

HULFT-HUB 性能検証資料

～ FUJITSU Server PRIMERGY RX200 S7 ～

作成日:2013/10/07

株式会社セゾン情報システムズ
HULFT事業部

目次

- ▶ 検証環境
 - 検証機器
 - 検証構成
 - HULFT-HUB検証パターン
 - 検証実施パターン
- ▶ 検証結果 - ジョブパターン -
 - ジョブパターン検証構成
 - ジョブパターン検証結果
 - ジョブパターン検証結果
 - ジョブパターンの結論
- ▶ 検証結果 - 蓄積パターン -
 - 蓄積パターン検証構成
 - 蓄積パターン検証結果
 - 蓄積パターン検証結果
 - 蓄積パターン検証結果
 - 蓄積パターンの結論
- ▶ 検証結果 - 中継パターン -
 - 中継パターン検証構成
 - 中継パターン検証結果
 - 中継パターンの結論



検証環境

検証機器

▶ 導入ソフトウェア

- HULFT-HUB Server for Linux-ENT Ver.3.2.1
- Red Hat Enterprise Linux 6.4 – 32Bit 互換ライブラリ

▶ ハードウェア

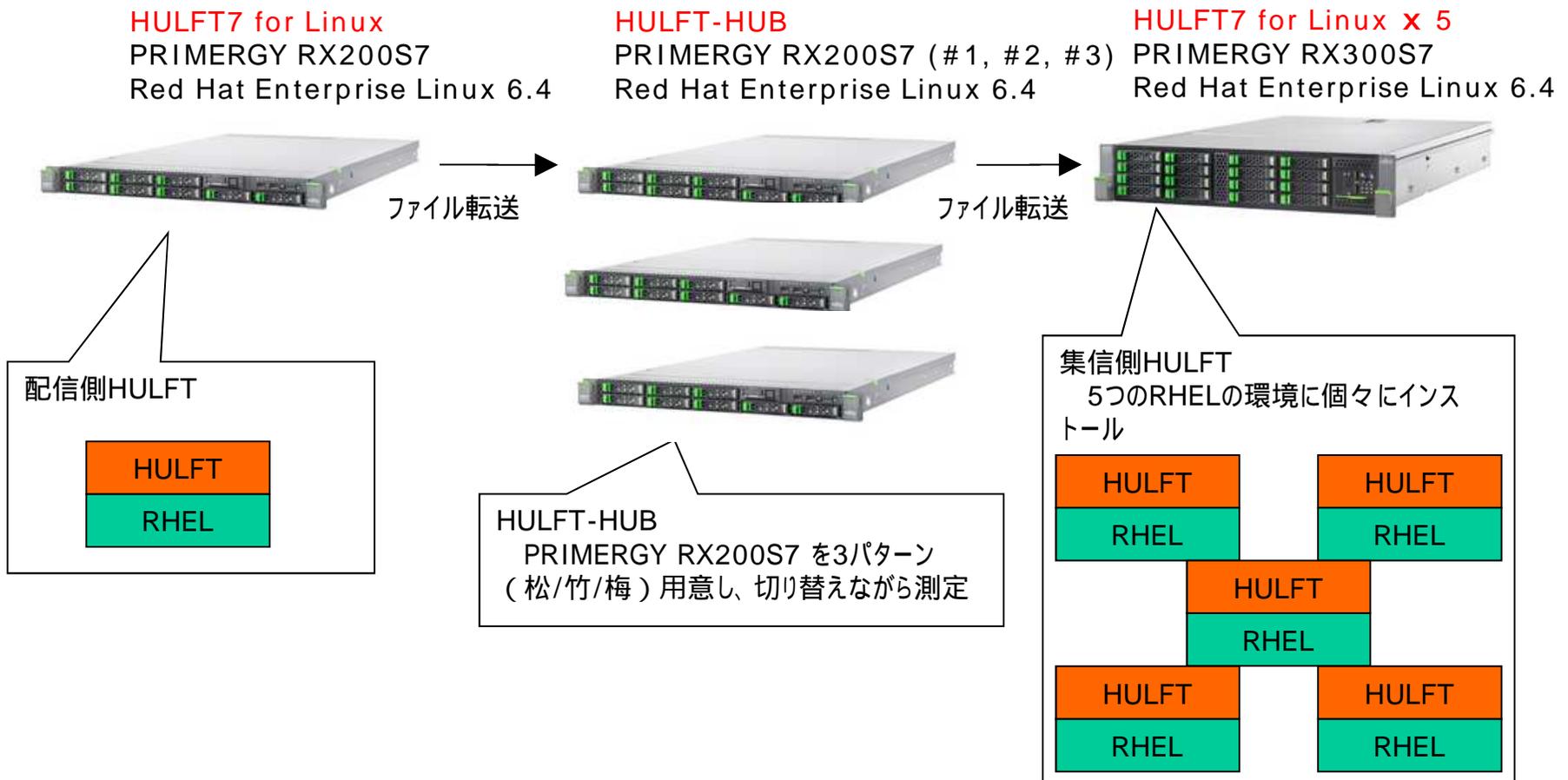
- 「PRIMERGY RX200 S7」
スペックの異なる3パターンの機器を用意

