

4D Transporter

by Laurent Ribardière

ACI

4D Transporter

4D Transporter は、4th Dimension 3.5 と4D Server 1.5から追加された新しい 4D ユーティリティプログラムで、MacOS上で使用します。4D Transporter は、あるプラットフォーム上で開発されたデータベースを他のプラットフォームで使用するためにトランスポートする際に使います。



4D Transporter 6.0.1

4D Transporter では、次のような機能を実行します：

Macintosh から Windows にデータベースをトランスポートする際に、Macintosh の 4D データベースファイルや他の関連ファイル、文書等をデータフォークとリソースフォークに分割し、Windows 上で使用できるようにします。

Windows から Macintosh へとデータベースをトランスポートする際には、Windows の 4D データベースや他の関連ファイル、文書等をデータとリソースファイルをマージし、Macintosh 上で使用できるようにします。

Macintosh から Windows 上にトランスポートする場合、オプションで、DOS の 8 文字の名前と 3 文字の拡張子というファイル名の長さに関する制限（主として Windows 3.1）が適用されます。

トランスポートはなぜ必要か？

Macintosh 上で 4D データベースを作成する場合、プログラムでは 2つのファイル（ストラクチャファイルとデータファイル）が入ったフォルダを作ります。例えば「Airlines」データベースを作ると、「Airlines」というストラクチャファイルと「Airlines.data」というデータファイルが入った「Airlines f」という名前のフォルダができます。



Windows 上で 4D データベースを作成する場合、プログラムによって 3つのファイル（ストラクチャファイル、リソースファイル、データファイル）が入ったディレクトリを作ります。例えば、「AIRLINES」データベースを作ると、「AIRLINES.4DB」というストラクチャファイル、「AIRLINES.RSR」というリソースファイル、「AIRLINES.4DD」というデータファイルが入った「AIRLINES」という名前のディレクトリができます。

注：Windows95を使用している場合、拡張子は画面上では表示されません。



このようなプラットフォーム間のファイル構成の相違は、Macintosh と Windows のオペレーティングシステムのファイルマネージャ（OS の一部で、ディスク上のファイルと文書の操作を行う）が大きく異なるために起こります。

Macintosh 上では、ファイルはすべて 2 つのフォーク（データフォークとリソースフォーク）と呼ばれる部分で構成されています。Windows 上では、ファイルには 1 つの部分しかありません。

Macintosh ファイルのデータフォークは、Windows ファイルや他のオペレーティングシステムのファイル（例えば UNIX）と理論的には同じです。Macintosh ファイルのリソースフォークにはリソースが納められています。

リソースは、ローカライズや修正を簡単に行えるようにするためのデータの断片です。これが Macintosh の設計者による本来の目的、つまり、アプリケーション全体を再構築したりせずに、Macintosh アプリケーションをある言語から他の言語に（例えば、英語からフランス語やスペイン語に）ローカライズできるようにするということでした。リソースには、文字列のリスト、アイコン、ピクチャなどが含まれています。また、ダイアログボックス、メニューなどを納めておくためにもリソースが使われます。例えば、Macintosh アプリケーションでは、メニューアイテムのテキストがリソースとして納められているので、メニューアイテムのスペルの誤りはアプリケーション全体を再構築せずに修正することができます。

その上、年を経るに従い、リソースは他種類のデータを保持するためにも使われるようになりました。単純なバイトの連なり（つまり文字から成る Text 文書）であるデータフォークとは違い、Macintosh ファイルのリソースフォークはもともと、構造を持ったバイトの連なりです。リソースと、リソースをアクセスするためのマップを持っています。

Windows^{*1} にはリソースフォークの概念が存在しないので、データフォークとリソースフォークの両方を持つ Macintosh ファイルをコピーすると、Windows プラットフォームではデータフォークだけがコピーされ、認識されます。

そのため、Macintosh 上と Windows 上の両方で動作するアプリケーションで作業する時、アプリケーションで管理する文書に Macintosh 上で実行する際にデータフォークとリソースフォークの両方が使われている場合には、この問題を考慮する必要があります。

