

9. LTFS とは

Linear Tape File System (以降 LTFS)とは、ストレージベンダーが共同で策定しているテープストレージ用の記録フォーマットであり、このフォーマットは 2016 年に国際標準規格として ISO にも採用されている。広義には LTFS フォーマットの実装やその製品を示す意味でも使用されている。LTFS フォーマットは、その名前が示すようにファイルシステムを経由してテープにアクセスするためにデザインされており、これによって一般的なファイルブラウザを用いてテープ内にファイルを保存したり、保存されたファイルを確認したりすることが実現されたのである。

この LTFS は 2010 年からオープンに技術公開されており、現在ではファイルシステム機能だけを単独で提供する無償ソフトウェアから、コンテンツ管理や業務ワークフローとの連携などの付加機能を統合した応用ソフトウェアまで、ハードウェアメーカーやサードパーティーソフトウェア会社で採用されている。そして、どのベンダーの LTFS 実装も共通する LTFS フォーマットで書き込まれているので、ソフトウェア間でのデータの受け渡しや世代を超えたデータの引継ぎも容易になったのである。

9.1. 共通規格登場の背景

コンピューター用テープストレージの登場は 1950 年代にまで遡るが、それぞれの時代において、テープの優れた特性を活かせるように他のストレージと使い分けられてきた。例えば、頻繁にアクセスするデータの保存先としてはディスクやフラッシュストレージを用いるが、アクセス頻度が低いながらも大容量で高信頼性のストレージを必要とする場合はテープを利用する、といった具合である。どちらか一方ですべてをまかなうのではなく、適材適所で用いるハイブリッドなストレージ環境が求められている。

近年のビッグデータ時代においては、IoT や AI 技術の普及によって研究開発機関やオフィス業務で分析対象となるデータの収集と蓄積が行われ、また SNS や e-Commerce などの生活スタイルの変化もデジタルデータを増加させる要因になっている。その一方で、監視カメラの映像や電子取引データ、サーバーのログなどは、リスクマネジメントの観点で保全期間を長くする要求が高くなり、これもストレージの容量を押し上げる要因になっている。このようなデータの爆発的な増大が現実となるなかで、テープが本来持っている特性である低コスト、大容量、長期保管特性が着目されており、前述のバックアップ用途の例だけではなく、アクセス頻度が低いデータをアーカイブ化することによってデータセンターの設備投資や運用のコストの最適化を図る利用が増えてきている。

留意すべきことであるが、「バックアップ」と「アーカイブ」は、オリジナルのデータがどこに存在するのかという点において大きく考え方を異にしている。バックアップは、オリジナルデータの複製をオリジナルとは別の場所に定期的に保管しておき、万一の災害や人的ミスによってオリジナルが失われた際に直前の複製によって修復することが役割である。一方でアーカイブは、オリジナルデータの複製を作るのではなく、オリジナルデータを長期保存に適したストレージに保管する。そのため、テープの長期保存特

性や高信頼性ととも、長期的にデータにアクセスできる環境を維持することが大切なのである。

■ 従来はバックアップが主体

- 自然災害、システム障害および人為的ミスなどによるデータ損失発生時に業務継続を実現
 - ・ 一次ストレージの複製としてデータを記録
 - ・ 定期的な上書きによるテープの再利用
- バックアップソフトウェアとライブラリー装置による自動運用
 - ・ 災害対策サイトを含むデータセンタ内での、主にクローズドな利用