	PRIMERGY RX200 S7		
	#1:松	#2:竹	#3:梅
CPU	Xeon E5-2690 2.90GHz/8コア/20MB	Xeon E5-2690 2.90GHz/8コア/20MB	Xeon E5-2643 3.30GHz/4コア/10MB
	x 2	x 1	x 1
MEM	32GB	16GB	8GB
HDD	200GB-SSD(PY-SD20NB)	600GB(PY-SH601C)	300GB(PYBSH305C)
	x 4 (RAID 1 + 0)	x 2	x 2
LAN	Dual port LANカード (10GBASE)	Dual port LANカード (10GBASE)	Dual port LANカード (10GBASE)

検証構成

「HULFT」 「HULFT-HUB」 「HULFT」の流れでファイル転送を実施

- CPU、メモリ、ディスクI/Oの項目を測定
- HULFT-HUBの転送履歴にて転送時間を測定



HULFT-HUB検証パターン

「HULFT-HUB」の下記利用パターンを検証モデルとして設定

▶ ジョブパターン …… 処理の自動化

予め作成したジョブフローによる実行制御が可能です。HULFT-HUBサーバ上で、データの待ち合わせや、データの加工、転送監視などの処理を自動化します。



▶ 蓄積パターン …… 非同期連携

転送データをHULFT-HUBサーバ上で蓄積できます。これにより、集信側への転送不能時のみの一旦蓄積や、蓄積してから順次送出など様々な制御が可能になります。



▶ 中継パターン …… セキュリティ強化

通信の中継・同法機能により、転送と運用管理、両方のセキュリティを強化できるので、HULFT転送のプロキシとして利用できます。



検証実施パターン

各検証パターンを「#1：松」、「#2：竹」、「#3：梅」のマシンスペックで測定

○：測定実施
-：未測定

		ジョブ	蓄積 (蓄積同時転送)	蓄積 (蓄積後送)	中継
		20~100多重	20~100多重	20~100多重	1,000多重
#1： 松	暗号あり	○	-	-	-
	暗号なし	-	-	-	-
#2： 竹	暗号あり	○	-	○	-
	暗号なし	-	○	○	-
#3： 梅	暗号あり	-	-	-	-
	暗号なし	-	-	-	○

- 「ジョブパターン」は、蓄積 処理 送付 データ削除の一連の処理を測定
- 「蓄積パターン」は、蓄積と転送を同時行う処理と、蓄積後に送付を行う処理の2通りで測定
- 「蓄積パターン」と「ジョブパターン」は多重度（20~100）を変えて測定
- 「中継パターン」は、低スペックでも問題ないことを確認するため1,000多重で測定

検証結果 -ジョブパターン-

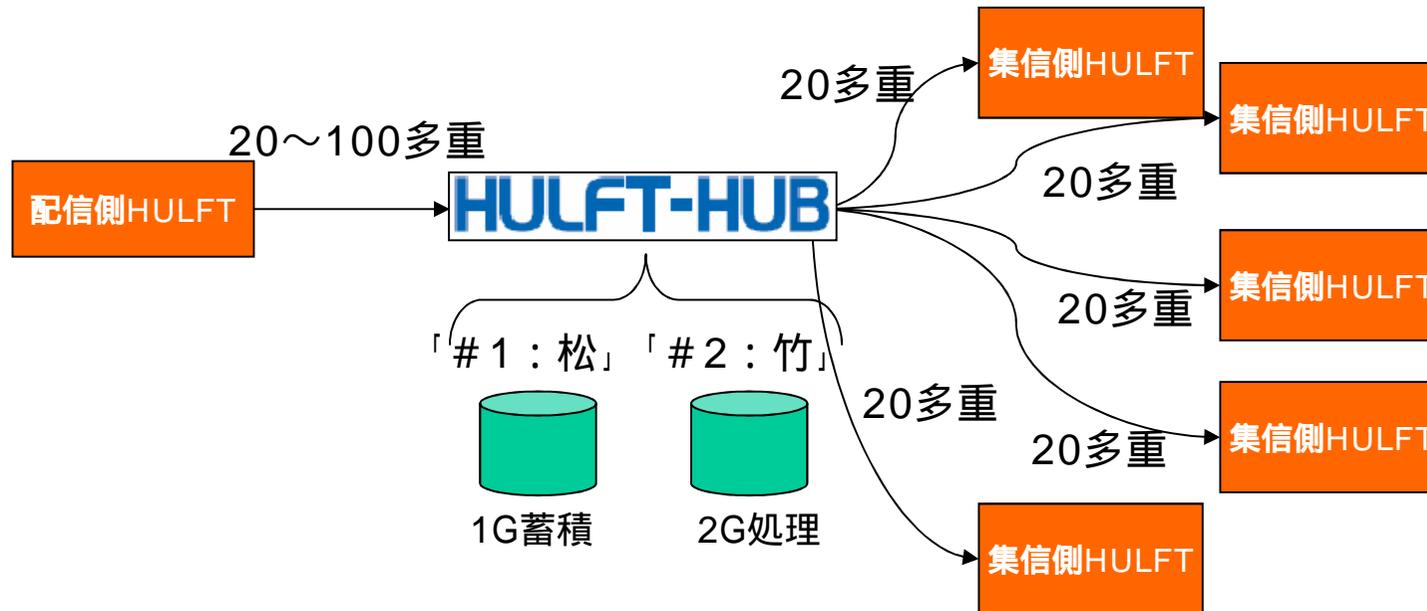
ジョブパターン検証構成

10MB × 20、40、60、100多重転送（配信側HULFTで5つの集信側HULFTへ同報）
HULFT-HUBのマシンは「#1：松」、「#2：竹」構成にて測定
20ファイル/1転送として受信し、ディスク負荷を分散
集信側のRAMディスク利用で、ディスクI/Oネックを回避



