

ニンテンドーDS、Wii向けゲーム開発者の ための18の秘技

2007年9月26日
(株)CRI・ミドルウェア

映像・音声を専門としたミドルウェア開発会社

- | | | |
|-------|--|-------------------|
| 1990年 | 人工知能・CD・音声・映像技術の研究開発
FM-Towns・メガCDなど | |
| 1993年 | サターン用CDシステムの開発 | |
| 1995年 | サターン用ADXのリリース | 50タイトル |
| 1997年 | Dreamcast用ADX・Sofdecリリース | 450タイトル |
| 2007年 | DS・Wii・GC・PS2・PS3・PSP
Xbox・Xbox360用ミドルウェア | 1200タイトル以上 |

ゲーム機に特化したミドルウェア

はじめに

■プラットフォーム固有情報について

- DSやWiiに関するプラットフォーム固有の計測値については、本セッションについては、お話することはできません。
- 具体的な値については、弊社のテクニカルサポートサイトにアクセスしていただくか、直接、お問い合わせください。

CRIWARE のご紹介

■ 『救声主』 (DS)

音声データの圧縮。512MビットのROMに9時間収録可能。

■ 『ファイルマジック』 (DS、Wii)

ゲームデータの圧縮。読み込み時間の短縮。

■ 『CRI Audio Daisy for Wii』 & 『CRI ADX』

サウンドのオーサリング。高性能なストリーミング再生。

■ 『シネマスタジオ』 & 『CRI Sofdec』

高画質な動画再生。ワイドTV、字幕、 α ムービーなどに対応。

アジェンダ

■ サウンド (DS)

1. 音声データを小さくするワザ
2. 音を綺麗にするワザ

■ データ圧縮・ファイルシステム (DS)

3. データ圧縮技術を使いこなすワザ
4. 大量のファイルを管理するワザ

■ ストリーミング (DS)

5. 音声再生中にデータをロードするワザ

■ ネットワーク (DS)

6. ネットワークを活用するワザ

■ 音声入力 (DS)

7. マイクを活用するワザ

■ サウンド (Wii)

8. リモコンの音を綺麗にするワザ
9. ループ音を綺麗につなげるワザ
10. 音の距離をうまく演出するワザ
11. リアルなエンジン音を作るワザ
12. 高度なランダムサウンドを作るワザ

■ ストリーミング (Wii)

13. 高速にデータを読み込むワザ
14. シーク音を少なくするワザ

■ 動画 (Wii)

15. ムービのファイルサイズを節約するワザ
16. ワイドテレビ用ムービエンコードのワザ
17. 多言語対応ムービ作成のワザ
18. 演出ムービのワザ


DS: サウンド関連の秘技

音声の圧縮

■ ゲーム内の大量音声といえばセリフ (肉声)

- DSではユーザー層もひろがり、音声によるインタフェースは重要。

■ MP3などの問題点

- 低ビットレートではフレームが切り替わる時にビット割り当てが変化する。
→ ノイズが発生し、聞き取りにくい。 

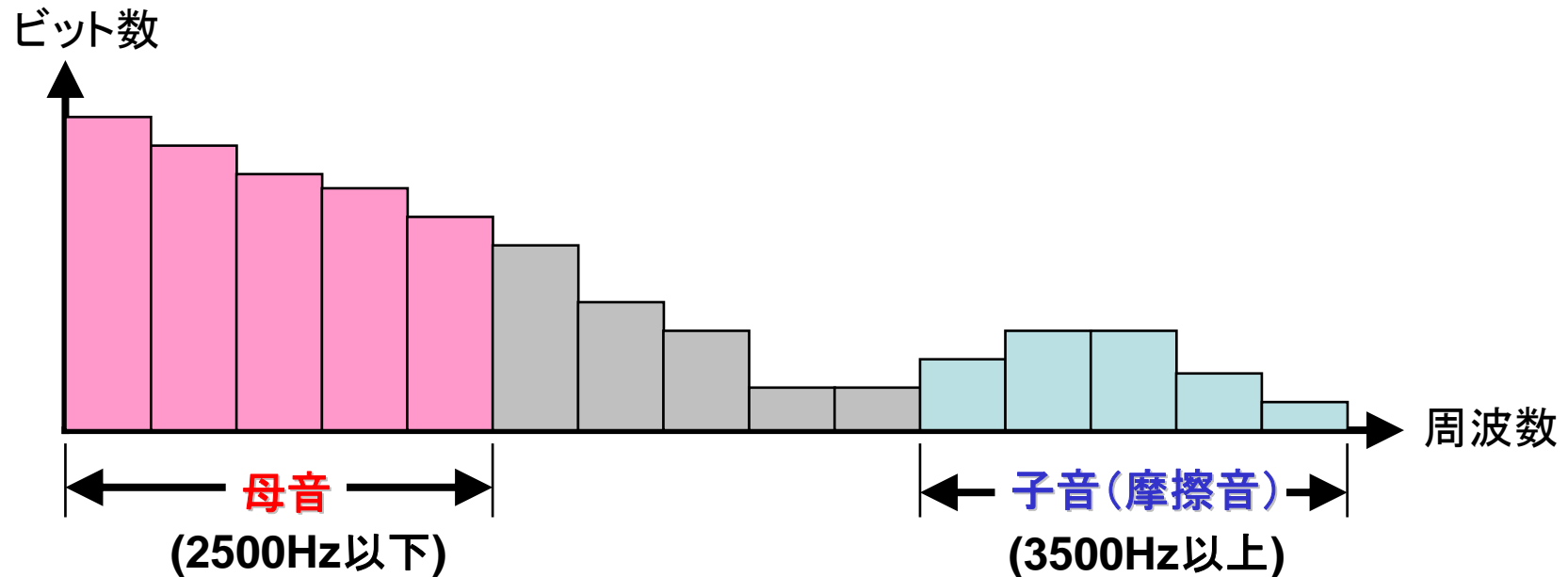
■ 『救声主』の圧縮方法

- 肉声の持つ特徴を利用してより圧縮する。

『救声主』の圧縮方法

■ 32帯域のサブバンド圧縮

- 16kHzのサンプリングレートで15%程度のCPU負荷。
- 母音の音韻情報のある周波数帯と子音(摩擦音)の周波数帯に着目。



『救声主』のデモンストレーション

■ 救声主の音質比較

- 512Mビット(64Kバイト)に収録できる長さを基準に比較。
- 『救声主』では9時間まで収録可能。
- エンコーダは、最適なサンプリング周波数とビット割り当てを決定。

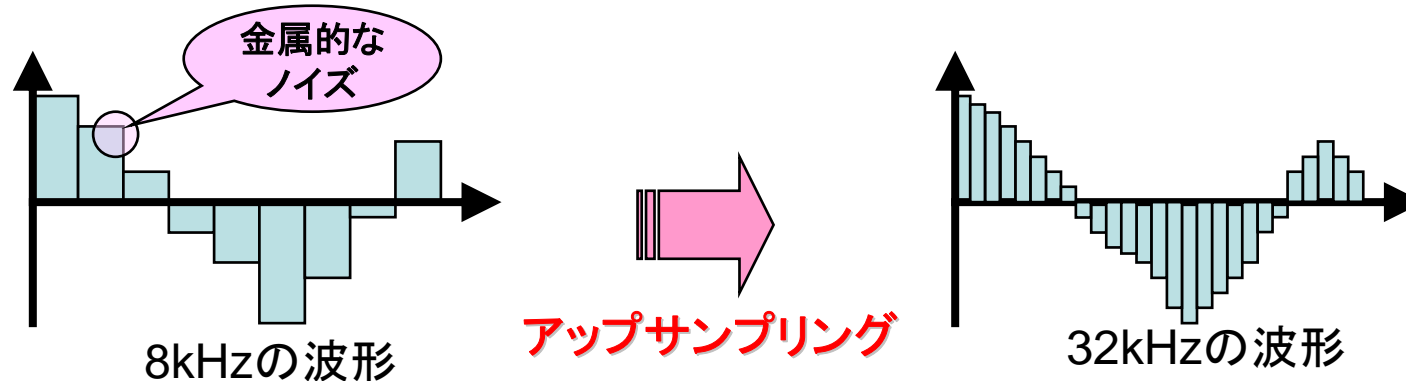
アップサンプリングによる高音質化

■ オーバーサンプリング技術によるCDの劇的な音質向上

- D/A変換後のアナログフィルタ回路が音質を決定。
- 初期のCDプレーヤーは高価なフィルタ回路により40万円ぐらい。
- 最近のCDプレーヤーは、サンプリングレートを上げてからD/A変換。

■ アップサンプリング処理によってDSの音質を改善

- アップサンプリング処理によってアンチエイリアシングノイズが軽減。

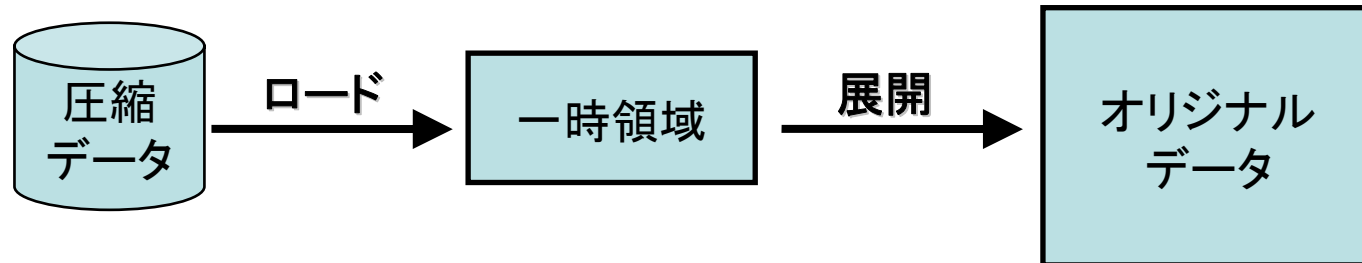


DS: データ圧縮関連の秘技

ゲームデータを圧縮するときの問題点

- **圧縮してみないとサイズがわからない**
 - 圧縮できない場合もある。(大きくなってしまうことも...)
- **圧縮したデータを展開するための一時バッファが必要**
 - 圧縮データを一時バッファにロード → 目的の領域へ展開。
- **事前に展開後のサイズを知りたい**
 - データ領域を何バイト確保するかをロードする前に知りたい。

