～ HA8000 クラスタシステム ～

ファイルシステムアンケート結果報告およびファイルシステム増強計画

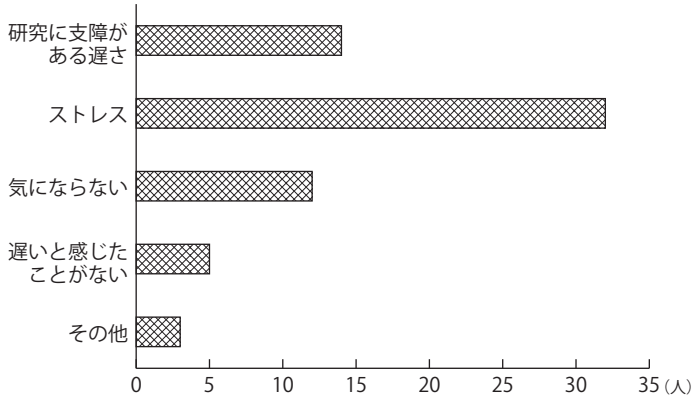
スーパーコンピューティング部門

2009 年 2 月 10 日に HA8000 クラスタシステム利用者全員にファイルシステムに関するアンケートへのご協力をお願いしました。20 日までに 89 件の回答を頂き、同一利用者による再回答などの重複分を除いた結果、有効回答は 71 件となりました。ご協力ありがとうございました。本稿ではアンケート結果およびファイルシステム増強計画の概要を発表します。

１．アンケート結果

Q1. ログインノードのレスポンスについて

HA8000 クラスタシステムのファイルシステムはファイル操作や細かいファイルの I/O を苦手とします。そのため、ログインノードにおいては ls コマンド、圧縮ファイルの展開、または多数のファイルからなるアプリケーションのコンパイルなどに時間を要しています。このことについてどうお感じですか？



「研究に支障がある遅さである」を選択された方のご意見

- 3rd party のライブラリをコンパイルするのに時間が掛かりすぎる。また、そのようなライブラリのバージョンアップに追随できない。
- ファイル数が数十を超えると、たとえば ls の結果が戻るのに数十秒もかかりとても困る。
- 限られた時間内での作業となるのでレスポンスは必要。HA8000 システムの利用促進の上では障害要因になる。
- 同一ディレクトリ上の 100 程度のファイルが存在すると ls コマンドがかえってくるまで 5 ～ 10 秒ほどかかるため、少し不便さを感じる。

「研究自体に大きな支障はないが、作業にストレスがたまる」を選択された方のご意見

- 解析結果が大きな記憶容量を必要とするためディスク容量を知りたいとき、特に遅さを感じる。
- 「研究に支障がある遅さである」と答えてもいいかもしれないが，時間帯によっては何のストレスもないときもあるのでこの選択肢にした。
- 込み合う時間帯ではストレスを感じる。
- ls などの結果表示までのレスポンスが悪い。
- 運用開始時にくらべると状況が改善している。
- 他の共同利用計算機を利用した事があるが、東大が一番遅いと感じる。
- home ディレクトリにバッチジョブの出力が溜まるので、放置していると処理が重くなる。
- ファイル削除の遅さにたまにフラストレーションを感じる。