このことは 4th Dimension (4D First) にも当てはまります。Macintosh 上で、4th Dimension (4D First) は、データベースのストラクチャファイルとデータファイルに関してデータフォークとリソースフォークの両方を使います。

Macintosh 用の 4D データベースファイルを直接 PC にコピーしても、コピー後のファイルで Windows バージョンの 4th Dimension (4D First) は動作しません。ファイル内のリソースフォークに納められた重要な情報が失われているからです。

*1. 従って、このような仮定上のバージョンでは Windows 上のリソースフォークを扱うことができないので、Windows バージョンの 4D Transporter は存在しません。

4D Transporter はユーティリティプログラムで、Macintosh 上のデータベースをWindows にトランスポートしたり、またその逆のトランスポートを行います。Macintosh から Windows へトランスポートする際、Macintosh ファイルの 2 つのフォークは 2 つのファイルに分割され、Windows 上でコピーしたり、開いたりできるようになります。Windows から Macintosh へトランスポートする際には、逆の処理が行われ、ストラクチャとリソースのファイルがマージされます。

4D のMacintosh 上のデータベースファイルについて

4D データベースファイルのデータフォークには「デザイン」モードで作成したオブジェクトに関する定義情報がすべて納められています。これには、レイアウト、レイアウトスクリプト、プロシージャ、カスタムメニュー等が含まれます。

4D データベースファイルのリソースフォークは、データファイルへのパスなども含めた文字列のようなリソースを保持するために使われます。また、リソースフォークには、68K Macintosh 用の古い形式の 4D Extensions も納められています。4D ディベロッパによっては、データベースで使用する文字列リストやピクチャ等を納めておくこともあります。

注：Macintosh の 4D データベースのストラクチャファイルでは、データフォークとリソースフォークともに、数K バイト、時には数メガバイトの情報を納めておくことができます。

4D データベースのデータファイルのデータフォークには、レコードすべて、内部アドレステーブル、インデックスなどが保存されています。

4D データベースのデータファイルのリソースフォークは、一般に空ですが、リソースをいくつか含んでいることがあります。例えば、Customizer Plus を使ってデータファイルとストラクチャファイルに関連付ける場合、両方のファイルのリソースフォークに、同じ WEDD リソースを追加することにより内部的に行われます。

注：Macintosh 4D データベースのデータファイルのデータフォークには、数ギガバイトのデータを納めることができます。リソースフォークは通常、空か、または非常に小さいものです。

4D Transporterによる処理と4Dのプラットフォームに依存しない技法

4D Transporter によって処理されることは非常に単純です：

Macintosh から Windows へデータベースをトランスポートする場合、Macintosh ファイルのデータフォークとリソースフォークを、Windows で使用できる 2 つの個別のファイルにコピーします。

Windows から Macintosh へデータベースをトランスポートする場合、ファイルと関連するリソースファイル（存在していれば）を、ファイルをデータフォークに持ち、関連のリソースファイルをリソースファイルに持つような1 つの Macintosh ファイルにコピーします。

4D Transporter では、次のようにして Macintosh の 4D データベースをトランスポートしません。

ストラクチャファイルのデータフォークを .4DB ファイルにコピーします。Windows 上では、このファイルを「データベースストラクチャ」ファイルと呼びます。

ストラクチャファイルのリソースフォークを .RSR ファイルにコピーします。Windows 上では、このファイルを「データベースリソース」ファイルと呼びます。

データファイルのデータフォークを（空でなければ）.4DD ファイルにコピーします。Windows 上では、このファイルを「データベースデータ」ファイルと呼びます。

前で説明した Airlines の例では、Macintosh のデータベースの Windows へのトランスポートによって、AirlinesファイルからAirlines.4DBとAirlines.RSR が作られ、Airlines.data からAirlines.4DD（場合によっては、Airlines.4DRも）が作られます。

4D Transporterでは、次のようにして Windows の 4Dデータベースをトランスポートします。

Windows 上の .4DB ファイルと.RSR ファイルを 1 つのファイルにマージします。このファイルは Macintosh データベースのストラクチャファイルになります。

Windows 上の .4DD ファイル（場合によっては .4DR もともに）を 1 つのファイルにマージします。このファイルは Macintosh データベースのデータファイルになります。