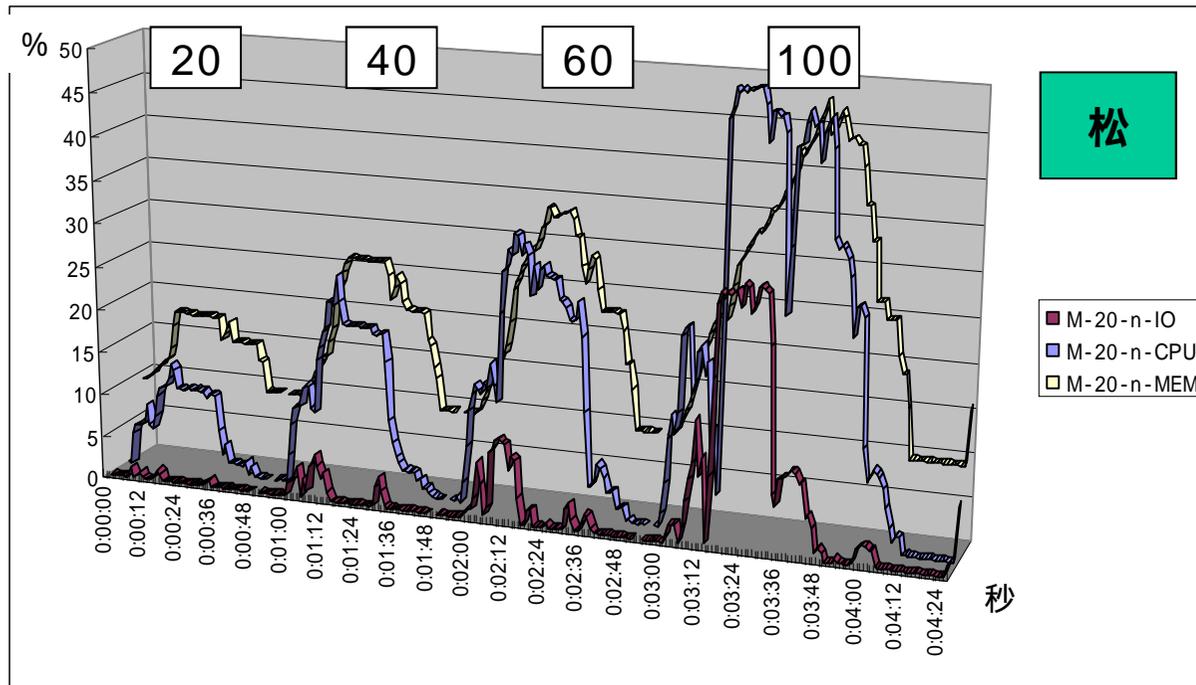
配信側：10MB × 20、40、60、100多重転送

集信側：20ファイル毎

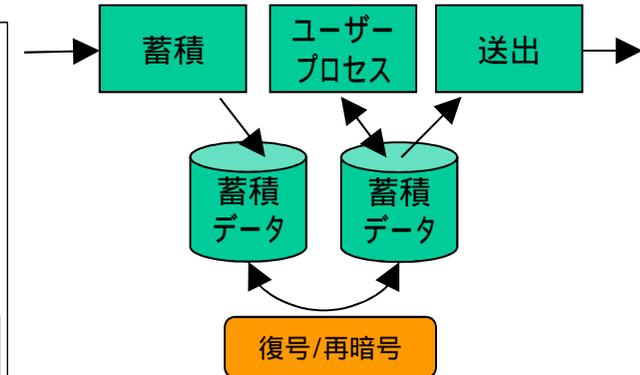


ジョブパターン検証結果

- ▶ 測定機器：「#1：松」
- ▶ 20～100多重のジョブ処理



「ジョブパターン」の動作



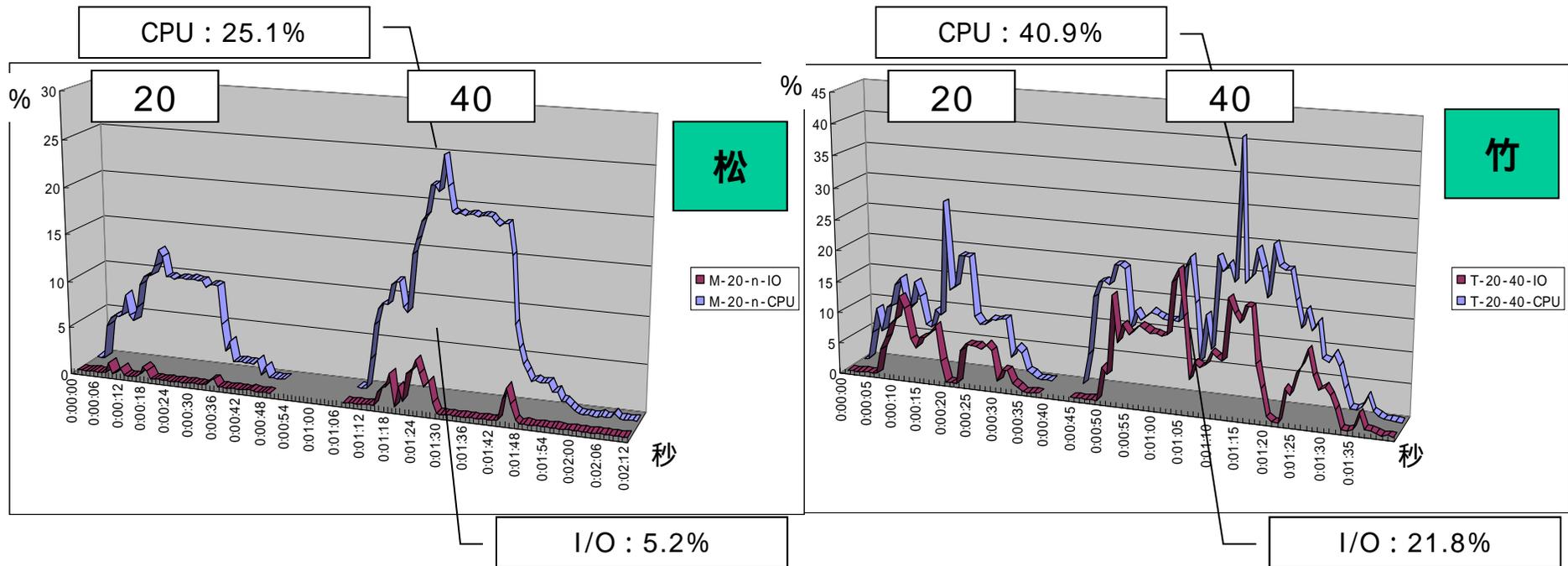
転送データを受信後、ジョブフロー実行プロセス起動、データの取り出し、ユーザプロセス実行、データ格納、転送、データ削除を行う。

データの取り出しと格納時も再暗号処理を行う。

- 多重度の増加に合わせてリニアにCPU負荷が増加
- メモリ使用量は、他の検証パターンに比べて「ジョブパターン」が一番高い

ジョブパターン検証結果

- ▶ 測定機器：「#1：松」、「#2：竹」
- ▶ 20、40多重のジョブ処理



- 「松」は、40多重でもCPU使用率が25%程度で余力あり
- 「松」は、「竹」に比べてディスクI/Oが数%程度の発生でスムーズに処理が実行

ジョブパターンの結論

推奨機器 「#1：松」

「ジョブパターン」では、ファイル転送単位でデータの蓄積、取得、加工、格納など、最低でも4回以上のディスクI/Oが発生し、かつ、データへの処理制御と、それを加工するユーザジョブ、送出处理などCPU、メモリーも高負荷になる場合が多いため、最上位構成の「松」を推奨。

