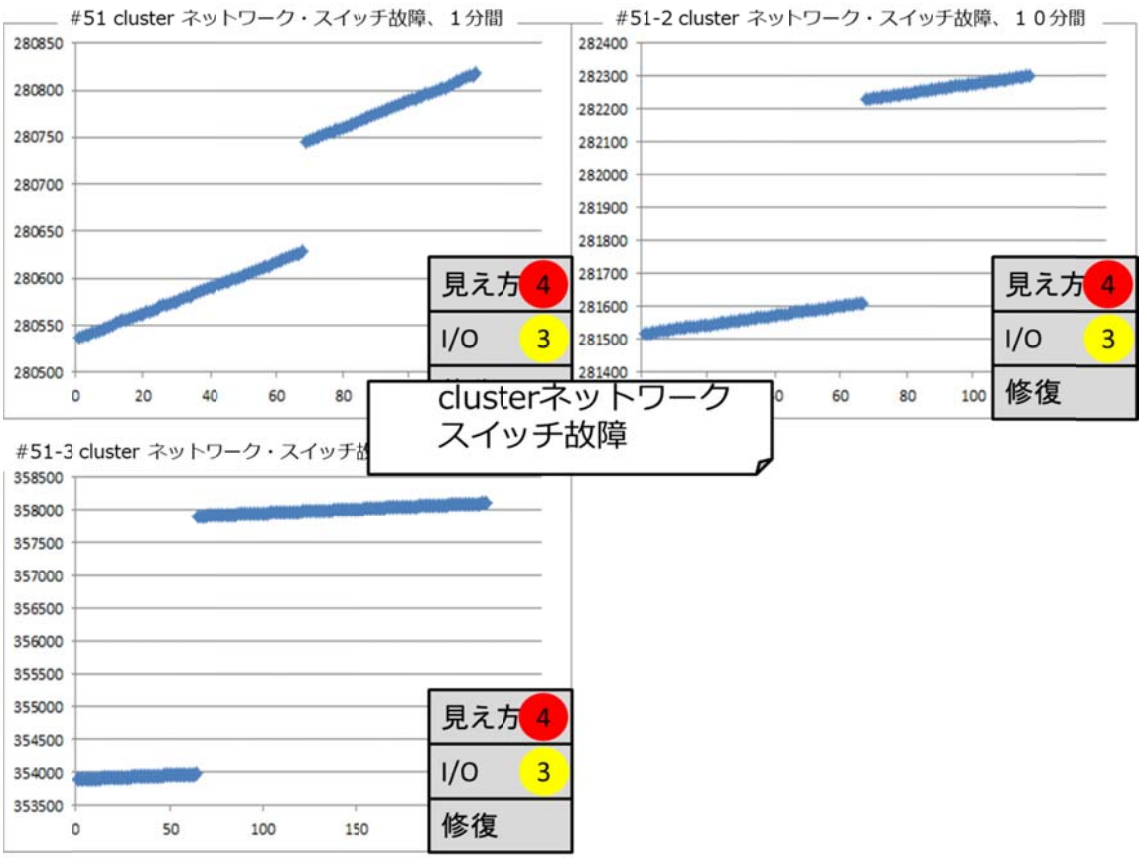


いずれのテスト項目も public network のスイッチがダウンしている間 I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、public network のスイッチが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。public network のダウンはコマンド (ceph –watch) と Ceph MON クラスタの通信もできないため、コマンド (ceph –watch) 出力から public network がダウンしたことを判別することはできません。