前で説明した「Airlines」データベースの例では、Windows のデータベースの Macintosh へのトランスポートによって、Airlines.4DB と Airlines.RSR から Airlinesファイルが作られ、Airlines.4DD（場合によっては、Airlines.4DRも）から Airlines.data ファイルが作られます。

2 つの操作は対称的であり、片方のプラットフォームでデータベースを作成、または修正し、その後、他方のプラットフォームに何度でもトランスポートすることが可能です。

4th Dimension (4D First) のプラットフォームに依存しない技法により、残りの部分を処理します。使用中のプラットフォームが何であれ、4th Dimension (4D First) では、オブジェクトとレコードをプラットフォームに依存しない形で納めるので、両方のプラットフォーム上で問題なく参照することができます。さらに、Windows パージョンの 4th Dimension (4D First) から、データベース内に存在する Macintosh のリソースを利用することができます。

4D Transporter では処理されないこと

Macintosh上で作業する場合、データベース内に追加されたり、修正されたりするレコードは、データファイルに Macintosh の形式（Motorola のバイト並び）で書き込まれます。

Windows上で作業する場合、データベース内に追加されたり、修正されたりするレコードはデータファイルにWindowsの形式（Intelのバイト並び）で書かれます。

注：デザインオブジェクトは常にMacintoshの形式（Motorolaのバイト並び）で書かれます。

データベースが場合によって、MacintoshとWindowsの両方で使われる場合、データファイルにはMacintoshとWindows“ネイティブ”なレコードが混在するようになります。MacintoshとWindowsの両バージョンは、どちらのプラットフォーム上で書かれたレコードも処理するように設計されている、というのが方針です。

4D Transporterでは、ファイルの内部フォーマットの変換は行いません。内部的なデータ変換は一切行われず、Macintoshのファイルの分割と、Windowsファイルのマージのみが行われます。

データファイル全体をMacintoshまたはWindows“ネイティブ”なレコードにする場合は、4D Toolsを使用します。データファイルの圧縮または、タグによる修復を行います。前で説明したように、こうした処理を行う必要はありませんが、片方のプラットフォームでだけ稼働させる場合に無駄なバイトスワップ操作（ほとんど実感されません）をなくすることができます。

4D Transporter の使用

4D Transporterアイコンをダブルクリックして、4D Transporterのウィンドウを表示させます。

注：Windows上では、4D Transporterは使用できません。



4D Transporterには3通りの使い方があります：

4D Transporterを起動し、トランスポートのオプションを選択し、トランスポートするファイルを選び、トランスポート処理を実行します。

4D Transporterのアプリケーションのアイコン上に、ファイル（複数も可）をドラッグしてドロップし、トランスポートのオプションを選択し、トランスポート処理を実行します。

トランスポートのオプションをディスクに保存し、4D Transporter のアプリケーションのアイコン上に、ファイル（複数も可）をドラッグしてドロップすると、トランスポートは自動的に行われます。

4D Transporter のオプション

ここでは、4D Transporter ウィンドウのさまざまなオプションについて説明します。

「DOSファイル名をつける」オプション

データベース名は Macintosh とWindows の両方で 31 文字以内に制限されています。ですが、Windows 3.1のもとでは、ファイル名には DOS の制限である 8 文字の名前に 3 文字の拡張子という制限があります。「DOSファイル名をつける」オプション（デフォルトで選択されています）により、4D Transporter では、Macintosh から Windows へファイルをトランスポートする際にこの規則に従ってファイル名を修正します。下表は、このオプションが選択されている場合と、選択されていない場合について、Windows のストラクチャファイルがどうなるかという例を示したものです。

Macintosh 上の名前	Windows 上の名前 オプション選択なし	Windows 上の名前 オプション選択あり
Ultimate Invoicing DB	Ultimate_Invoicing_DB.4DB *1	LTMTNVCN.4DB
My Best DB, so far!	My_Best_DB,_so_far!.4DB *2	MBSTDBSF.4DB