一般的な圧縮データの展開方法



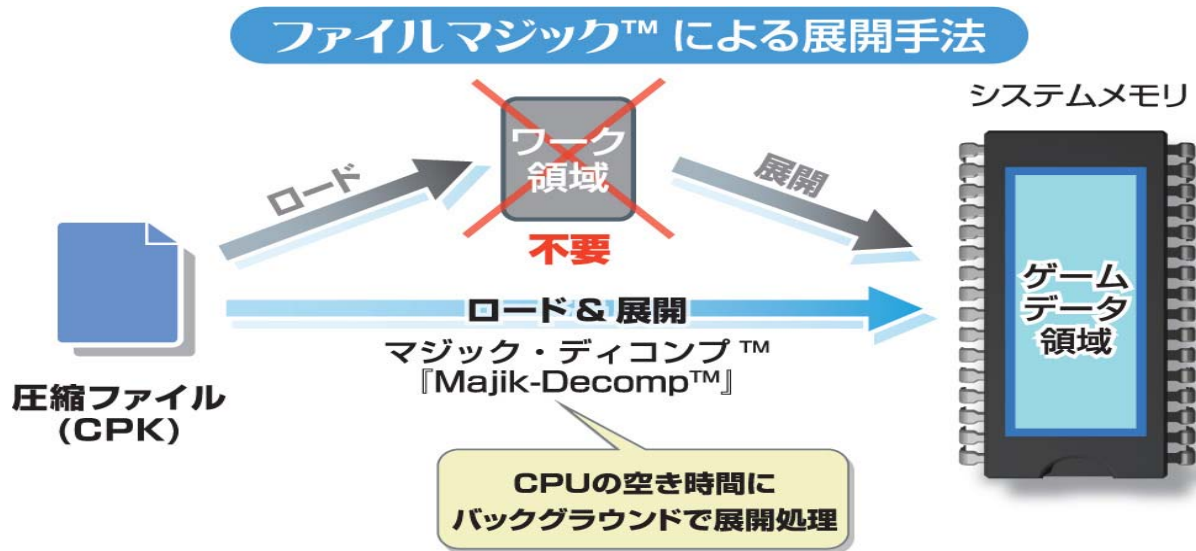
『ファイルマジック』による圧縮・展開

■ 展開時に一時バッファを必要としない『自己エリア展開』

- 展開用の一時バッファを使用せずに、ロードされた領域で展開。

■ プログラマは圧縮されているかどうかを意識せずにロード

- Load関数を呼ぶだけで、圧縮されているデータは自動的に展開される。



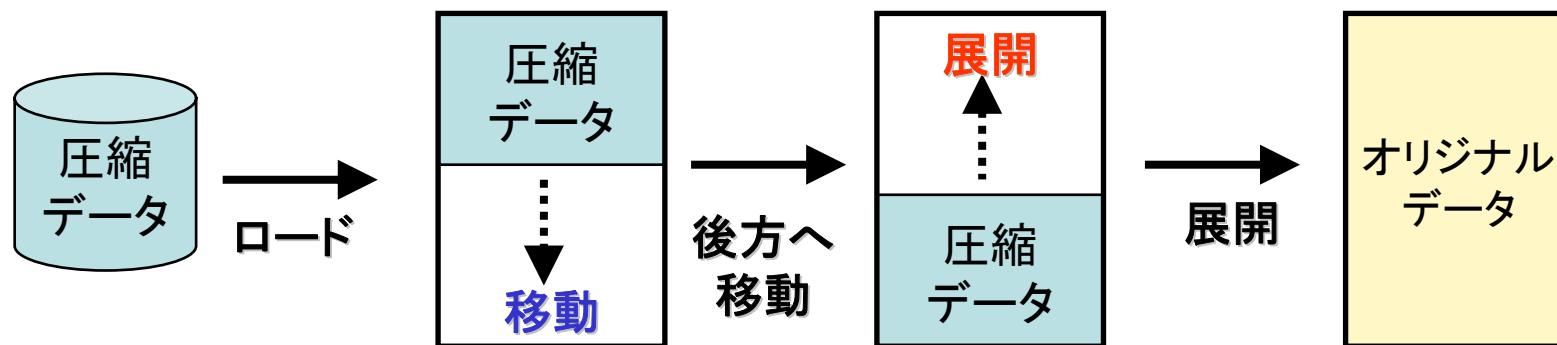
自己エリア展開を実装しようとする

■ 一般的な圧縮手法はファイルの先頭から圧縮

- ロード後、データを領域の後方に移動する必要あり。
- また移動先に直接ロードしようとする则アライメントが問題になる。

■ 展開データが圧縮データを壊すことが...

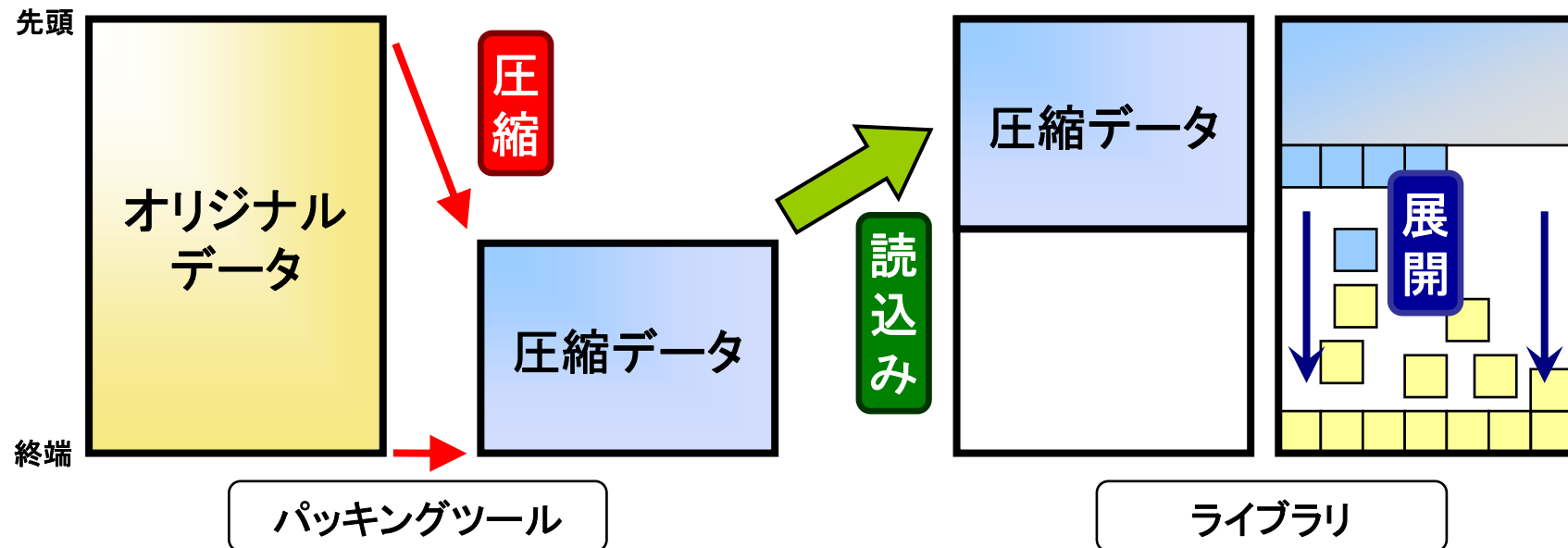
- エンコード時に展開できるかチェックする必要がある。



マジック・ディコンプ™のアルゴリズム

■ ファイル後方より圧縮・展開

- ファイルデータの終端から圧縮。ロードしたデータは後方から展開。
- 高速な『自己エリア展開』を実現。



かしこいパッキング

■ ディレクトリ情報の工夫

- 圧縮データサイズと展開後のデータサイズを持つ。
- 64Kバイト以内のファイルとそれ以上のファイルの管理テーブルを分ける。

■ 「圧縮可能か」「自己エリア展開可能か」の事前チェック

- 圧縮して小さくなるか、展開可能かをチェックしながらパッキング。

<64Kバイト以下のファイル>

ファイルID	圧縮されて サイズ	オリジナル サイズ

← 6バイト →

<64Kバイトより大きいファイル>

ファイルID	圧縮サイズ	オリジナルサイズ

← 10バイト →

DS: ストリーミング関連の秘技

音声ストリーミング中のデータロード

■ ROM容量の増加 → ファイルシステムが必要

- DMA もDVDドライブのピックアップと同じ
 - 複数のファイルに同時にはアクセスできないため調停が必要
 - **ストリーミング技術**

■ 音声ストリーミングの隙間を利用してデータを読み込む

- 『救声主』+『ファイルマジック』では音が途切れないように自動的に流量をコントロール。
- 通常のスリーム再生中でも『ファイルマジック』を使用すると、途切れないように読み込み量を調節可能。