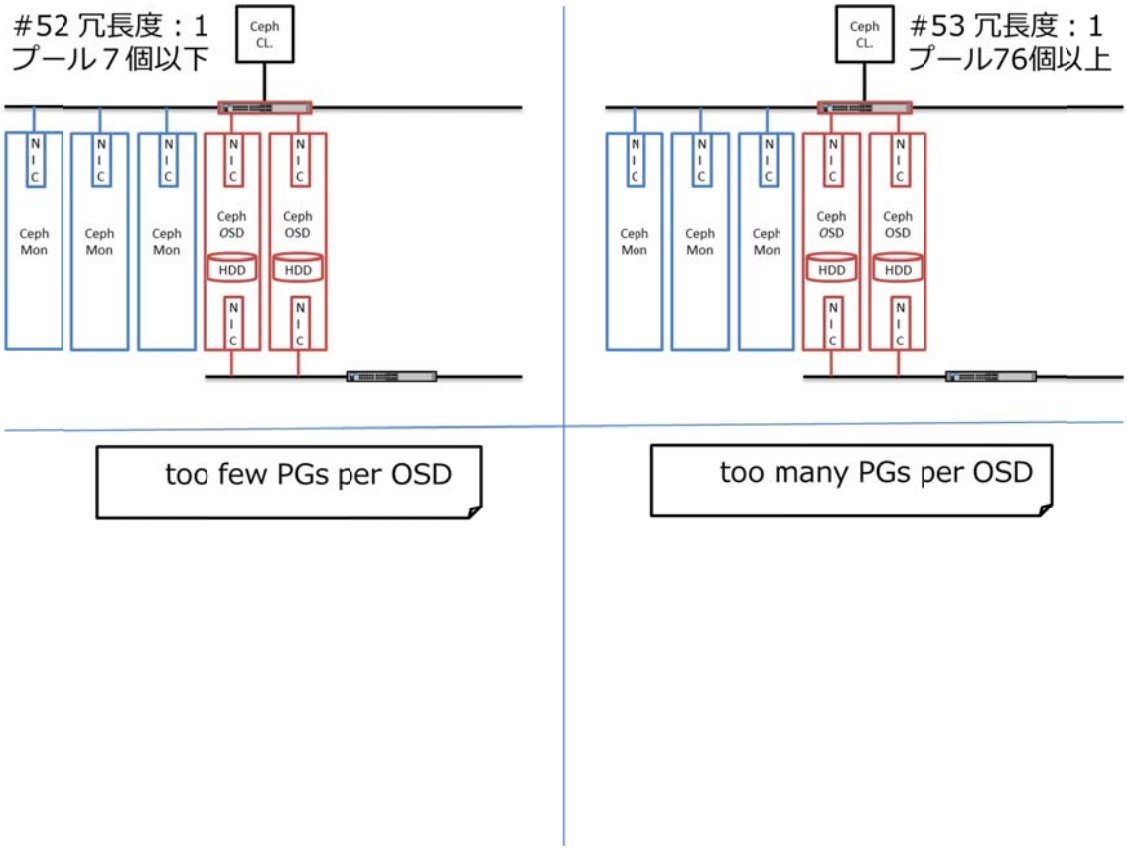
テスト項目 #51 から #51-3 です。

ここではネットワーク 2 系統のうち、cluster network のスイッチのダウンが発生しています。スイッチがダウンした状態が持続した時間が余白に記載されています。



いずれのテスト項目も cluster network のスイッチがダウンしている間 I/O がブロックされていたことが分かります。I/O プロセスはいずれも終了していないので、cluster network のスイッチが復旧した時点で I/O のブロックが解け、I/O エラーは発生していないことが分かります。cluster network は、Ceph OSD 間の通信に使用されます。Ceph クライアントからのオブジェクト書き出しの I/O 要求は、CRUSH アルゴリズムで決まるプライマリ OSD が受信し、プライマリ OSD が CRUSH アルゴリズムで決まるレプリカ OSD に I/O 要求を配布し、レプリカ OSD からの応答を回収してから Ceph クライアントに応答します。よって、プライマリ OSD とレプリカ OSD の間の通信ができない状況のため、Ceph クライアントの I/O 要求の応答がなかったものと思われます。

いずれのテスト項目も、コマンド (ceph –watch) の出力から cluster network のスイッチダウンを判別することはできませんでした。具体的には、各 Ceph OSD がダウンとダウンからの復旧を繰り返しているように見えていました。これは、cluster network がダウンしたことで、Ceph OSD 間の監視の通信ができなくなったことで、Ceph OSD 同士が互いを Ceph OSD ダウンとしたと判定しこれを Ceph MON に報告する一方で、public network はダウンしていないので、各 Ceph OSD はランダムな Ceph MON 1 台に対して定期的な通信を継続するため、Ceph MON がこれをダウンした Ceph MON の復旧あるいは先のダウン報告の誤報と判定しているものと思われます。