*1. Windows 3.1 のもとでは ULTIMA 1.4DB になります。

*2. Windows 3.1 のもとでは MY_BS 1.4DB になります。

注：バージョン6の4th Dimension（4D First）では、Windows3.1上での動作は保証してません。

どちらの場合も、空白文字や、¥、*、)などの文字は下線文字に置き換えられます。DOS の命名規則を適用する場合、子音以外の文字は取り除かれ、名前が長すぎる場合は、8 文字になるように切り捨てられます。

どのオプションが選択された場合も、4th Dimension はトランスポート後のデータベースを使って作業することができます。DOS の命名規則の適用は、Windows 3.1.1 のもとでしか役に立ちませんが、他の Windows アプリケーション（長いファイル名がサポートされていないアプリケーション）からもファイルをアクセスできます。また、MBSTDBSF.4DBの方が MY_BES 1.4DBより理解しやすいと思う場合もあるでしょう。Windows NT あるいは Windows 95 のもとでは、長い名前を使うことができます。

「環境設定表示」オプション

このオプションにより、4D Transporter では、アプリケーションが起動されたり、アイコン上にファイルがドラッグしてドロップされた場合にウィンドウを表示します。これはデフォルトで選択されているオプションです。このオプションを選択しない場合、アイコン上にファイルをドラッグ、ドロップされた時に、4D Transporter はウィンドウを表示しなくなります。代わりに、ディスク上に保存されているトランスポートのオプションを使い、直ちにトランスポート処理を行います。

注：「環境設定表示」オプションが選択されていないと、4D Transporter のウインドウにアクセスできなくなってしまう。元に戻すためには、4D Transport の起動時に Option キーを押し続けます。

「設定内容保存」ボタン

何らかのオプションを変更し、次回 4D Transporter でそれを自動的に使おうとする場合は、このボタンをクリックします。オプションを保存し、「環境設定表示」オプションを選択しないと、ファイルのトランスポートを迅速に行うことができます。4D Transport は、直ちにトランスポート処理を実行し、終了します。

トランスポート：Mac PC

Macintosh から Windows へファイルをトランスポートする場合にこのオプションを選択します。

トランスポート：PC Mac

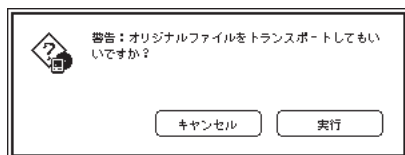
Windows から Macintosh へファイルをトランスポートする場合にこのオプションを選択します。

「オリジナルファイルをトランスポート」または「コピーしてからトランスポート」

デフォルトでは「コピーしてからトランスポート」オプションが選択されています。

「コピーしてからトランスポート」が選択されている場合、4D Transporter では、Macintosh から Windows にファイルをトランスポートするときには、「PC f」という名前のフォルダを作成し、MacからPC にファイルをトランスポートするときには、「Mac f」という名前のフォルダを作成します。その後、分割あるいはマージが行われたファイルがそのフォルダにコピーされます。複数ファイルをドラッグしドロップすると、最初のファイルと同じ階層にフォルダが作られます。同じ名前のフォルダがすでに存在している場合、4D Transporter では「PC.2 f」または「Mac.2 f」などという名前のフォルダを作成します。

「オリジナルファイルをトランスポート」を選択していると、4D Transporter ではフォルダを作成せずにファイルの分割やマージを行います。このオプションを使用する場合、4D Transporter では元のファイルを保存しません。そこで、4D Transporter からはこの選択に対して、次のような確認が求められます。



注：間違えてトランスポートしてしまった場合は、逆のトランスポートを行って、ファイルを元に戻すことができます。

トランスポートのオプションの選択を終えたら、「トランスポート」ボタンをクリックしてトランスポート処理を行います。

トランスポートのオプションは任意に変更できます。またファイルも必要なだけいくつでも移動させることができます。終了したら、「終了」ボタンをクリックします。

TIP：ドラッグ&ドロップやトランスポートオプションの保存、4D Transporter の非表示により、自動的にトランスポート処理を行うことができます。データベースの両方向へのトランスポートを行ったり、オプションをたびたび変更する場合には 4D Transporter のコピーを複数作成しておき、コピーごとにそれぞれ目的の処理を行います。