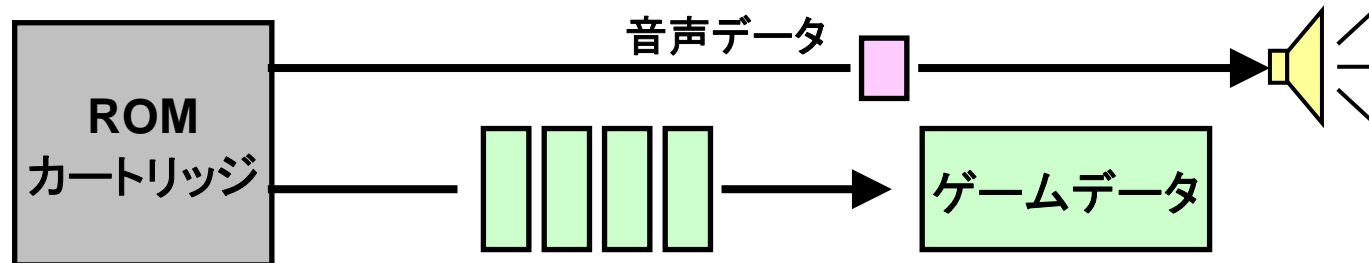
音声ストリーミングの仕組み

■ 読み込みプライオリティコントロール

- 音声データの読み込み > ゲームデータの読み込み

■ ゲームデータの読み込み単位(ユニット)をコントロール

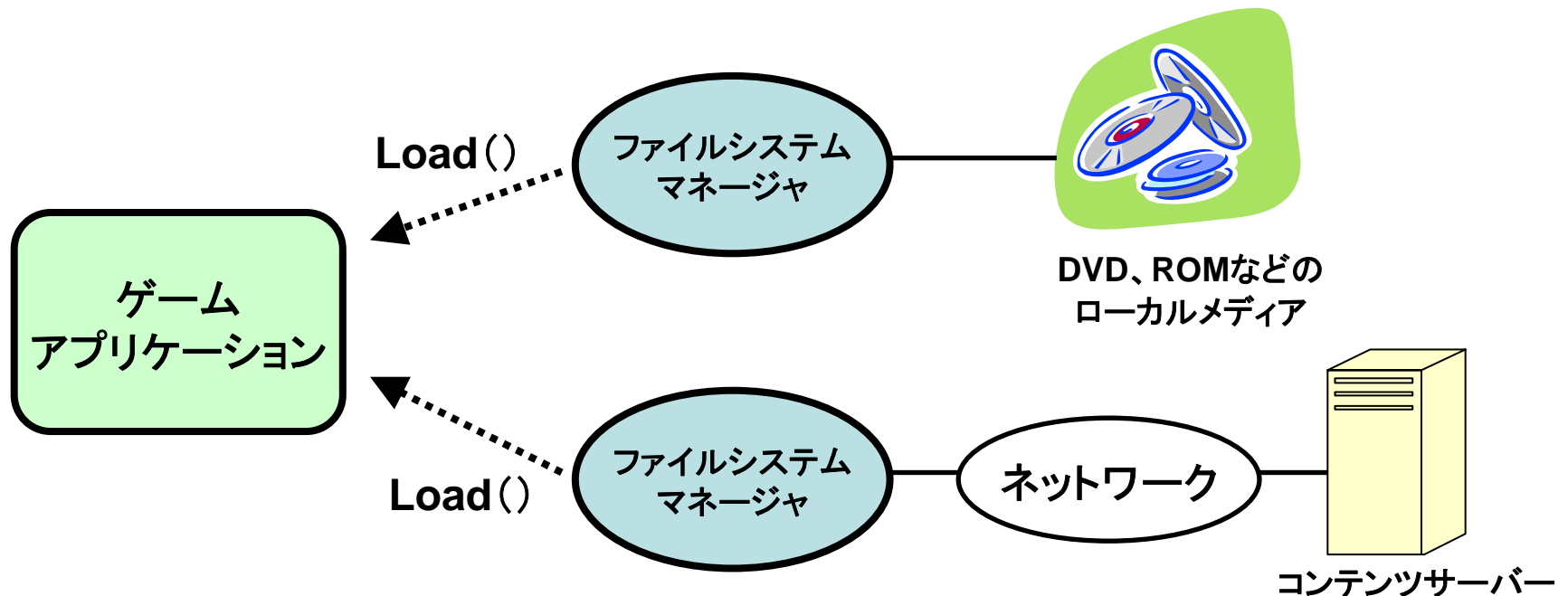
- ゲームデータを複数のユニット分割して読み込む。
- ユニットを読み込むが終了するたびに、音声データ読み込みとのプライオリティを再評価。
- ユニットサイズは、ROMタイプによって変更する。



DS: ネットワーク関連の秘技

『ファイルマジック』によるネットワークファイルシステム

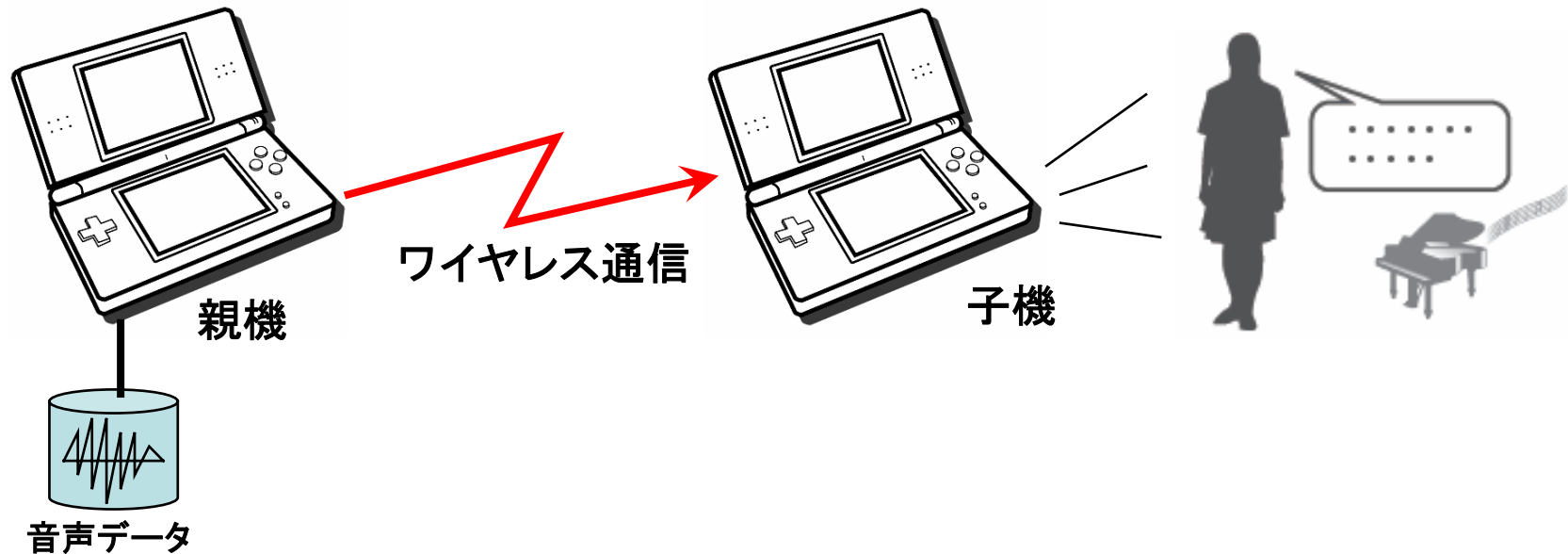
- ネットワークを介して別のマシン上のファイルをアクセス
 - ゲームプログラムは、ほとんど変更せずにネットワークに対応。



ネットワークストリーミング再生のデモンストレーション

■ 親機のカートリッジ内の音声データを子機で直接再生

- 『救声主』によって16kHzの音声(24kbps程度)を直接ストリーミング再生。
- Wii 上のファイルを再生したりすることも可能に...