「遅いが特に気にならない」を選択された方のご意見

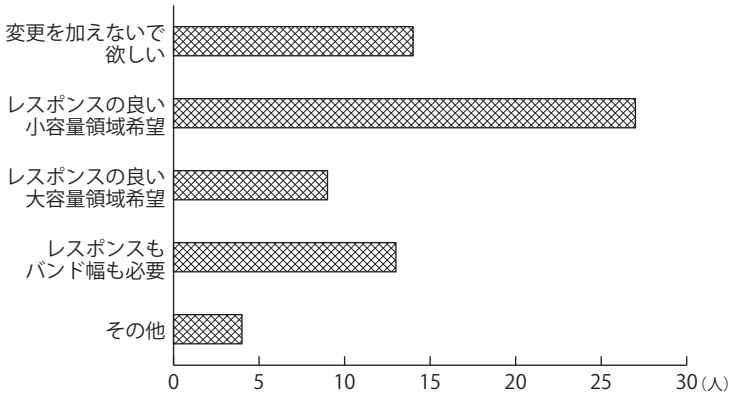
- 昔の SR 等に比べると速くて驚いた。

「その他」を選択された方のご意見

- レスポンスがよいときは ls コマンドやコンパイルに関してストレスはないが、一旦レスポンスが悪くなると非常に時間がかかるのでストレスがたまる。
- シェルスクリプトで入力ファイルを大量に作成しているが、非常に遅いと感じる。 22 のファイル（334 bytes から 2793bytes）を作成するのに手元の Intel Xeon3060 @ 2.40GHz にて経過時間 9 秒で作成できるが、HA8000 のログインノードでは 1 分かかっている。 また、チューニングのため、コンパイルオプションやコンパイラを変更して Gaussian/GAMESS/ABINIT-MP をコンパイルしているが、手元の計算機で 1 時間ほどのコンパイル時間が必要なる場合、HA8000 のログインノードでは一度のコンパイルに半日から 1 日かかることがある。特に日立のコンパイラが gfortran や intel コンパイラに比べて数倍も遅い。

Q2. 全体的なレスポンス改善希望について

ファイルシステムのレスポンス改善のための方法として希望に近いものをお選び下さい。
下の選択肢ほど改修のための停止期間が長くなり、システムが不安定になるリスクが高くなります。許容できる一番上のものをお選び下さい。



「今のままでよいのでシステムに変更を加えないで欲しい」を選択された方のご意見

- プログラム側での対処でとりあえず現状のレスポンスで問題無い。可能であれば改善を望みたいが、システムの不安定化や長期停止は避けたい。
- 現状でも /home の並列 IO の性能は高いと感じたが、更に性能が上がればなお良い。

「現在の /home の他に、プログラムの作成や小さな入出力データを保存するためのレスポンスの良い領域が欲しい。10GB から 20GB 程度で十分である」を選択された方の自由記述欄でのご意見はありませんでした。

「現在の /home の他に、大容量のレスポンスの良い領域が欲しい。ファイルの作成や削除などのファイル操作の性能が重要であり、入出力バンド幅はそれほど重視しない」を選択された方のご意見

- コンパイルオプションを変えた Gaussian/GAMESS/ABINIT-MP の実行モジュール群の作成と保存で、1 ケースあたり 100MB 程度を必要とする。これらの実行モジュールを複数保存するのでそれだけで 1GB 程度必要とする。これに計算結果の解析をログインノード行うには 10GB から 20GB では容量が足りない。

「現在の /home の他に、レスポンスの良さと高い入出力バンド幅を兼ね備えた領域が欲しい」を選択された方の自由記述欄でのご意見はありませんでした。

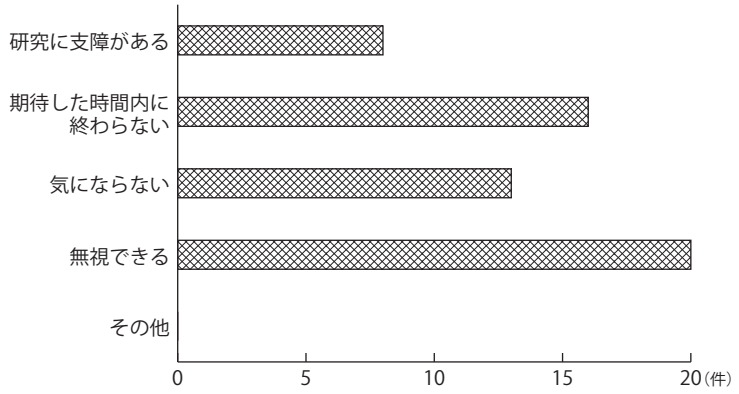
「その他」を選択された方のご意見

- /home は小容量低レイテンシとして ~/.ssh/authorized_keys や ~/.bashrc 等のファイルが迅速に読み込まれるようにして欲しい。また、3rd party のソフトウェアやライブラリが迅速にコンパイルできることが必要。それとは別に大容量・高レイテンシ・中バンド幅のファイルシステムがあれば良い。
- 何がレスポンスに影響しているか分からないので選択のしようがないが、改善して欲しい。

※ Q3 および Q4 はアプリケーション名および使用ノード数をおききました。統計的な影響はないので省略します。

Q5. ファイルシステムの性能について

お使いのアプリケーションのファイル入出力に対して現在の HA8000 クラスタシステムのファイル性能は十分でしょうか。



「研究に支障がある遅さである」を選択された方のご意見

- 125GB を超えるスクラッチファイルを利用しようとすると /tmp ではなく /short を使わざるを得ず、/tmp の性能であれば 48 時間の経過時間制限値内で計算が終わるが、/short では終わらない。
- メタデータアクセスが遅い。ほとんどのソフトウェアのデフォルトとして使われている 4KB 単位の write が非常に遅い。