▶ CPU

転送データが暗号化されている場合には、データの取り出し、格納、転送時に復号と再暗号処理が行われるため、「松」以外の構成ではCPU使用率が高くなりすぎる。

▶ メモリ

多重度に比例して増加するが、100多重時でも47.9%の使用量にとどまる。

▶ ディスクI/O

複数回のディスクI/Oが発生するが、ディスク構成が有効に機能している。

検証結果 -蓄積パターン-

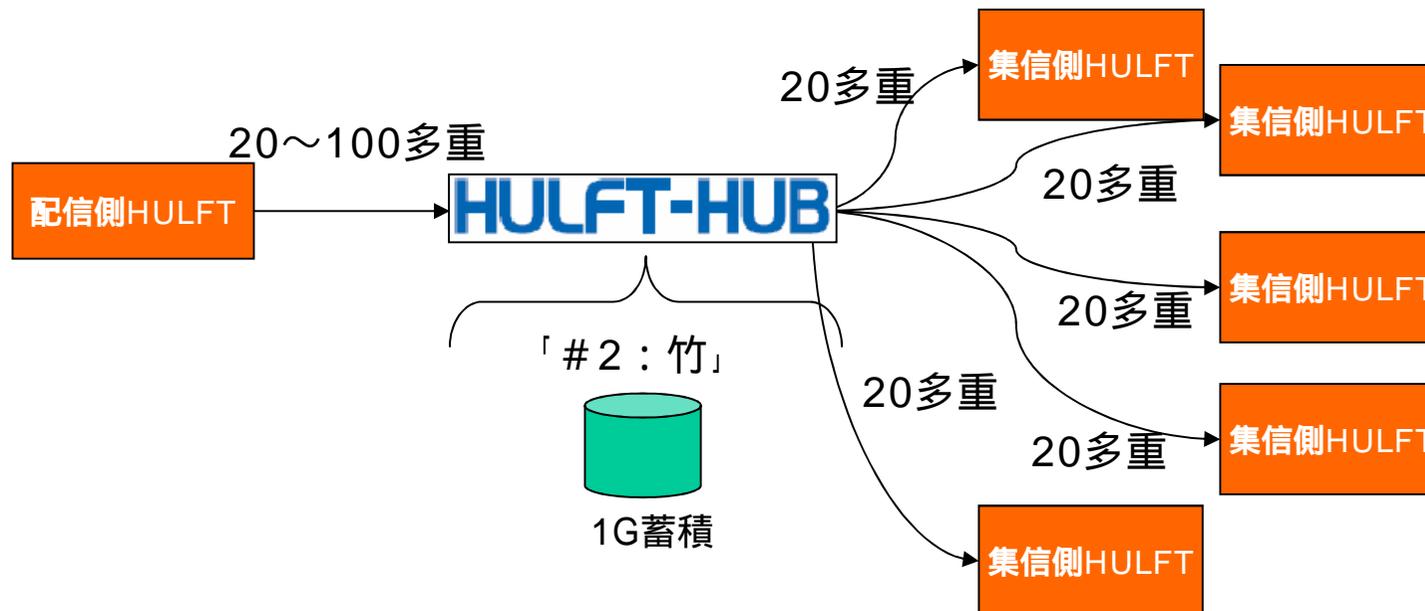
蓄積パターン検証構成

10MB × 20、40、60、100多重転送（配信側HULFTで5つの集信側HULFTへ同報）
HULFT-HUBのマシンは「#2：竹」構成にて測定
20ファイル/1転送として受信し、ディスク負荷を分散
集信側のRAMディスク利用で、ディスクI/Oネックを回避