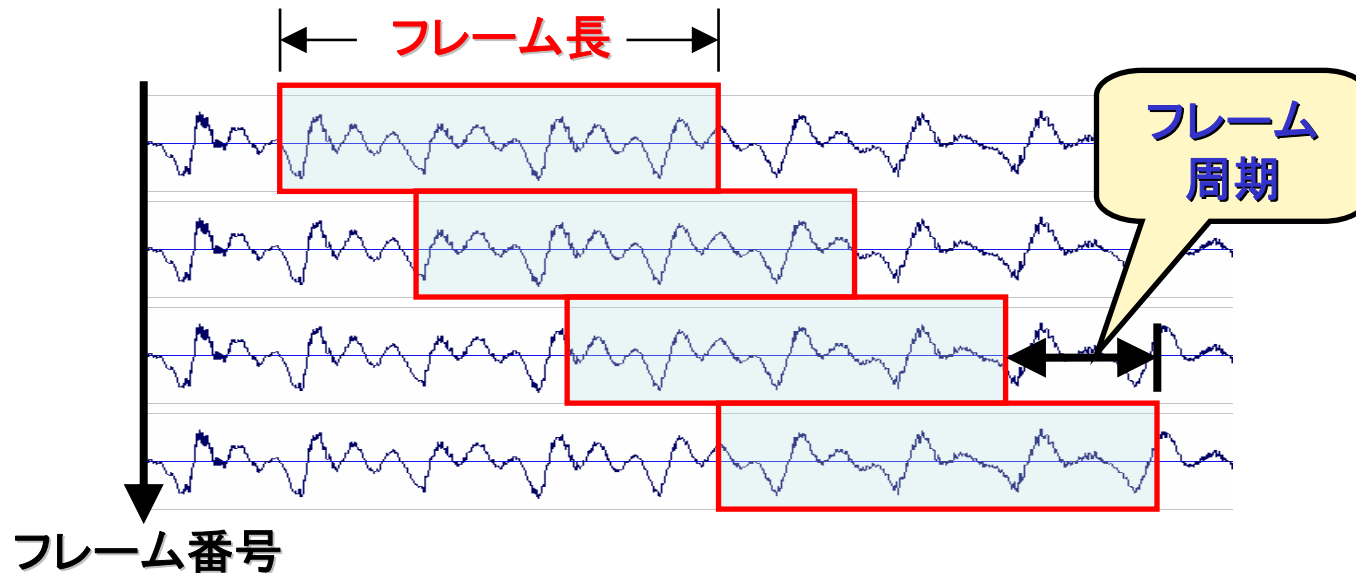


DS: マイク関連の秘技

音声解析の基本

■ フレーム長とフレーム周期

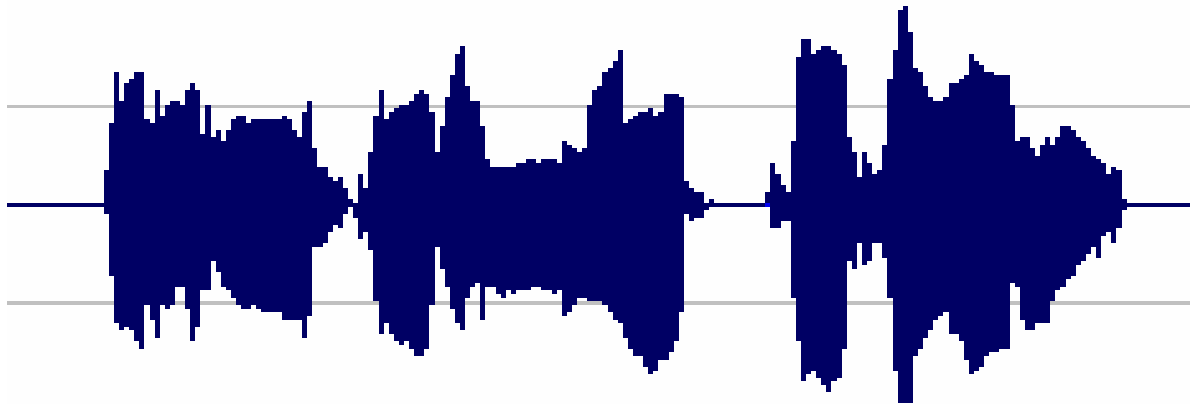
- フレーム長 → 分析したい音声の最低周波数が決まる。
- フレーム周期 → 1秒間に何回分析するか → 分析精度



振幅の抽出

■ 振幅の抽出

- 各フレームごとの最大振幅を検査し表示。
- フレーム長の取り方がキーポイント。
- フレーム長を16.6msecぐらい(60Hz相当)に設定すると低い声の人でも綺麗に振幅が抽出できる。



低負荷なピッチ（音程）抽出

■ AMDF法

- 自己相関やFFTを用いるのが一般的。→ CPU負荷が大きい。
- AMDFは、減算と加算のみでピッチ（音程）抽出が可能。
- 4msec程度とCPU負荷が小さい。

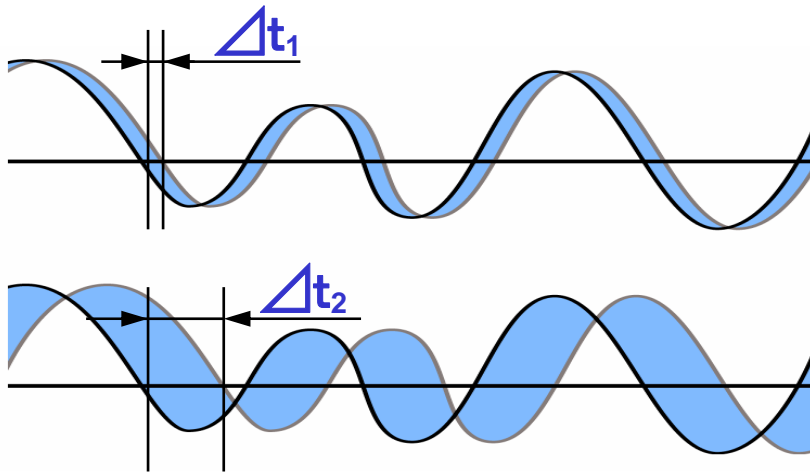
■ 応用例

- 英語のイントネーションの抽出。
- 鼻歌から楽譜に変換。

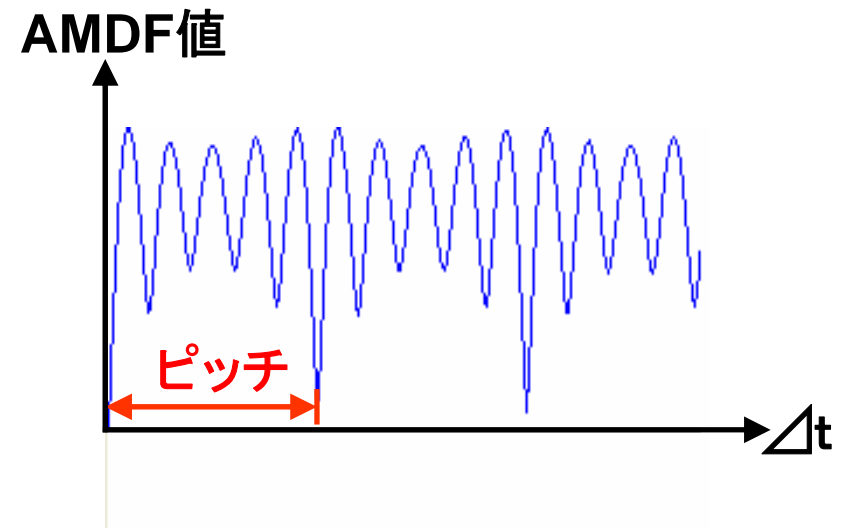
AMDFによるピッチ抽出法

■ AMDF

- 信号のずらしながら、差分の絶対値の総和を計算。
 - ・ 完全に一致 AMDF値=0
 - ・ 少しずつずらすと段々大きくなり、1周期分にずれると小さくなる。



※ AMDF値は上記の水色の部分の面積



振幅抽出・ピッチ抽出のデモンストレーション

■ DSのマイクより12kHzでサンプリング

- フレーム長を50Hzで振幅を抽出。
- 25%程度のCPU負荷でリアルタイムにピッチ(音程)を抽出。
- 周波数から音階名("ドレミ"に相当するアルファベット)に変換。


ワザ Wii: サウンド関連の秘技

リサンプリングによるリモコンの音質の改善


- **リモコンの最大サンプリング周波数にあわせてリサンプル**
 - 高い周波数のままコンプレッサをかけるとノイズが増加。
- **リサンプリングによって欠落する高周波成分を補強**
 - イコライザやキューのあるフィルタを通して、高域を持ち上げてからリサンプリングする。
- **音声の場合は、摩擦音を残すようにイコライジング**
 - 2.5～3kHzの摩擦音の成分を強調すると聞き取りやすくなる。

リモコンからの音声再生のデモンストレーション


■ リモコンからの音声の比較 「歴史と運河の町、ユトレヒト。」

(1) オリジナルの音声 


聞き取り難く、ノイズが発生している。

(2) コンプレッサー処理した音声 


アンチエリアシングノイズが目立つ。

(3) 6kHzにリサンプリングした音声 

アンチエリアシングノイズは減少するが、子音が聞き取りにくい。

(4) 2-3kHzを4デシベル程度ブーストした音声 

だいぶ音質が改善され、ゲームにとってリーズナブル。

(5) 32kHzにアップサンプリングし、非圧縮で再生した音声 (最高品質) 