データを長期保存し新しい利用を促進する要因

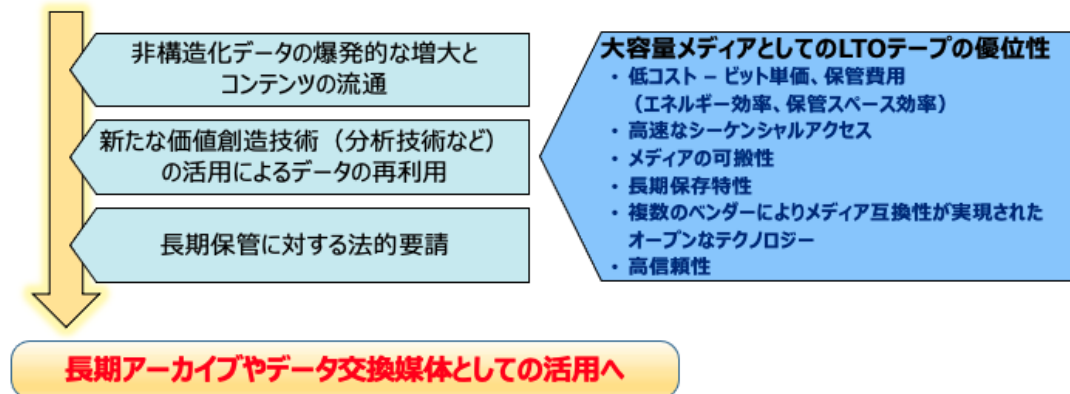


図 1. テープの新たな活用

9.2. アーカイブストレージの要件

データをディスクからテープに移動することによってディスクストレージの容量増加を抑制することは、従来から階層化ストレージ(Hierarchical Storage Management: HSM)として知られている手法だが、一般に HSM が導入された環境からテープだけを取り出して参照することはできない。ストレージシステムを更新・移行する際には、いったんすべてのデータをテープから読み戻すなどの時間的負担が発生するようなケースもある。データに対するアクセスの利便性のためにはディスクとの密接な統合が重要であるが、それと同時に、アーカイブの長期的な運用を考えた場合にはストレージシステムの更新サイクルの影響を受けにくい特性を兼ね備えたストレージが必要となる。

図 2 はアーカイブストレージに対するさまざまな要件を 3 つのレイヤーにマッピングしたもので、LTFS は上 2 つを念頭に作られている。例えば、要件(3)に対して LTFS フォーマットは自己記述性をもつように設計されており、外部のデータベースなどから切り離してテープ単独でアクセスできるようにしている。また、LTFS フォーマットはオープンな規格であり、要件(1)や(6)への解決策となっている。

アクセスインターフェースはアーカイブシステムにデータを入出力する方式で、LTFS ファイルシステム実装は POSIX 互換のファイルシステムを採用している。また、Openstack Swift や Amazon S3 のようなオブジェクトインターフェースを介したテープアーカイブの利用も始まっている。