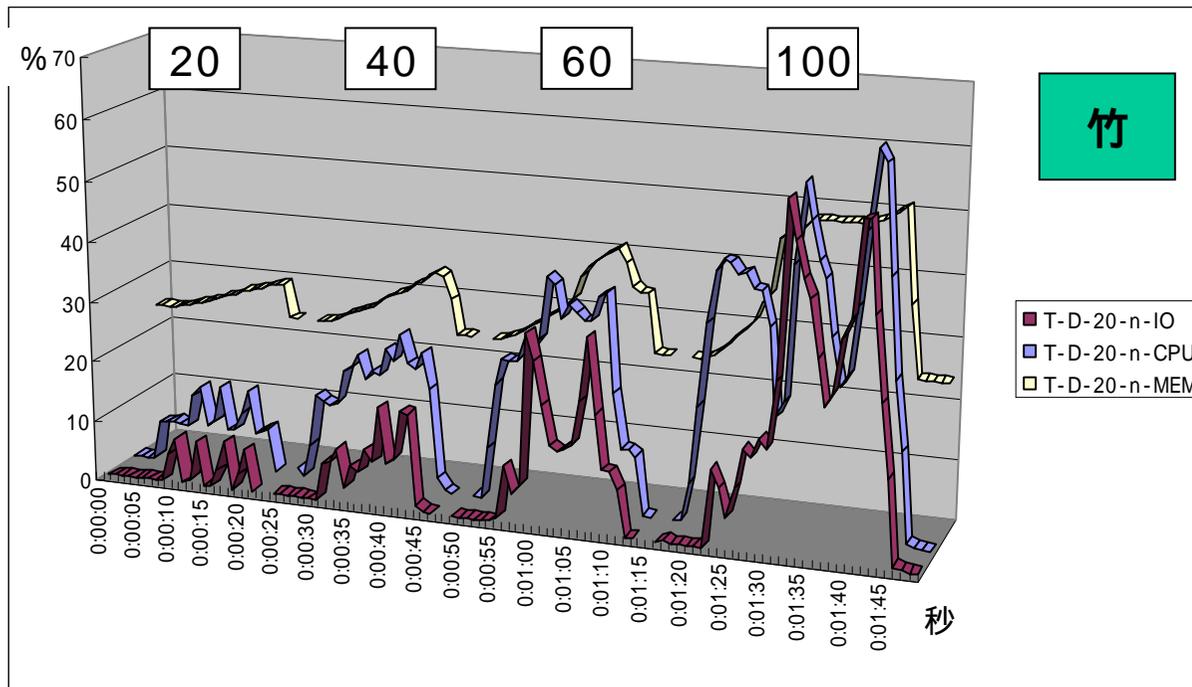
配信側：10MB × 20、40、60、100多重転送

集信側：20ファイル毎

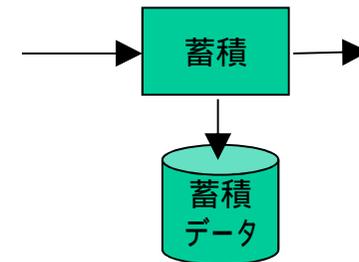


蓄積パターン検証結果

- ▶ 測定機器：「#2：竹」
- ▶ 20～100多重の「転送と蓄積を同時に行う」処理



「転送と蓄積を同時に行う」動作

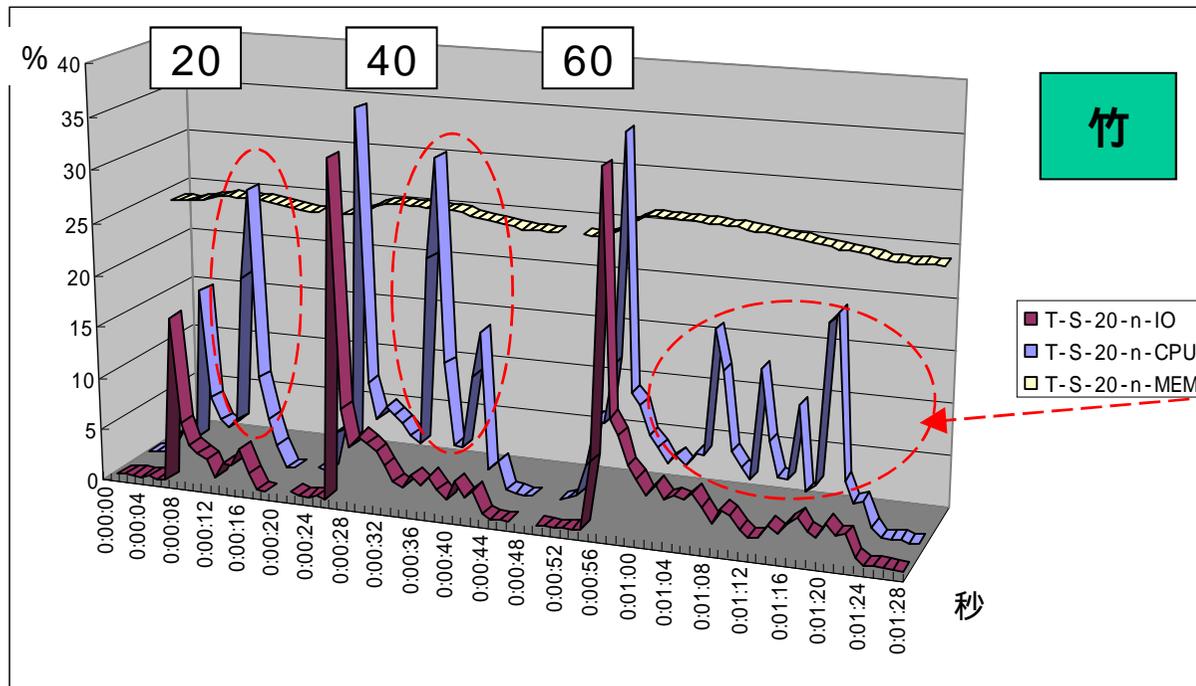


配信側HULFTから集信側HULFTへデータを転送しながら、ディスクへの書き込みも同時に行う。