ループ音を綺麗につなげる

■ ステレオのループ

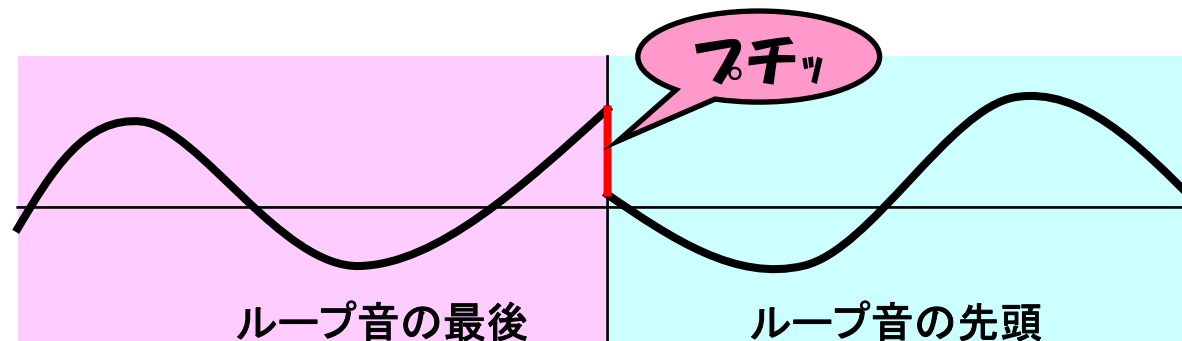
- ステレオのループ音声は接続部にノイズが入りやすい。

■ インタロクロスフェード法

- インタロ部の最後とループエンドの直前をクロスフェードさせる。

■ 強制接続法

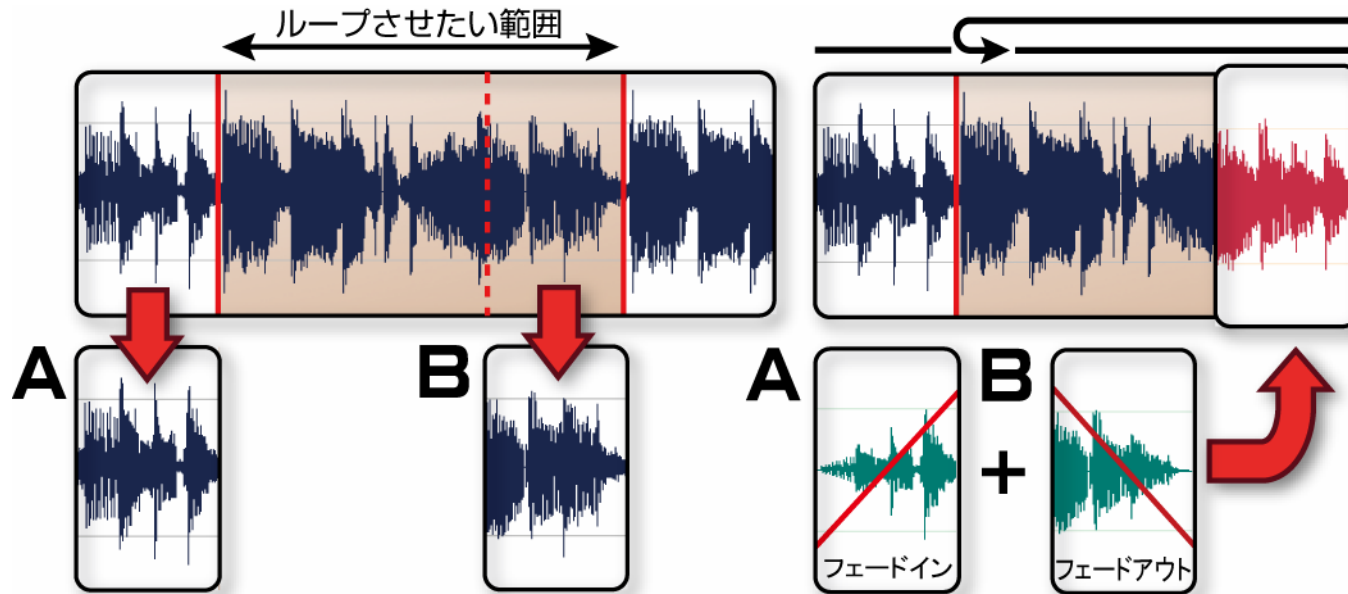
- 強制的にループの最初と最後を接続する。



イントロクロスフェード法

■ イントロクロスフェード法

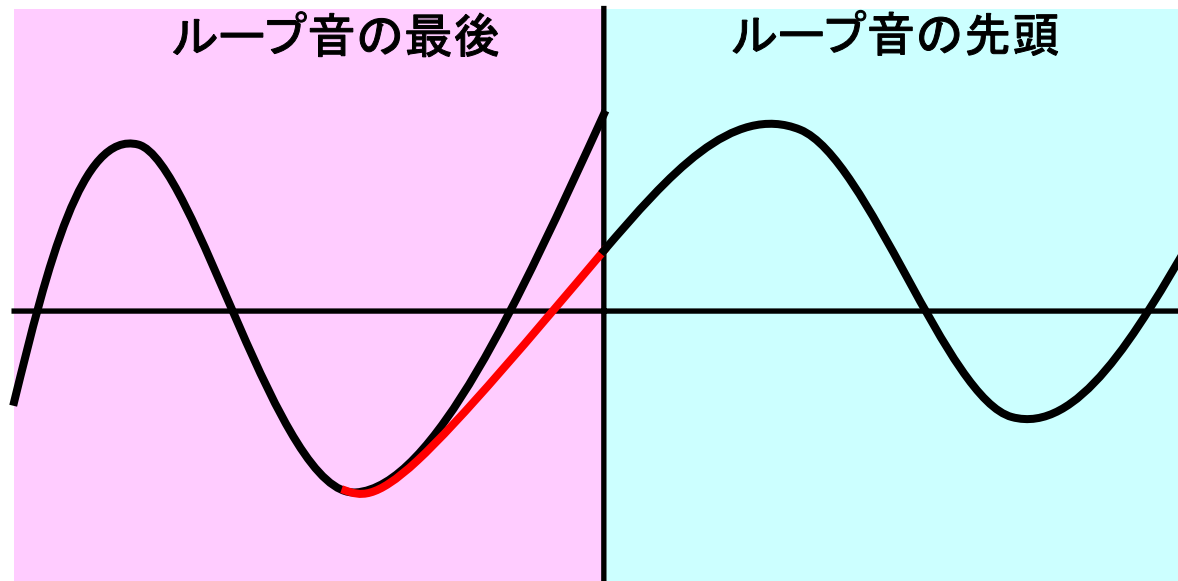
- イントロ部の最後とループエンドの直前をクロスフェードさせる。



強制接続法

■ 強制接続法

- 段差のできている信号を強制的に接続する。
- 音が歪んでしまうが、結構、許容できることも多い。



効果音の役割に応じた距離減衰

■ 効果音にはそれぞれの役割あった演出が必要

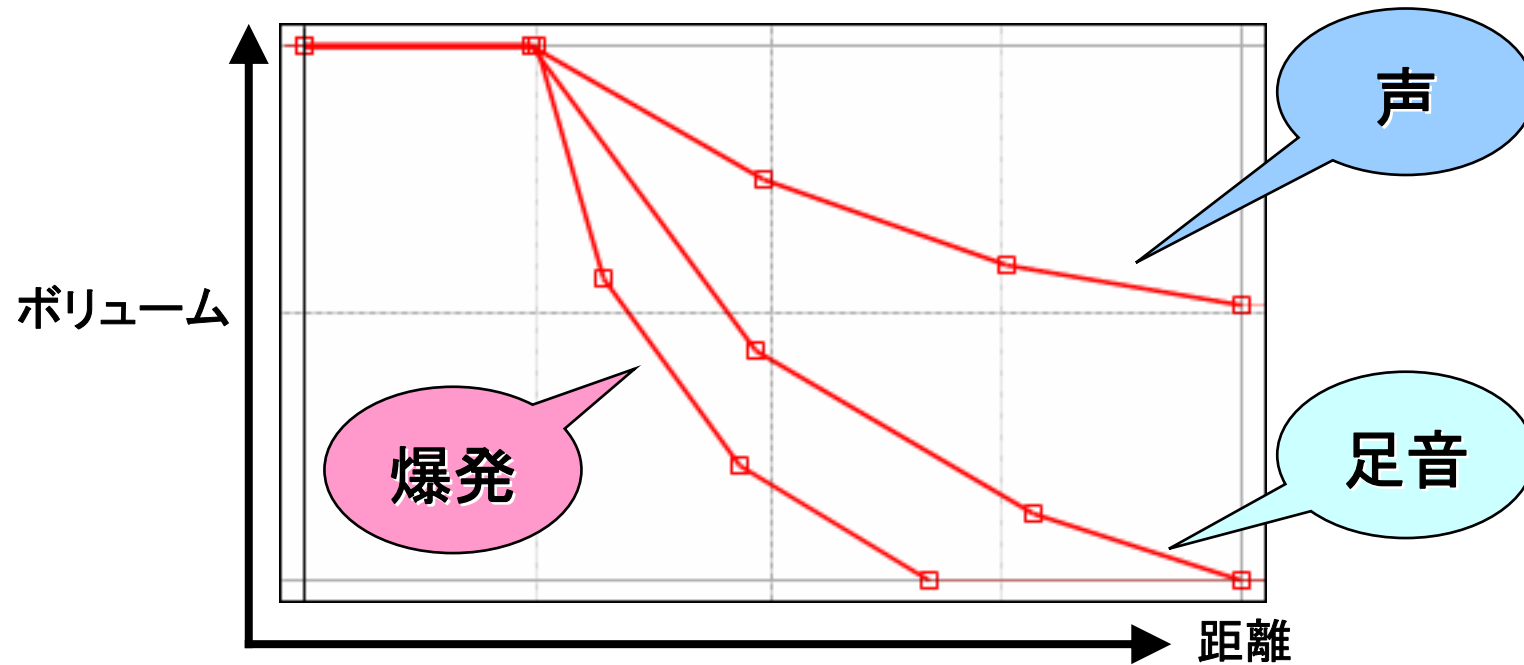
- セリフは常に聞こえてほしい。
- 足音は動きにリアリティを出すために重要。
- FPSにおける足音は位置情報を知るためさらに重要。
- 近距離の爆発音は重要だが、遠くなったらすぐに消えてほしい。
- 距離が遠くなったら残響音を深くかけたい。