「研究自体に大きな支障はないが、ファイルアクセスが遅いために期待した時間内に終わらない」を選択された方のご意見

- 可視化のジョブでは計算は一瞬でほとんどがファイル読み込みの時間となる。そのため並列 IO は速ければ速いほどよい。
- 同一ファイルを複数ノードから同時に読み込む場合でも低速なのでアプリケーションの改修が必要になった。
- モデル内時間で一時間ごとにヒストリ出力していると、計算時間 (12 分 / 日) よりもファイル I/O にかかる時間 (24 分 / 日) のほうが二倍程度長い。I/O 性能的には SGI Altix で同一モデルを走らせたときの 1/2 から 2/3 程度しか出ていない。

「遅いが特に気にならない」を選択された方のご意見

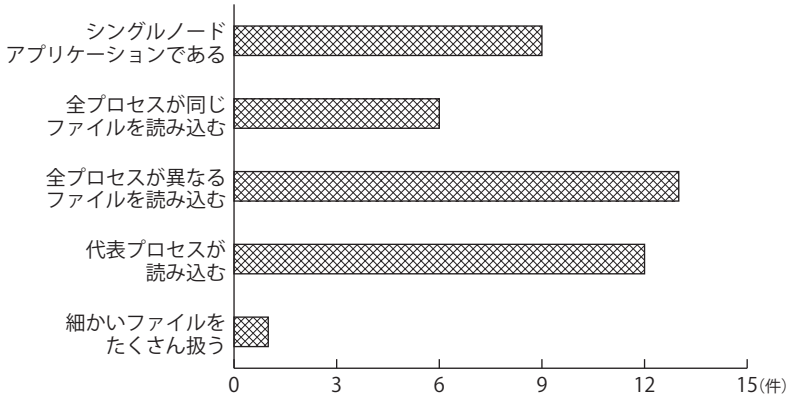
- ファイルアクセスは速い方がよいが、バッチ利用ではその時間も含めて考えるので大きな支障はない。
- 48 時間でシミュレーションがおわらないこともたたあるので、メモリー上の情報をすべて書き出し、次の実行時にそれを読むといった比較的大容量の入出力をしているが、特に支障はない。

- 結果ファイルの書き出しが非常に遅い。自分のマシンの nfs マウントの方が何倍も速い。
- 共有ディスクを使用した際には、想定時間内に処理を完了することが不可能であったが、ローカルディスクを使用し、処理の前後でデータの配置、回収を行うように改良したことで、想定時間内に処理を完了することが可能となった。

「計算時間に比べファイル入出力は非常に短いのでファイルシステム性能は気にならない」と回答された方の自由記述欄でのご意見はありませんでした。

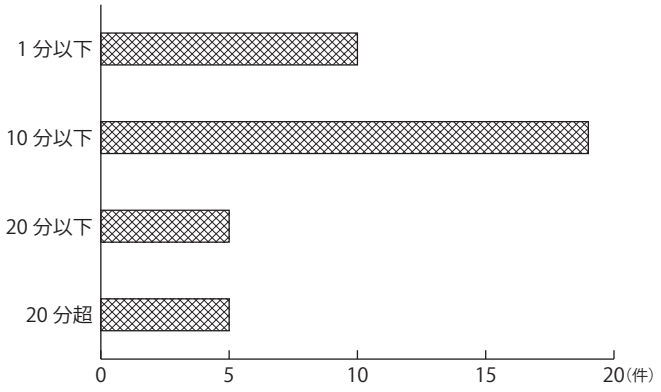
Q6. 読み込み量とアクセス方式

アプリケーションが一回の実行に中に行うファイル読み込みについて、量とアクセスパターンをお答えください。複数のファイルを読み込む場合は全体性能への影響が最も大きいものについてお答えください。