ファイルのタイプと 4D Transporter

次の表は、Macintosh と Windows 上で 4D Transporter によって設定されるファイルの対応表です。


Macintoshファイル	Windows のファイル拡張子	
	データフォーク用	リソースフォーク用
4D データベースの ストラクチャファイル	.4DB	.RSR
4D データベースの データファイル	.4DD	.4DR (場合によって)
Proc.Ext	不使用 (n/a)	Proc.ESR
Mac4DX フォルダに 入れられる 4D Extensions	.4DX	.RSR
Macintosh ファイルすべて *1	.DTA	.RSR (場合によって)

1. 任意タイプのファイルを開いたり、ドラッグ&ドロップしたい場合は、Command キーを押します。

次の表は、Macintosh のファイルタイプと Windows/DOS のファイル拡張子をまとめたものです。(表内の図は、バージョン3のアイコンを使用しています。)

ファイル	Macintosh ファイルタイプ	Windows ファイル拡張子	説明
ストラクチャファイル 	BASS	.4DB、.RSR	データベースのストラクチャファイルは、新しくデータベースを作成するたびに作られます。Windowsのもとでは、リソースファイル“データベース名.RSR”も作られます。この2番目のファイルには、データベースに関する Macintosh のリソースが入っています。
コンパイル後のストラクチャファイル 	BASY	.4DC、.RSR	データベースのコンパイル後のストラクチャファイルでデータベースのコンパイル時に 4D Compiler によって作られます。
データファイル 	dat5	.4DD、.4DR	データファイルは新しくデータベースを作成するたびに作られます。また、既存のデータベースを開く際に Macintosh 上では Option キー、Windows 上では ALT キーを押して新しくファイルを作ります。Windows 上で Customizer Plus を使用して WEDD リソースを作ると“データベース名.4DR”ファイルが作られます。2番目のファイルには、データファイル内の Macintosh のリソースが入っています。
データセグメント 	dax5	.4DS	データセグメントファイルは「データファイルオープン」ダイアログボックスの「分割」ボタンを押すか、または「デザイン」モードで「データセグメント」メニューコマンドを選択して作成します。
Macintosh ヘルプファイル 	4Hlp	不使用 (n/a)	Macintosh のヘルプファイルには、Macintosh のパンーンヘルプが入っています。システムの「初期設定」フォルダ内の「ACI」フォルダに置かれています。Windows 上では4D コマンドに関するヘルプはWindowsヘルプファイルで提供されています。
Windows ヘルプファイル  4d	不使用 (n/a)	.HLP	Windowsヘルプファイルには「ランゲージリファレンスマニュアル」と同じ内容が入っており、4D.EXE と同じディレクトリに置かれています。Macintosh 上では、4D コマンドに関するヘルプは、バルーンヘルプに表示されます。

ファイル	Macintosh ファイルタイプ	Windows ファイル拡張子	説明
ログファイル 	4LOG	.4DL	ログファイルは「ユーザ」モードの「ファイル」メニューから「ログファイル」コマンドを使用するか、それに相当する 4D コマンドを使用した時に作られます。
ASCII ファイル 	TEXT	.TXT	4D で作られたテキストファイルまたは、4D に ASCII テキストとして読み込めるように他のアプリケーションで作成したテキストファイル。
ワークステーション のリソースファイル 	res	.RES	ワークステーションのリソースファイルには、4D Client によってサーバからダウンロードされた Macintosh のリソースが入っています。Macintosh 上では“データベース名.res”という名前で、システムの「初期設定」フォルダ内の「ACI」フォルダに置かれています。Windows 上では、アクティブになっている Windows ディレクトリ内の「ACI」フォルダに置かれています。
ワークステーション のローカルファイル 	LOC4	.REX	ワークステーションのローカルファイルには、4D Client によってサーバからダウンロードされた 4D のデザインオブジェクトが入っています。Macintosh 上では“データベース名.rex”「ACI」フォルダに置かれています。Windows 上では、アクティブになっている Windows ディレクトリ内の「ACI」フォルダに置かれています。
Proc.Ext ファイル 	PEXT	.ESR	Macintosh の Proc.Ext ファイルには、68K 用の外部パッケージ、PPC 用の外部パッケージ (Memory Fragments)、関連リソースが入っています。このファイルは Windows 上では使われませんが、データベースを 4D Server とともに使用する場合は存在することがあり、データベースのストラクチャファイルと同じ階層に置く必要があります。
4D Extensions (Macintosh) 	4DPX	.4DX	4D の Macintosh Extension は「Mac4DX」フォルダ (ストラクチャと同じ階層に置かれています) に配置します。さらに、Windows のもとでは、同じファイル名で、拡張子が .RSR になっている関連ファイルがあります。この 2 番目のファイルには、4D Extension の Macintosh のリソースが入っています。これらのファイルは Windows 上では使われませんが、データベースを 4D Server とともに使用する場合は存在することがあります。