■ 距離減衰のカーブを各効果音に設定する

- 各効果音のカテゴリごとに距離減衰カーブをアタッチできる。

効果音の役割に応じた距離減衰

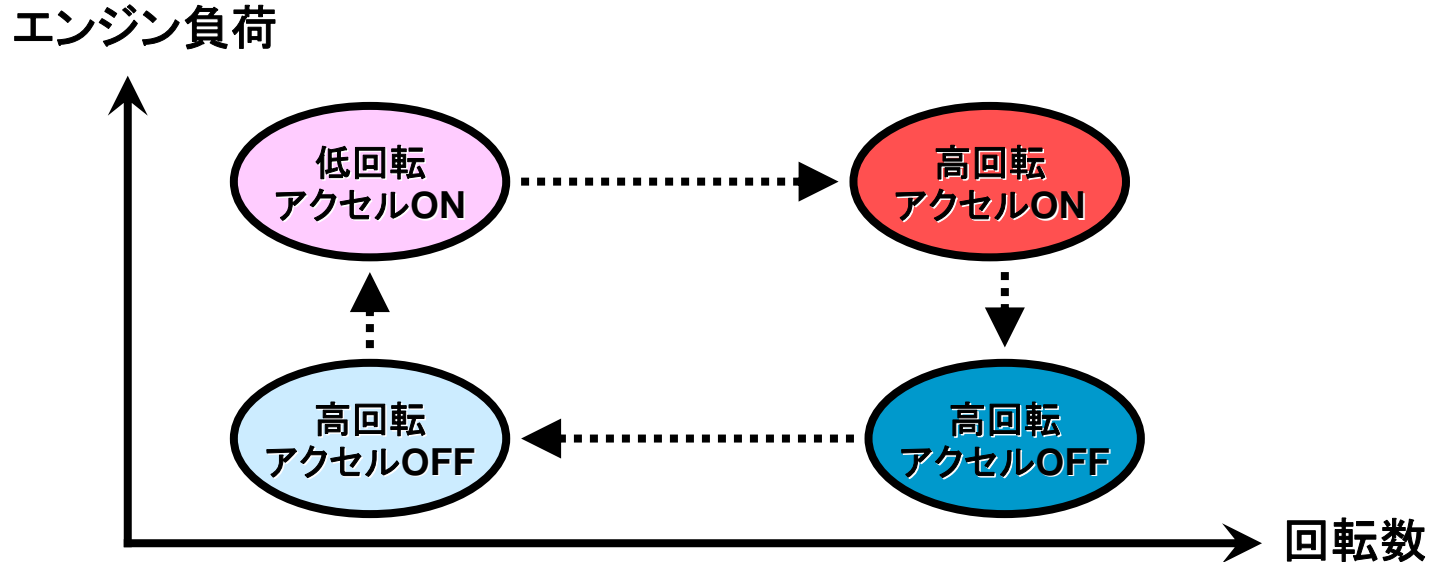
■ CRI Audio では効果音タイプ別に距離減衰カーブを設定



CRI Audioによるエンジン音のデモンストレーション

■ 実際のアーケードゲームに使用されているエンジン音

- 株式会社セガ AM3研のご好意によりご提供いただきました。
- トヨタ・カローラレビン AE85型モデルのエンジン音。



リアルなエンジン音の作り方

■ いくつかのエンジンを回転数にあわせてクロスフェード

- いくつかの回転数のエンジンをサンプリングし、クロスフェードさせる。

■ 加速中と減速中の音を用意

- 加速中はターボ音。減速中はエンジnbrレーキ音。

■ プログラマは車の状態に合わせてコントロール

- エンジンの回転数とアクセルの状態によってコントロール。

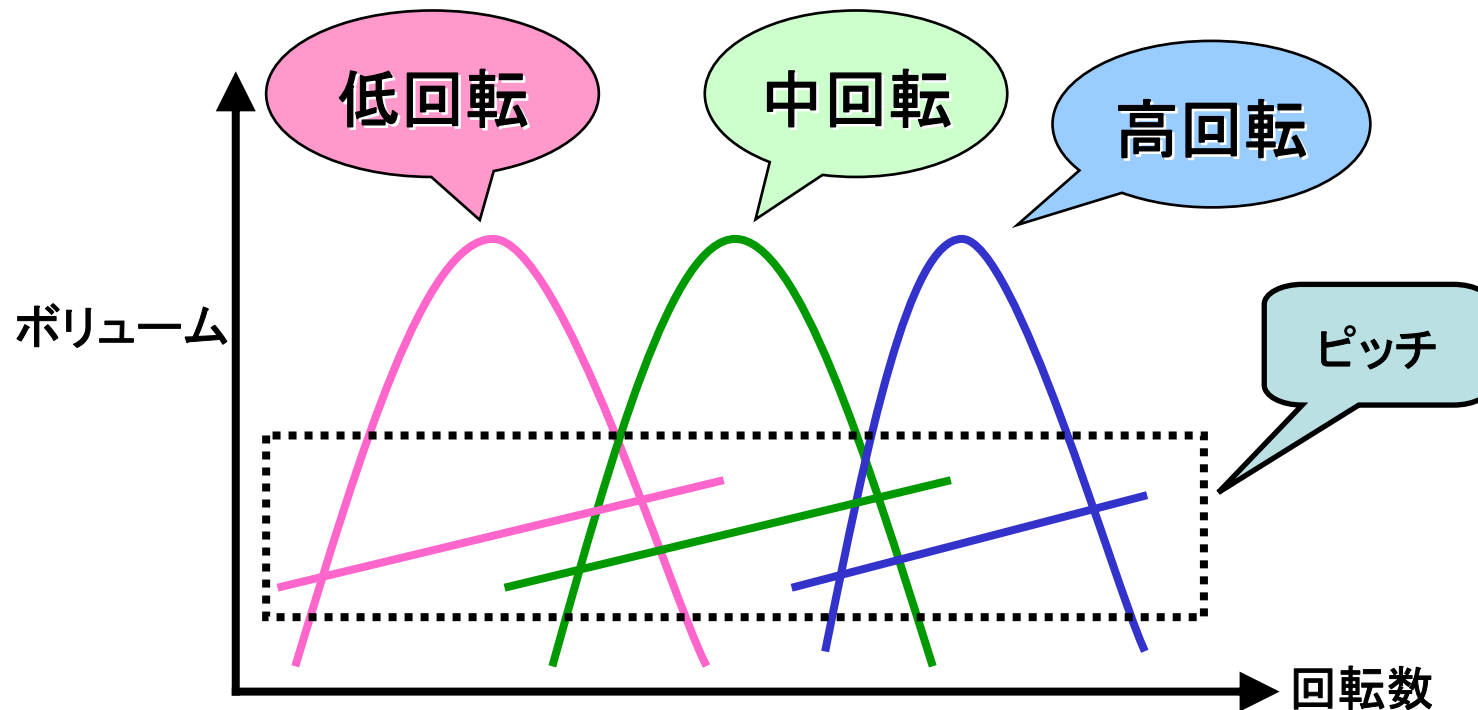
■ さらに距離による音の変化も演出する

- 距離によってフィルタを適用することにより臨場感が向上。

リアルなエンジン音の作り方 (基本)

■ 複数のエンジン音をクロスフェード

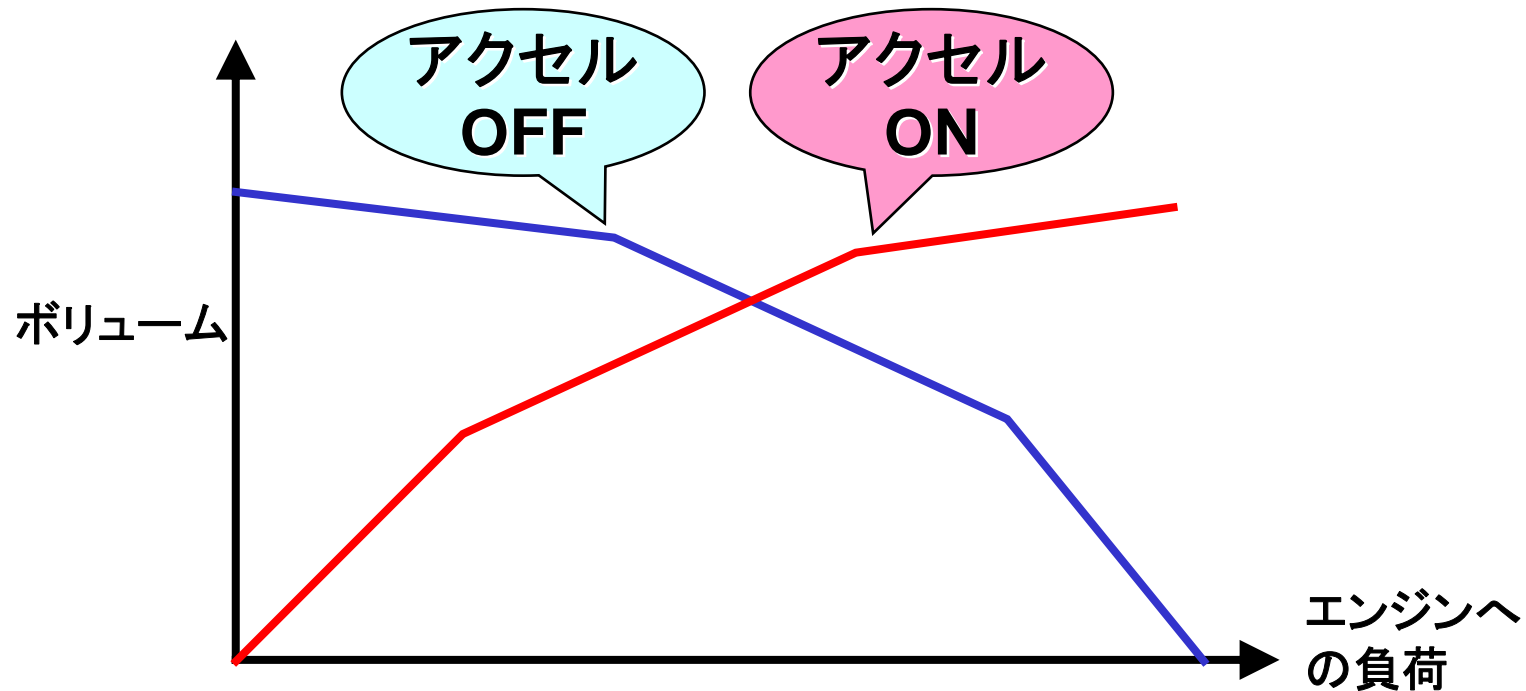
- 回転数の異なるエンジン音をサンプリングし、クロスフェードさせる。



リアルなエンジン音の作り方 (応用)