Q7. ファイル入力の許容時間

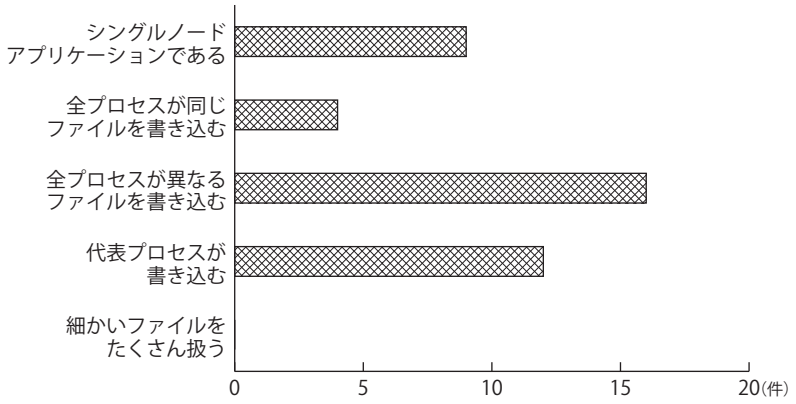
Q6 の入力に要する時間の合計として何分まで許容できますか



※ Q6 でお聞きしたファイル読み込み量と Q7 でお聞きした許容時間から計算した要求バンド幅は 1MB/s 程度から 2GB/s 程度と利用者によって 1000 倍以上の差がありました。

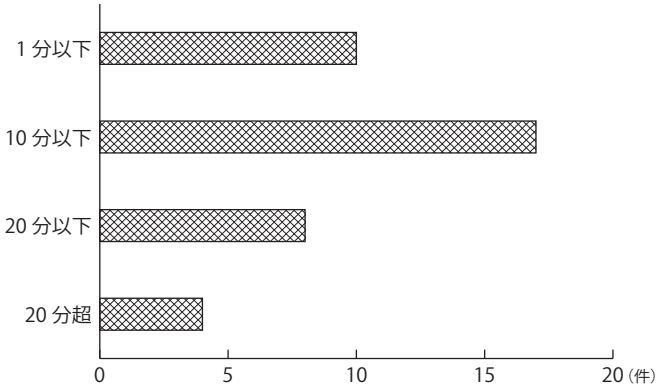
Q8. 書き込み量とアクセス方式

アプリケーションが一回の実行中に行うファイル書き込みについて、量とアクセスパターンをお答えください。複数のファイルを書き込む場合は全体性能への影響が最も大きいものについてお答えください。



Q9. ファイル出力の許容時間

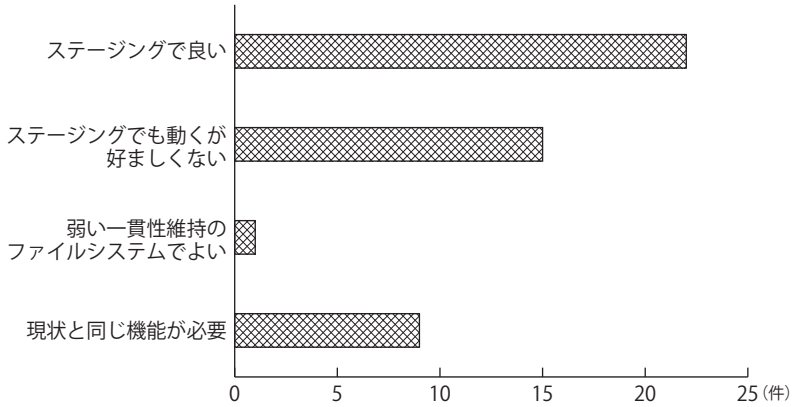
Q8 の出力に要する時間の合計として何分まで許容できますか



※ 読み込みと同じく要求バンド幅は利用者によって大きく異なる結果となりました。

Q10. 共有ファイルシステムの必要性について

現在 /home, /short は常に全ノードから同じファイル、同じ内容が見えています。このような機能は必要ですか。



「ステージング機能があればファイルサーバに直接アクセスできる必要はない」を選択された方のご意見

- ステージング機能があったとしても現在のファイルシステムの性能では実用的ではない。
- ステージングがあればファイルサーバに直接アクセスできる必要は無いが、入力・出力ファイルが大きいため、16 コアで 100GB しかない /tmp では容量が足りない。安心して使うのであれば 1 コアあたり 100GB 欲しい。

「ステージングでも実行不可能でないが、使用するファイルが多くステージングの記述が煩雑であるため、ファイルサーバのデータに直接アクセスできることが望ましい」を選択された方のご意見