- 転送と蓄積を同時に行なう場合、多重度の増加にあわせてリニアにCPU負荷が増加
- CPUピークは処理後半の山で、転送完了時に行われる「データの削除」処理が要因

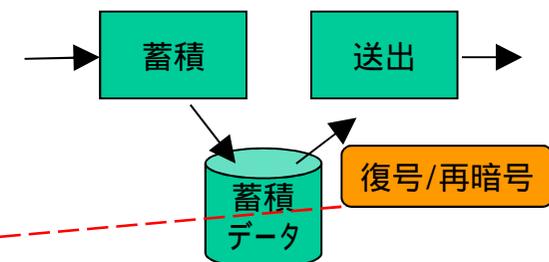
蓄積パターン検証結果

- ▶ 測定機器：「#2：竹」
- ▶ 20～60多重の「蓄積後に転送を行う」処理
- ▶ 転送データはAES暗号データ



「蓄積後に送出行を」動作

AES暗号データを転送

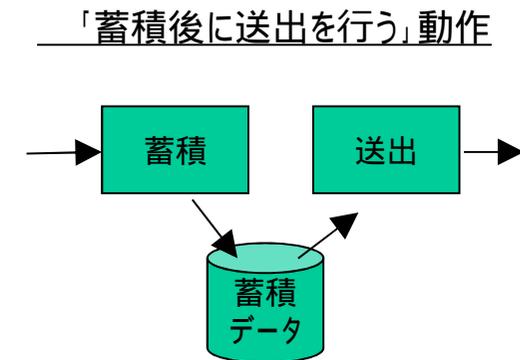
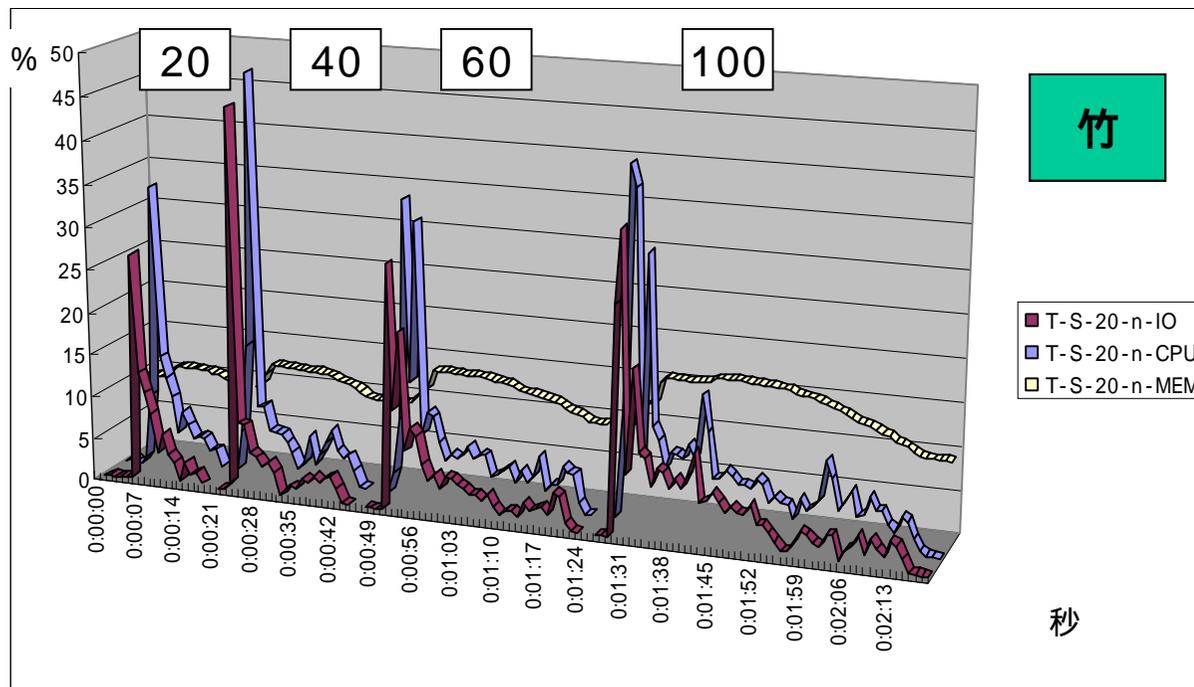