■ エンジンの負荷状況による変化

- アクセルONの音とアクセルOFFの音をクロスフェードさせる。



高度なランダムサウンド

■ リアルな効果音には確率分布のコントロールが必要

- 常に変化するランダムサウンドは自然界にはあまりない。
- たまに大きくかわったりすることでよりリアルな効果音を実現。

■ いくつかのパラメータの相関関係を持ったランダムサウンド

- ランダムにボリュームやピッチが変化するとき、相関関係をもっていると感じのいいランダムサウンドになる。

■ CRI Audio によるデモンストレーション

- 足音、雷鳴、銃声、カリンバなど

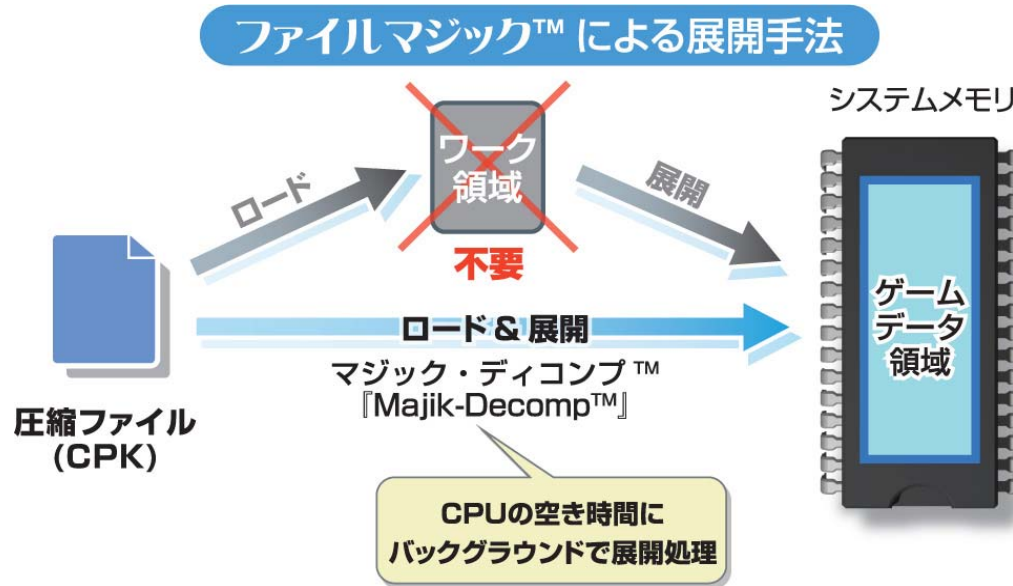


Wii: ストリーミング関連の秘技

データ圧縮による読み込みの高速化『ファイルマジック』

■ データを圧縮することにより読み込み時間を短縮

- 展開時間はDVDからの読み込み時間に比べて非常に小さい。
- 『ファイルマジック』は、ZLIBの2倍から3倍程度高速。



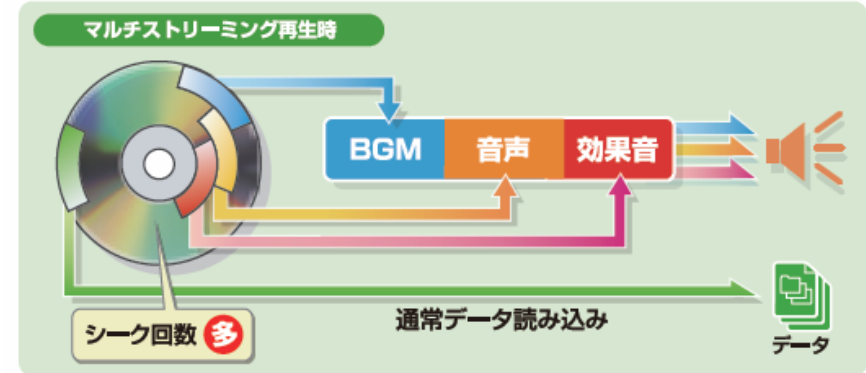
データ圧縮の効用

1. ファイルサイズの削減
2. 読み込み時間の短縮

ストリーム再生中のデータ読み込みの高速化 『CRI Audio』

■ D-BAS™ (ダイナミックバッファアロケーションシステム)

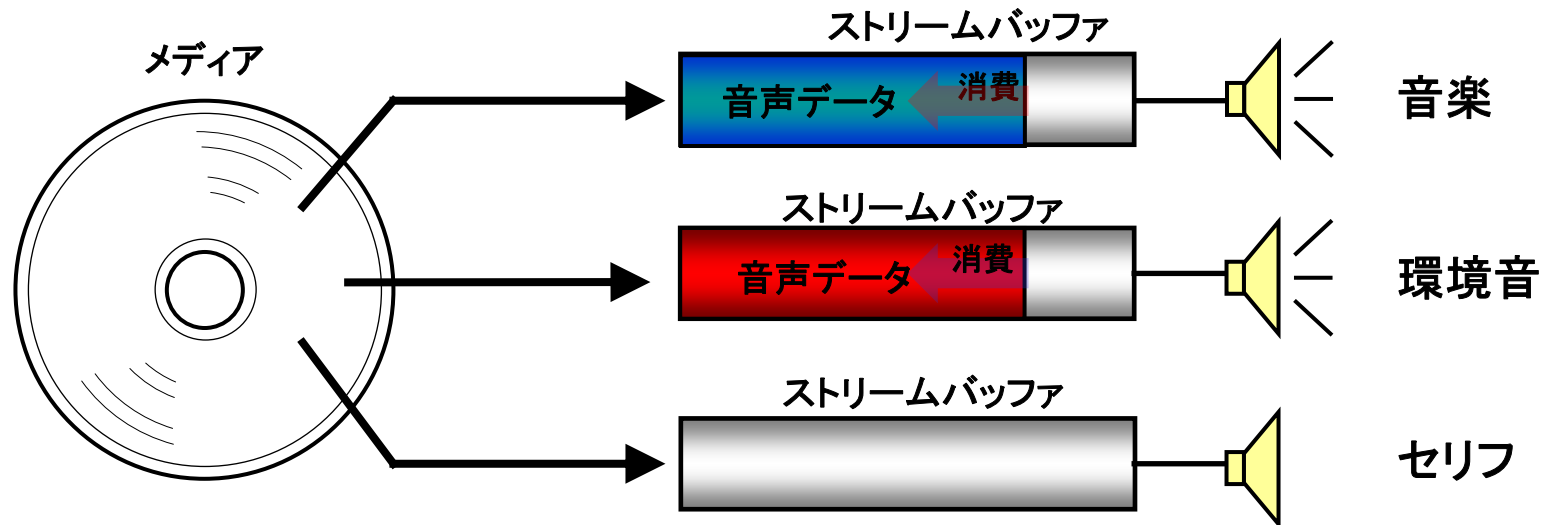
- CRI Audioのストリーミング再生に使われているバッファ管理技術。
- 事前に与えられたストリーミング用バッファを状況に応じて効率的に利用。
- ストリーミング再生中のデータ読み込みの高速化に寄与。