- ステージングを採用する場合、実行状態の監視ができなくなるので、デバッグの手間が増える。使用するファイルは多くはないがステージングの記述を間違えるとジョブが無駄になるのでその分バッチスクリプトの難易度が高い。
- あまり複雑な操作はやりたくない。
- 既に自前でステージングを実施しているが、今後プログラムが改訂され入出力データが変更になる度に設定変更等が生じるのは煩雑なため、可能であれば高速な共有ファイルシステムがあることが望ましい。

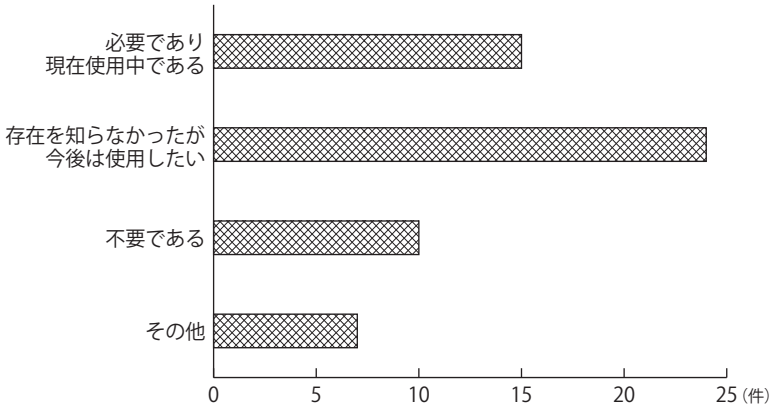
「アクセスするファイルが実行前には決定しないのでステージングは不可能であり、ファイルサーバに直接アクセスできることは必須である。ただしプロセス間の読み書きのタイミングには依存関係はなく、書き込みが即座に他のノードでの読み込みに反映される必要はない」を選択された方の自由記述欄におけるご意見はありませんでした。

「あるプロセスが書いたファイルを他のプロセスが読み込むため、共有ファイルシステムは必須であり、書き込みは即座に全ノードに反映される必要がある」を選択された方のご意見

- 現状はかなりよいシステムだと考える。変えないで欲しい。

Q.11 一定時間で消える領域の必要性について

/short は 5 日経つと自動的にファイルが削除される代わりに事実上容量無制限のファイルシステムとなっています。このような領域は必要ですか？



「必要であり、現在使用中である」を選択された方のご意見

- 使用してはいるが、読み書きが遅すぎる。
- ステージングが出来れば本来不要であるが、コア（プロセス）あたりの /tmp 容量が小さすぎるために結果として必要になっている。
- /tmp の容量が小さすぎるためにステージング用に用いている。ファイルをステージングする別のスキームが導入される場合には /short は不要になる。
- 許される範囲で長期に利用したい。
- 現在のジョブ待ち時間を考えるとジョブ実行前にリスタートファイル等が消えることがあり、5 日間は短すぎる。

「存在を知らなかったが今後は使用したい」を選択された方のご意見

- 一時利用ファイルを /tmp に出力し、定期的に運用で容量の確認を行うことを考えているが、/short を使えるのであれば手間を削減できる。アクセス速度に問題が無いことを確認した上で利用を検討したい。

「不要である」を選択された方の自由記述欄におけるご意見はありませんでした。

「その他」を選択された方のご意見

- 現段階では不要であるが、さらに大規模な計算を行う場合に使用する可能性がある。
- 速いのなら使えるかもしれないが、HSFS (/home と同じ速度) では使い物にならない。 一般に、同じ性能ならばこのような領域をあえて用意する必要性は感じない。

2．今後の増強計画について

2.1 アンケート結果のまとめ

アンケートの結果、次のような利用者の皆様の意見、システム利用状況が明らかとなりました。

- 75% の利用者がログインノードのレスポンスに不満をお持ちである
- ファイルシステムの遅さが 42% のアプリケーションに無視できない影響を与えている
- ファイル量に依らず、多くの利用者はファイル I/O が 10 分以内に終わることを期待されている
- ファイルアクセスパターンは多種多様である
- ステージングで動作するアプリケーションが 78% に上るが、使い勝手の観点から好ましくないと回答された利用者も多い

2.2 今後の方針

アンケート結果をふまえまして、次のような増強計画を検討しております。検討段階ですので必ずしも実現するものでないことをご了承願います。

高レスポンス領域の新設（2009 年 4 月提供開始目標）

ログインノードのレスポンスを改善するため、NFS サーバを準備し、利用者全員に 10GB から 20GB 程度の領域を追加負担金なしでご提供することを検討しております。また希望される利用者にはホームディレクトリをこの NFS 領域に設定変更することも検討します。NFS はレスポンスの点では HSFS より快適です。