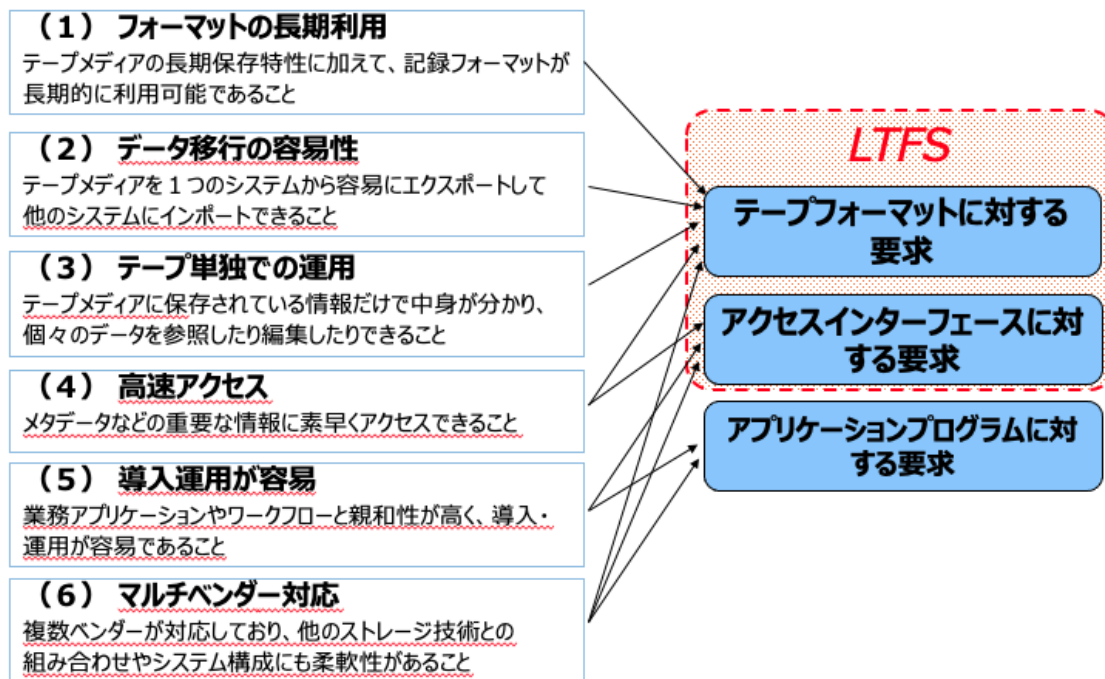


図 2. 長期アーカイブにおける要件

9.3. LTFS フォーマットの標準化

LTFS が登場した初期段階では、設計を行った IBM が LTFS フォーマット仕様書を公開していたが、Version 2.2 以降は Storage Networking Industry Association (SNIA) の技術ワークグループによる管理体制に移行し、2021 年 8 月現在の最新版は 2020 年に承認された Version 2.5.1 である。また 2016 年に国際標準 ISO/IEC 20919:2016 として承認された LTFS フォーマットは、2021 年には Version 2.5.1 相当するものが、ISO/IEC 20919:2021 として更新・承認されている。

この LTFS フォーマット仕様書ではテープ上のデータの並びやメタデータの XML 記述書式などの論理的フォーマットが定義されており、一方でファイルシステムの API や内部挙動、テープドライブへのコマンドプロトコルなどについては、GitHub 上で公開されている LTFS ファイルシステム実装のソースコードが参考になる。

また LTFS Compliance Verification という第三者検証テストを LTO コンソーシアムが運営しており、このロゴプログラムでは検証対象のソフトウェアが作成したテープがフォーマット仕様書に準拠していることを検証している。

このように LTFS はフォーマット仕様書、オープンソース、互換性検証の 3 つの柱によって支えられており、利用者が安心して利用できるようなエコシステムを形成している。