配信側HULFTからのデータをディスクに書き込み、その後、集信側HULFTへ転送を行う際に再暗号処理を行う。

- CPUピークは、データ蓄積時の山で、後半のデータ転送時にもCPU使用率が高くなっているのはAES暗号データの復号と再暗号処理が行われるのが要因

蓄積パターン検証結果

- ▶ 測定機器：「#2：竹」
- ▶ 20～100多重の「蓄積後に転送を行う」処理
- ▶ 転送データは暗号なし



配信側HULFTからのデータをディスクに書き込み、その後、集信側HULFTへ転送を行う

- CPUピークは、データ蓄積時の山で、後半のデータ転送時のCPU使用率は低い
- メモリ使用量は、AES暗号データ転送時に比べ、半分近く減少

蓄積パターンの結論

推奨機器 「#2：竹」

転送データを一旦蓄積してから転送を行う「蓄積パターン」では、中間ファイルを作成しない「中継パターン」とは異なり、ファイル転送単位でのディスクI/Oと、別途プロセスを生成するため、ディスクI/O性能とCPUリソースの高い「竹」構成を推奨。

▶ CPU

転送データの蓄積時にCPU負荷がかかり、また、転送データが暗号化されている場合は、復号と再暗号処理に負荷がかかるため、「竹」構成以上は必要。

▶ メモリ

転送と蓄積を同時に行う場合は、多重度に比例して増加するが、100多重時でも51.4%の使用量にとどまる。

▶ ディスクI/O

複数回のディスクI/Oが発生するが、ディスク構成が有効に機能している。

検証結果 -中継パターン-

中継パターン検証構成

10MB × 1,000多重転送（配信側HULFTで5つの集信側HULFTへ同報）

HULFT-HUBのマシンは「#3：梅」構成にて測定

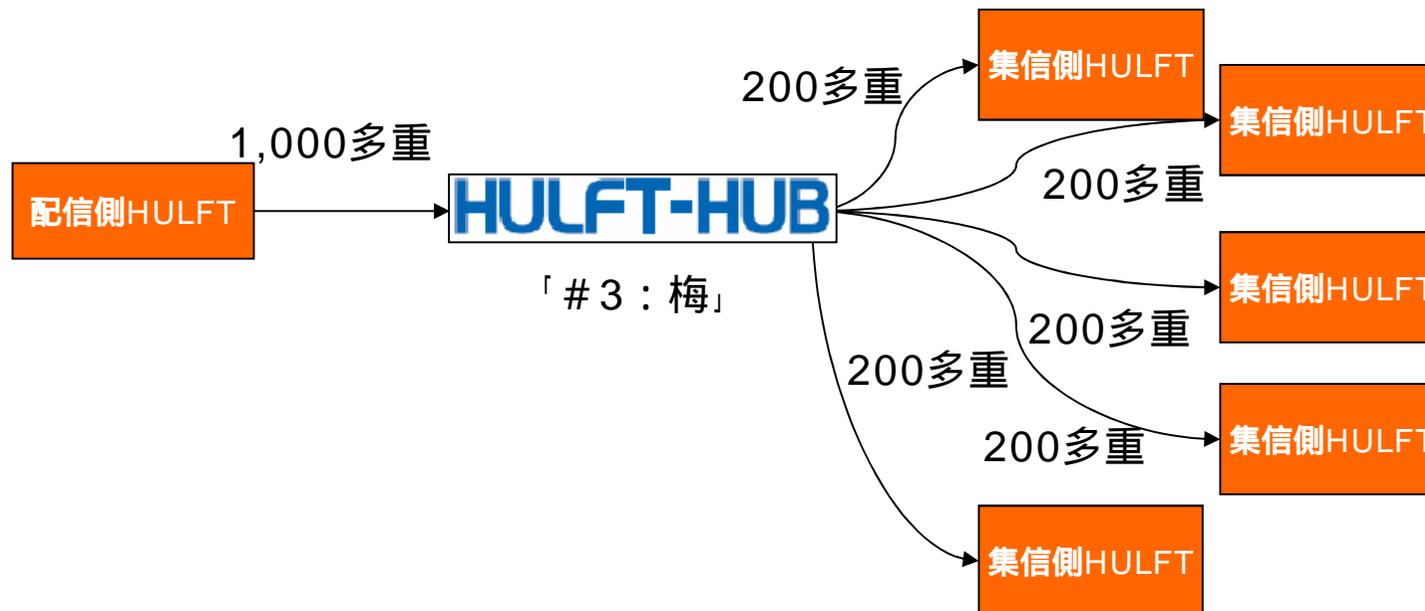
200ファイル/1転送として受信し、ディスク負荷を分散

集信側のRAMディスク利用で、ディスクI/Oネックを回避



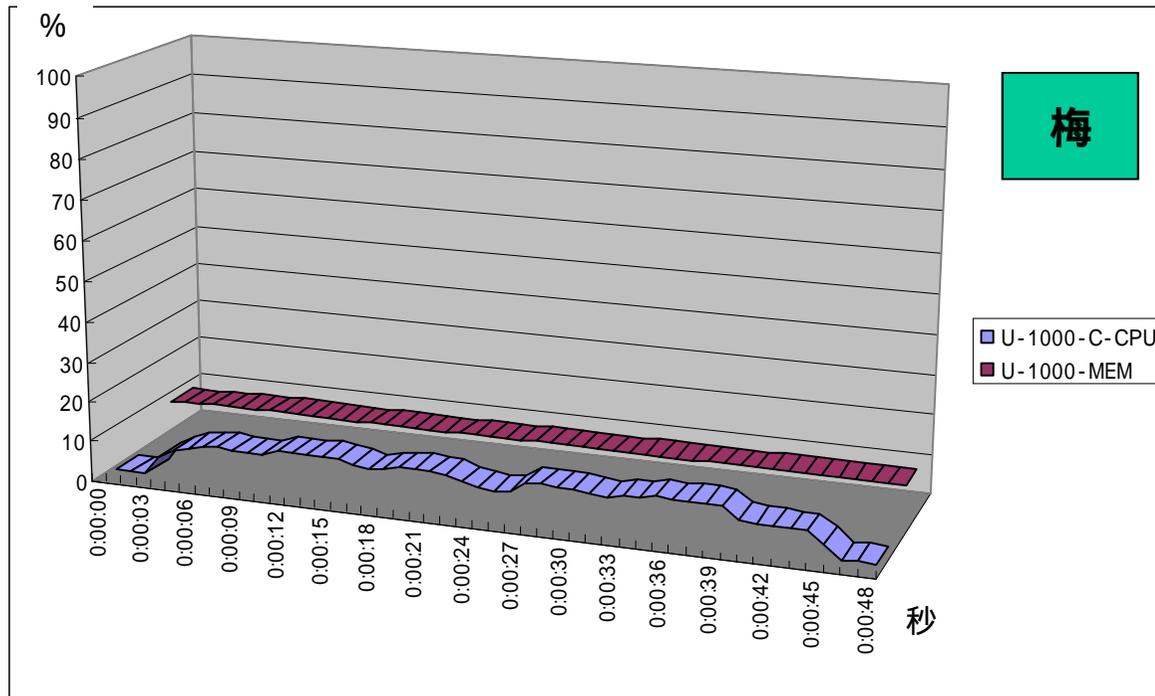