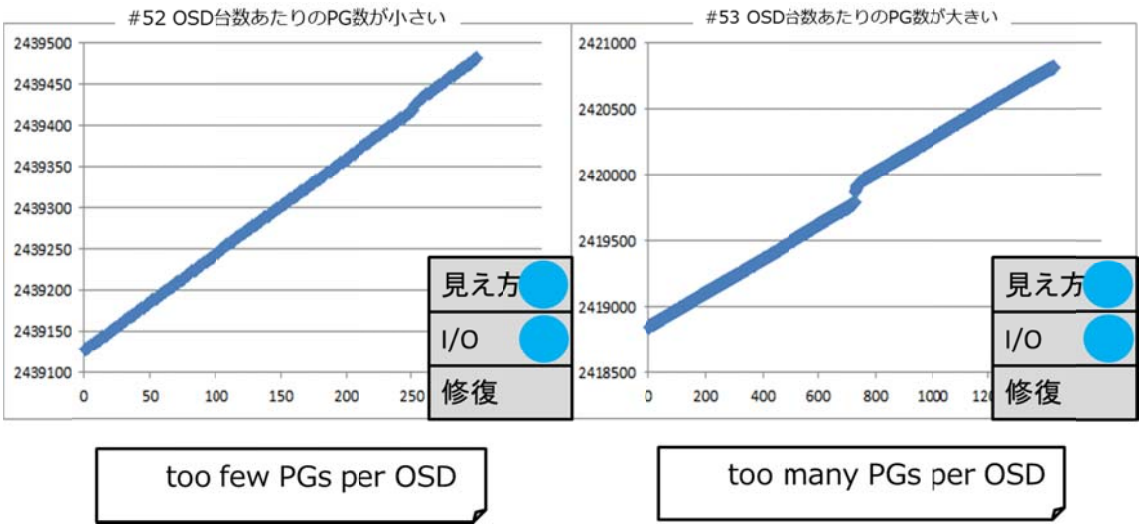


最後の項目は Ceph OSD 台数あたりの PG 数が少なすぎる、あるいは多すぎる場合に関する項目です。テスト項目 #52 は PG 数が少なすぎる場合で、テスト項目 #53 は PG 数が多すぎる場合です。

本稿の構成では FileStore と呼ばれるストレージバックエンドが使用されているので、オブジェクトは Ceph OSD 上のファイルです。可用性のため、オブジェクトのレプリカが複数の Ceph OSD 上に置かれます。あるオブジェクトのレプリカを置く Ceph OSD セットのパターンを Placement Group (PG) といいます。PG はプールを作成すると Ceph OSD 数に応じた数が自動的に定義されます。オブジェクトと Ceph OSD のマッピングは直接的ではなく間接的に行われます。オブジェクトは一旦 PG にマッピングされ、PG が Ceph OSD にマッピングされます。プールは、1 つの Ceph ストレージ・クラスタを複数の用途で使用する場合の、用途毎の論理的なパーティションで、データ冗長度等はプール毎に設定する属性です。よって、プールの数は運用設計に依存します。

PG 数の調節はプール数とプール当たりの PG 数のパラメータで可能です。オブジェクトの I/O およびデータ冗長度の維持は PG 毎に行われるので、Ceph OSD 数に応じた PG 数を設定します。Ceph OSD 数に対して現在の PG 数が多すぎたり少なすぎたりすることは性能上好ましくないため、いずれの場合も Ceph クラスタの状態が HEALTH_OK から HEALTH_WARN に遷移し、ワーニングメッセージが出力されます。PG 数は予め設計・設定するものなので、通常の運用においてこれらのワーニングが出力される状況はプール数あるいは Ceph OSD 数の増減があった状況等が考えられます。

本稿では性能上の観点は設定していないので、いずれのテスト項目においても I/O が継続されていることのみを確認します。



いずれのテスト項目もワーニングメッセージが出力されている間も I/O が継続されていたことが分かります。テスト項目 #53 で I/O の進捗が滞っている箇所が見られますが、これは Ceph OSD の 1 台で意図しないディスク故障が発生し、Ceph OSD デーモンが EIO エラーにより再起動を繰り返していたためで、本テスト項目との関係はありません。当該 Ceph OSD の対処を行うことで現象が収束することを確認しています。

OpenStack/Ceph 異常系テスト (1) の具体的なテスト項目 (想定されるハードウェア障害) とその結果は以上です。

今回設定したシステムの構成とハードウェア障害のパターンの範囲において、アプリケーションの I/O への影響という観点に関しては、総じて非常に分かりやすい結果が出たと思われます。すなわち、ハードウェアの故障によってアプリケーションの I/O がブロックされる場合があるが、ハードウェアの故障を取り除くことによって I/O のブロックは解ける、という見え方です。
一方で運用上の問題も検出されました。すなわち、コマンド (ceph -watch) 出力でハードウェア障害のイベント発生が分からないことや、コマンド (ceph -watch) 出力と実際のイベントが一致しないことがあるという点です。
フィールドサポートの場面としては障害箇所の特定のためのオペレーション (コマンド実行等) をタイムリーに実施することが困難な場合も想定されます。このような場合に対しては取得済みの Ceph のログからの障害箇所の特定が必要と思われます。
上記問題のフィードバックとして、この後に述べる OpenStack/Ceph 異常系テスト (2) においては、対処方法の情報として、ログだけからハード障害を切り分けできるかどうかという観点を加えています。そのために、テスト (2) の実施に先立ってまず現状の Ceph ログの問題点と対処案を検討します。

※本書掲載内容の複写･無断転載を禁じます。

- 本書は 2017 年 8 月現在の情報に基づいて作成しております。
- VA Linux Systems Japan、VA Linux および VA Linux ロゴは VA Linux Systems Japan の商標または登録商標です。
- Linux は Linus Torvalds の米国およびその他の国における登録商標または商標です。
- その他、記載されている企業名、ブランド名および製品名は、各企業の商標または登録商標です。

VA Linux Systems Japan 株式会社
お問合せ: sales@valinux.co.jp
http://www.valinux.co.jp/