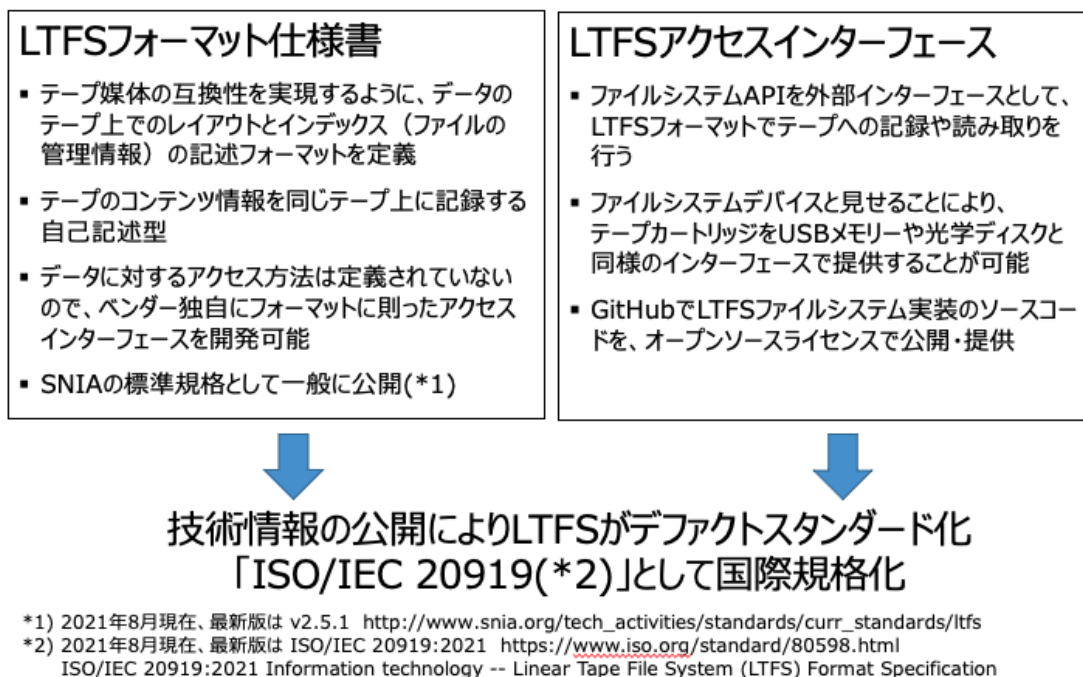


図 3. LTFS を構成する要素

9.4. LTFS の特長

LTFS フォーマットはテープをファイルシステムとして使用することを前提としているため、通常のディスクベースのファイルシステムとは異なる設計上の工夫が存在している。

一番の特徴は、テープをインデックス・パーティション(以降 IP)とデータ・パーティション(以降 DP)に二分割し、一方を上書き型で他方を追記型記録で利用していることと言える。これにより、テープの先頭部分に常に最新のインデックスを保存することが可能になり、テープのマウント処理が高速化されている。このテープのパーティショニング機能は LTO-5 から備わった機能で、LTFS が動作する他の種類のテープドライブでも必須の機能である。フォーマット仕様書上は、パーティションの大きさの規定はないが、通常は IP を最小にして残りすべてを DP に充てる。LTO の場合、IP に 2 ラップ、ガードバンドと呼ばれる 2 つのパーティションを分ける領域として 2 ラップ、また DP 終端 1 ラップ分が最終インデックス用としてリザーブされているため、全体の数パーセントがデータの書き込み領域としては使われない計算である。例えば LTO-9 テープの場合テープトータルで 280 個のラップあるので、 $(2+2+1)/280=1.7\%$ がこれに当たる。(ラップはテープ長手方向にテープ両端間に連続する記録領域)

二番目の特徴は、複数のインデックスを保管していることである。インデックスは、ファイルの名前や属性、タイムスタンプなどのメタデータを記述したものであり、XML 形式で書かれている。このインデックスは IP に最新版が 1 つだけ書かれているが、DP には過去のインデックスのすべてが収められており、

古いインデックスを使えば過去のある時点でのファイル情報の復元(ロールバック)が可能である。通常のディスク用ファイルシステムでは、メタデータをディスク上から適宜部分的に読み出して処理をしているが、LTFS の場合は、テープがランダムアクセスにならないようにマウント時に全メタデータをメモリー上に展開して動作している。1ファイルあたり約1KBのメモリーが消費されることになり、100万ファイルでは約1GBの空きメモリーの確保が推奨されている。

■ LTO-5から、テープのパーティショニングをサポート

■ 1本のテープを非対称に論理2分割

- ✓ **インデックスパーティション**：インデックスを保存する小さい領域。
最新のインデックスをテープの先頭部分に配置し、テープに含まれるユーザーデータの内容をテープロード直後に高速に一覧可能
- ✓ **データパーティション**：ユーザーデータを追記保管する大きい領域