配信側：10MB × 1,000多重転送

集信側：200ファイル毎

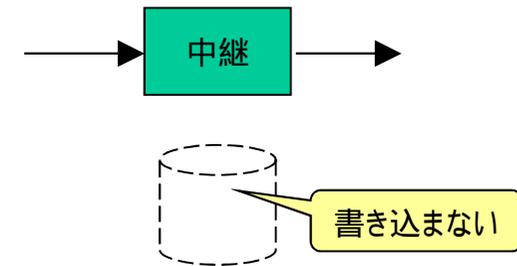


中継パターン検証結果

- ▶ 測定機器：「#3：梅」
- ▶ 1,000多重の中継処理



「中継パターン」の動作



転送データをディスクへ書き込むことなく集信側HULFTへ転送する。

- CPU負荷は、ピーク時で10%程度
- メモリ使用量は起動時の9.3% から9.8%へ僅かに増加

中継パターンの結論

推奨機器 「#3：梅」

ファイルを溜めずに中継し、また、複数のファイル転送を1プロセスで処理を行うことで、リソースや処理性能に影響が少ないため、低リソースの「梅」構成を推奨。

▶ CPU

配信側HULFTの転送能力が低すぎたため、HULFT-HUBの中継多重度が抑えられてしまったが、いずれにしてもCPU負荷はほぼないと思われる。

▶ メモリ

転送データを効率的に中継するため、メモリ使用量が「ジョブパターン」や「蓄積パターン」よりも大幅に抑えられている。

▶ Disk I/O

Disk I/O負荷は、ほぼゼロの状態。

お問い合わせ先

【取扱い会社】

富士通ミドルウェア株式会社

パートナーアライアンス統括部 プロダクト営業部

Tel : 045-475-9626

Mail : fmw-hulft1@ml.jp.fujitsu.com

<http://jp.fujitsu.com/group/fmw/services/hulft/>

【お問い合わせ先(開発元)】



株式会社 セゾン情報システムズ

HULFT事業部 パートナービジネス営業部

Tel : 0120-80-8680 (受付時間 : 平日9 : 30 ~ 18 : 00)

Mail : hulft@saison.co.jp

本資料の測定値について

実運用環境によって測定結果が変わる可能性がありますので、マシン構成を選定する際のあくまでも参考値としてお考え下さい。