一方、NFS はサーバを一台しか配置できないため大量入出力には不向きです。したがいまして、大規模ジョブが誤って NFS サーバに大量のリクエストを発行すると NFS は非常に重くなり、多くの利用者に影響します。NFS 領域提供の際には各利用者にはこの特性を十分理解して使用していただくようお願いいたします。

大容量 NFS 領域の提供（2009 年 4 月提供開始目標）

レスポンスが重要で、かつ大量のデータ入出力が必要な利用者のため、申込制で大容量 NFS 領域を提供することを検討しております。本 NFS 領域をご使用の場合も HSFS 領域は引き続きご利用いただけますが、NFS 領域と HSFS 領域との合計でお申し込みのディスク容量となるよう制限値を設定させていただきます。配分は申込時にご希望をお聞きしますが、提供できる NFS 領域の総量に制限がありますのでご希望に添えない場合もございます。総量の増加が伴わない場合は追加の負担金はいただきません。

NFS サーバは前項で紹介した全利用者に提供するものとは別のサーバ、別のディスクを用います。この領域には大量の入出力をしていただいても結構ですが、他の利用者が同じ時間帯に同じように高い負荷をかけている場合はレスポンスが悪くなる可能性もあります。また、あまりにも負荷が高い場合はサーバダウンに至る可能性もありますので、試験サービスなどを通じて状況を確認しながら段階的な導入を行う予定です。

ネットワーク増強（2009 年 8 月導入目標）

現在、計算ノードからファイルサーバにアクセスできるネットワークはノード間通信に利用している高速な Myrinet とは独立となっており、16 ノードにつき 2Gbps のバンド幅となっております。このため、現状では 16 ノードジョブ全体からファイルアクセスを行っても、理論上の限界が 2Gbps(250MB/s) となります。例えば各ノードの使用可能メモリ量である 28GB を上限まで使用しているアプリケーションが、そのデータをすべてディスクに書き出す場合、16 ノードでは 448GB のデータとなります。これを多くの利用者が希望されている「10 分以内」で書き出すためには 448GB / 600 秒 = 746MB/s の性能が必要です。つまり現在のネットワーク構成ではファイルシステムをどれだけ増強してもネットワーク性能の限界で十分な性能は得られません。そこでファイルアクセスの通信も Myrinet を経由するようにネットワークを再構成し、各ノードから 10Gbps でファイルサーバと通信できるようなネットワーク構成とすることを検討しています。これが実現すれば 16 ノードにつき 2Gbps という制限がなくなりますので、ファイルアクセスの性能が向上することが期待できます。

レスポンスの良い新たな並列ファイルシステムの導入（2009 年 10 月提供開始目標）

レスポンスの良い共有ファイルシステムを必要とする利用者の方のために、大量ノードからの入出力にも耐え、かつ細かなファイル操作も NFS と同程度の性能で処理できる新たな並列ファイルシステムの導入を検討します。このようなファイルシステムが導入でき、前項のネットワーク増強が完了すればファイルシステムに関する問題はほぼすべて解決できるものと期待しております。

しかしながら新たなファイルシステムを導入する際はソフトウェアの欠陥によるシステムダウンの可能性など、システムの安定性に関わるリスクを十分に検討する必要があります。現在のファイルシステム性能でも（ストレスはあるものの）研究を阻害するほどではないと回答された利用者が半数以上であったことも十分考慮し、安定性等を含めた現在の総合的なサービス品質は堅持すべく慎重な導入を行います。

３．おわりに

ファイルシステム性能に関しましては多くの利用者の方にご迷惑をおかけしておりますことをお詫び申し上げます。情報基盤センターでは今回のアンケートなどでいただいたご要望を参考に、システム改善のための増強計画を進めて参ります。ご意見などは常時 voice@cc.u-tokyo.ac.jp で承りますので、要望などがございましたらご連絡ください。また、今後、ファイルシステム増強計画が進むに従いまして、より具体的な内容のアンケートをお願いする可能性があります。その際は再びご協力をお願いいたします。

今後とも HA8000 クラスタシステムのご利用をよろしくお願いいたします。

（アンケート担当：松葉　浩也）