図：分割されたテープとファイル保存の模式

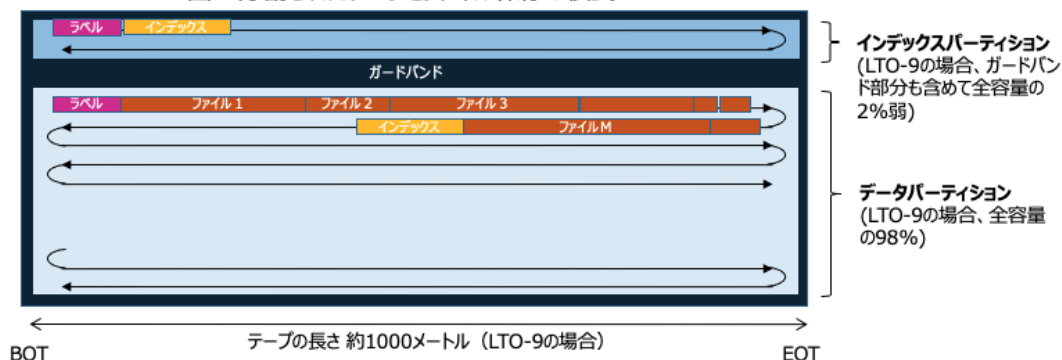


図4. テープ上のレイアウト

第三の特徴は、ファイルの書き込み・更新時におけるテープ上でのストレージの使い方にある。ファイルは追記型で利用される DP に連続して書き込まれるので、読み出す際もテープ本来の転送速度で高速に読み出すことができる。その一方で、追記型のためファイルを削除した場合にそのファイルが書かれていた領域を再利用することはできず、代わりにメタデータからその領域への参照を削除する。このことは逆に言えば削除済みのデータもインデックスのロールバックで読み出し可能であるとも言えるが、削除されたファイルが多くなると不必要な領域が離散的にテープに発生することになり、次第にテープのスペース利用効率が悪くなる。そのため、削除領域が大きくなった際に別テープに詰め込んで書き込む機能(リクレイム機能)を有するソフトウェアも存在する。

またファイルの一部だけを書き換えるようなファイル操作が行われた場合には、コピーオンライト方式で差分だけがテープに追記されるので無駄なスペースは発生しない。

他に、配置ポリシーの設定により、サムネイルやカタログリスト等、特定のファイルを IP にも配置することで、後のマウント時に効率良くまとめてアクセスできるようにする仕組みも提供している。

ファイルシステムの直接の機能ではないが、テープドライブに備わるデータ圧縮機能も活用されており、残量以上のサイズのファイルを書き込むことが可能である。その際、圧縮率を書き込み前に予測できないため、ファイルをコピーする毎に残量を確認しながら使うことが必要である。

ファイルシステムによるアクセスのメリットは、ファイルを扱うアプリケーションがテープでもそのまま使えることにあるが、時にはディスクでは気づきにくいアクセスがバックグラウンドで自動的に発生していることが顕在化してくる。例えば、ファイルブラウザは JPEG ファイルを見つけるとサムネイル画像を生成していたり、またファイルの検索が用意になるようにファイルから要約情報を抽出してインデクシングを行っていたりする OS もある。これらがテープへの書き込み時などにパフォーマンスに影響を与える場合には、設定変更によって快適に LTFS を利用することができる。

9.5. LTFS が開くテープの新しい使い方

テープがオープンで自己記述性をもつ媒体になると、どのような事が可能になるのだろうか？

一例としては、

- 1) USB フラッシュメモリーと同じような操作感で、テープをマウントするだけで書き込まれている内容がファイルブラウザで一覧でき、またソフトウェアでファイルを直接開くことができる。
- 2) LTFS フォーマットで書かれたテープは LTFS 対応ソフトウェアがインストールされた PC であればどこでも読み出すことが可能なので、大量のデータを LTFS テープに書き込んで別の手に手渡すことが可能になる。
- 3) テープを書き込んだ PC の OS バージョンやツールが時間の経過とともに古くなっても、新しい PC に LTFS ソフトウェアをインストールすれば、そのままテープを読み出すことができる。

これらが示すように、LTFS によって、テープの長期保管性能、可搬性、大容量といった特長にファイルアクセスの利便性が加味され、従来のバックアップ用途とは異なる新しい使い方が可能になったのである。

2010 年に LTFS が初登場した際には、まずは放送や映画産業に革新をもたらした。この動きは現在では異なる産業分野に広がっており、ビッグデータ時代を支える長期アーカイブやデータ交換用フォーマットとして活躍している。