ファイル	Macintosh ファイルタイプ	Windows ファイル拡張子	説明
4D Extensions (Windows)	不使用 (n/a)	.4DX	4D のWindows Extension (DLL) は「 Mac4DX 」フォルダ (ストラクチャと同じ階層に置かれています) に配置します。さらに、Windows のもとでは、同じファイル名で、拡張子が .RSR になっている関連ファイルがあります。この 2 番目のファイルには、4D Extension の Macintosh のリソースが入っています。これらのファイルは Macintosh 上では使われませんが、データベースを 4D Server とともに使用する場合は存在することがあります。
リソースフォーク すべて	不使用 (n/a)	.RSR	その他のファイルの Macintosh リソースファイルが入っています。
Hot link ファイル 	4DHL	.4LK	Hot link ファイルは 4D で作られ、稼働中のセッションとの間のホットリンクが保存されています。

他の4D文書のファイルタイプと対応するWindowsのファイル拡張子を次に示します。

ファイル	Macintosh	Windows
クイックレポート文書	4DSE	.4QR
ラベル文書	4DET	.4LB
SAVE SET 文書	SETT	.4ST
SAVE VARIABLE 文書	VAR1	.4VR
ユーザとグループ保存	4DUG	.4UG
パス名文書	paTH	.PTH
ASCII フィルタ文書	FILT	.4FI
Apply formula 文書	EFRM	.4FR
検索文書	RECH	.4DF
フォーミュラ (検索、ソート...)	TFRM	.TFR

注 : Windows の .4DB と .RSR ファイルを Macintosh のストラクチャファイルにマージするには、.4DB だけを 4D Transporter アイコンにドラッグします。プログラムでは .RSR が同じ階層にあれば、自動的に操作に組み入れます。.4DD と .4DR ファイル、.DTA と .RSR ファイルにも同じ方法が取られます。

4D First Converter

バージョン1の4D Firstで作成されたデータベースをバージョン3の4th Dimension データベースに変換するためのユーティリティです。

注：ストラクチャファイルはコピーされません。4D Firstのストラクチャファイルを保存しておきたい場合は、4D First Converterを使用する前にそのストラクチャファイルのコピーを作成しておく必要があります。

バージョン6の4D Firstで作成されたストラクチャファイルは、バージョン6の4th Dimensionに直接変換できるので、このユーティリティを使用する必要はありません。バージョン1の4D Firstで作成したストラクチャファイルをいったんバージョン3の4th Dimensionデータベースに変換し、それをバージョン6の4th Dimensionで使用する場合にこのユーティリティを使用します。

4D Firstデータベースを4Dデータベースに変換するには、次のように行います：



1. 4D First Converter アプリケーションのアイコンをダブルクリックして起動する。変換してよいかどうかを確認するダイアログボックスが表示されます。

2. 「OK」ボタンをクリックする。

下図のような「ストラクチャファイル選択」ダイアログボックスが表示されます。



3. 変換する 4D First ストラクチャファイルを選択し、「開く」ボタンをクリックする。4D Firstのデータベースが、バージョン3の4th Dimensionデータベースに変換されます。

External Mover Plus

これは、4th Dimension (4D First) のデータベース内に外部パッケージを組み込むためのユーティリティです。ただし、バージョン6の4th Dimension (4D First) で使用することはほとんどありません。これは、旧バージョンの4th Dimension (4D First) と互換性を保つために同封されています。