従来のストリーミング手法

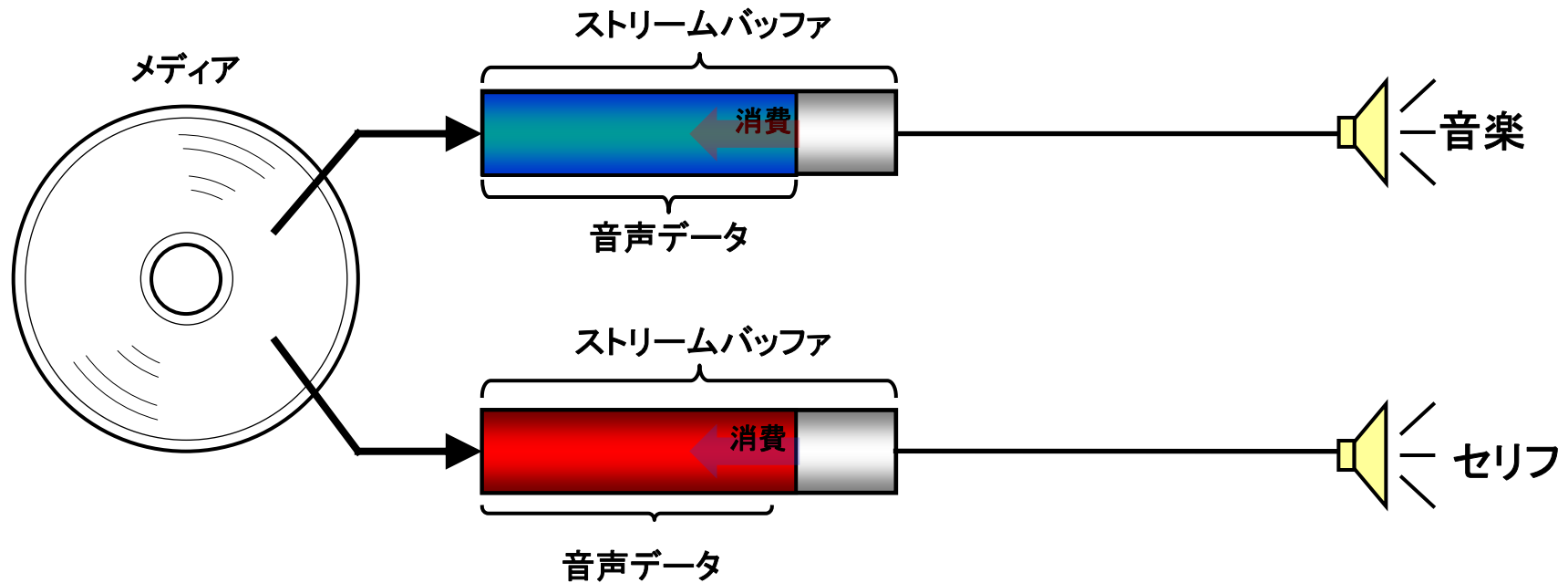
■ 同時に再生するストリーム数分のバッファを静的に確保

- データロード中に音楽しか再生していないと、他のバッファは無駄になる。
- すべてのストリームバッファを有効に利用したい。 → **D-BAS**



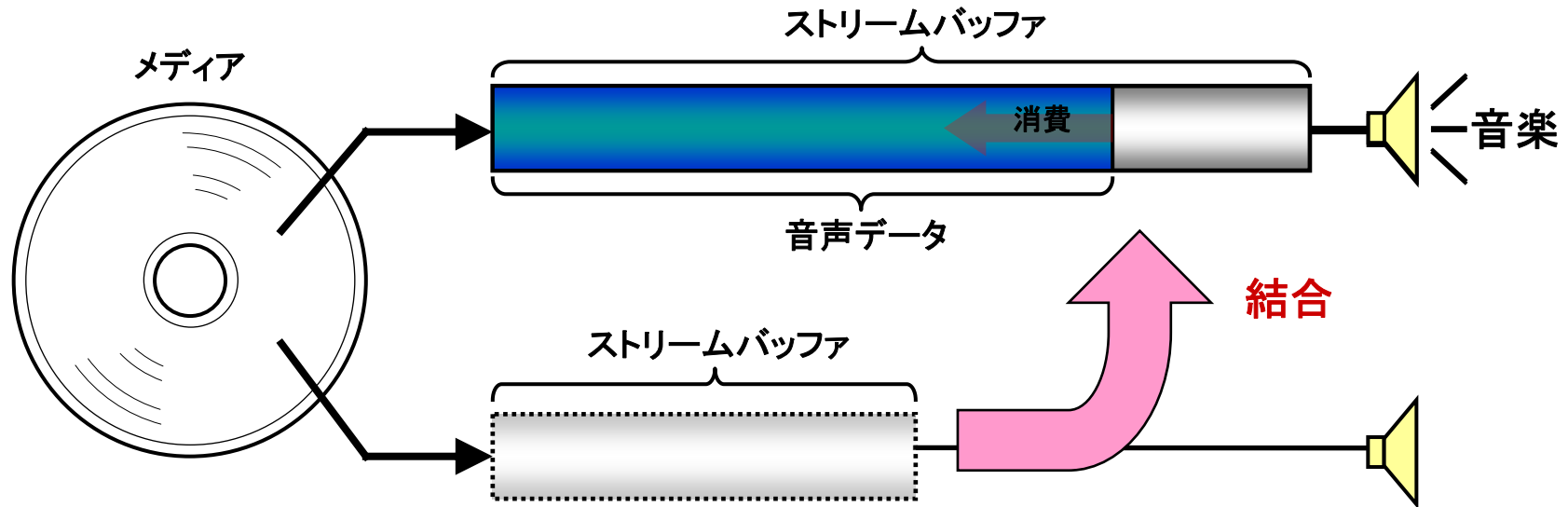
D-BAS™ の仕組み1

■ 音楽とセリフを再生



D-BAS™ の仕組み2

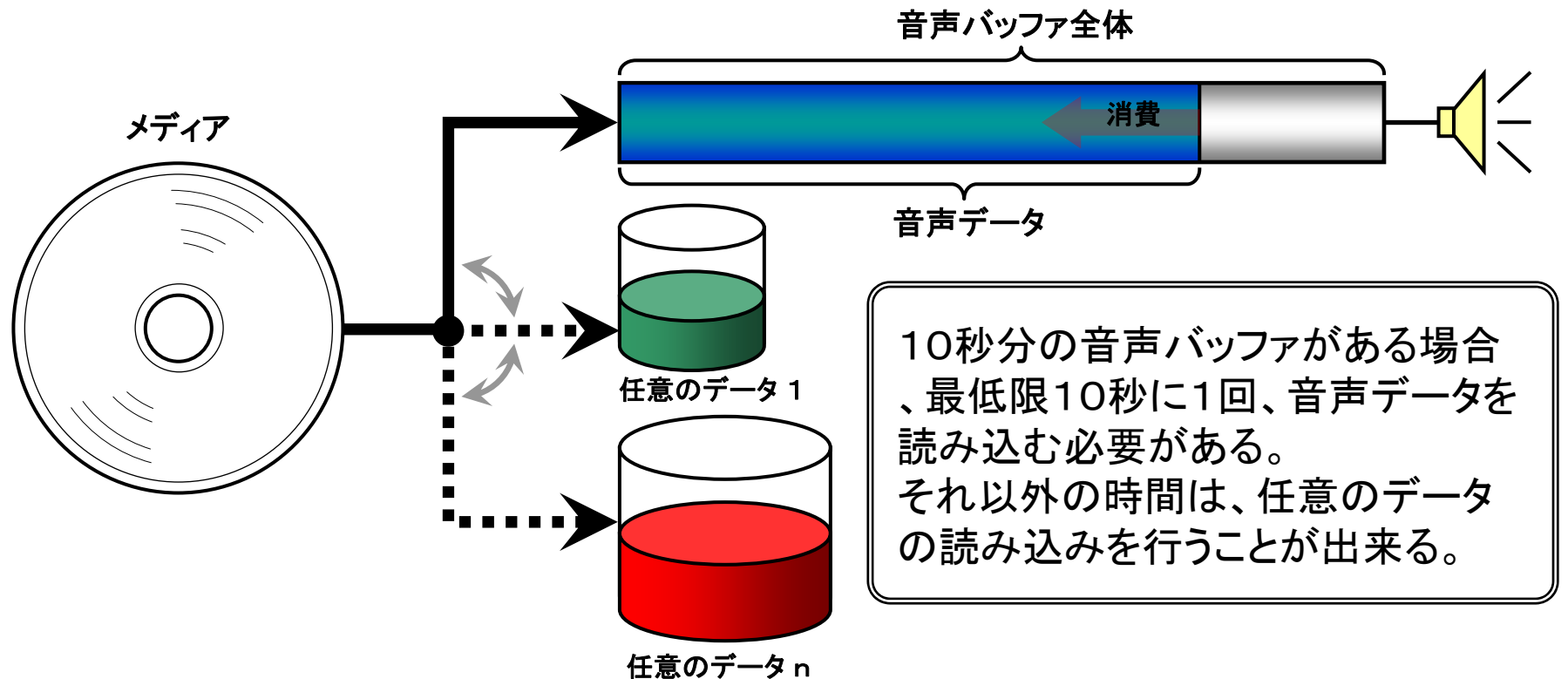
■ セリフの再生が終了すると、音楽用にバッファを結合



D-BAS™ によるデータ読み込みの高速化

■ ストリームバッファをできる限り有効利用しシークを削減

- 音声データ読み込みのためのシークを可能な限り抑え、読み込み時間を短縮。



『ファイルマジック』+ D-BASによる究極の奥義

『ファイルマジック』

データ圧縮

「読み込み速度が2倍に！」

D-BAS™

高性能ストリーミング

「シーク回数が減少！」

音楽のストリーム再生中でも
高速なゲームデータのロードが可能。

D-BASによるシーク音の軽減

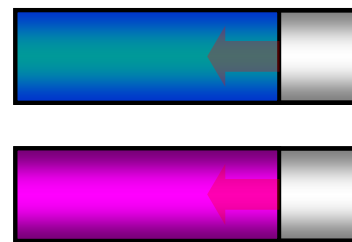
■ D-BAS™ のもうひとつの効用

- ストリーミング用バッファを状況に応じて動的に管理。
- 複数のストリームが再生されたときでも、できるかぎりバッファ大きくする。
→ シーク音が少なくなり、ドライブの寿命を延ばす。

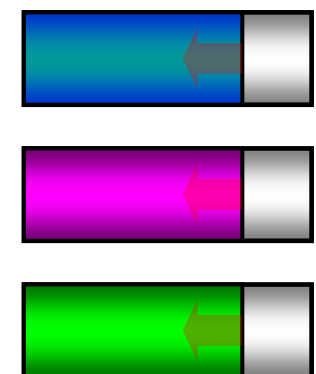
1ストリーム再生



2ストリーム再生



3ストリーム再生



同時に再生されているストリーム数に応じて、自動的にストリームバッファを分割する。



Wii: ムービ^ワ関連の秘技^ザ

Wii : ムービー関連のワザ

■ ムービーのファイルサイズを節約するワザ

- ファイルサイズとビットレート。
- 高圧縮しつつ少しでも画質をよくする。

■ ワイドテレビ用ムービーエンコードのワザ

- Wii向けのワイドテレビ用ムービーを作るには。
- ワイドテレビとノーマルテレビでムービーを流用する。

■ 多言語対応ムービー作成のワザ

- ムービーファイルに複数の音声と字幕を入れる。
- 字幕の表示タイミングをどうやって編集するか。

■ 演出ムービーのワザ

- 演出ムービーに必要な再生機能。

シネマスタジオ と CRI Sofdec

■ CRI Movie 『シネマスタジオ』

- メモリ使用量／CPU負荷を改善した新規ミドルウェア。
- ユーザ独自のファイルシステム／サウンドシステムへの容易な統合。
- CRI Audio との連携。

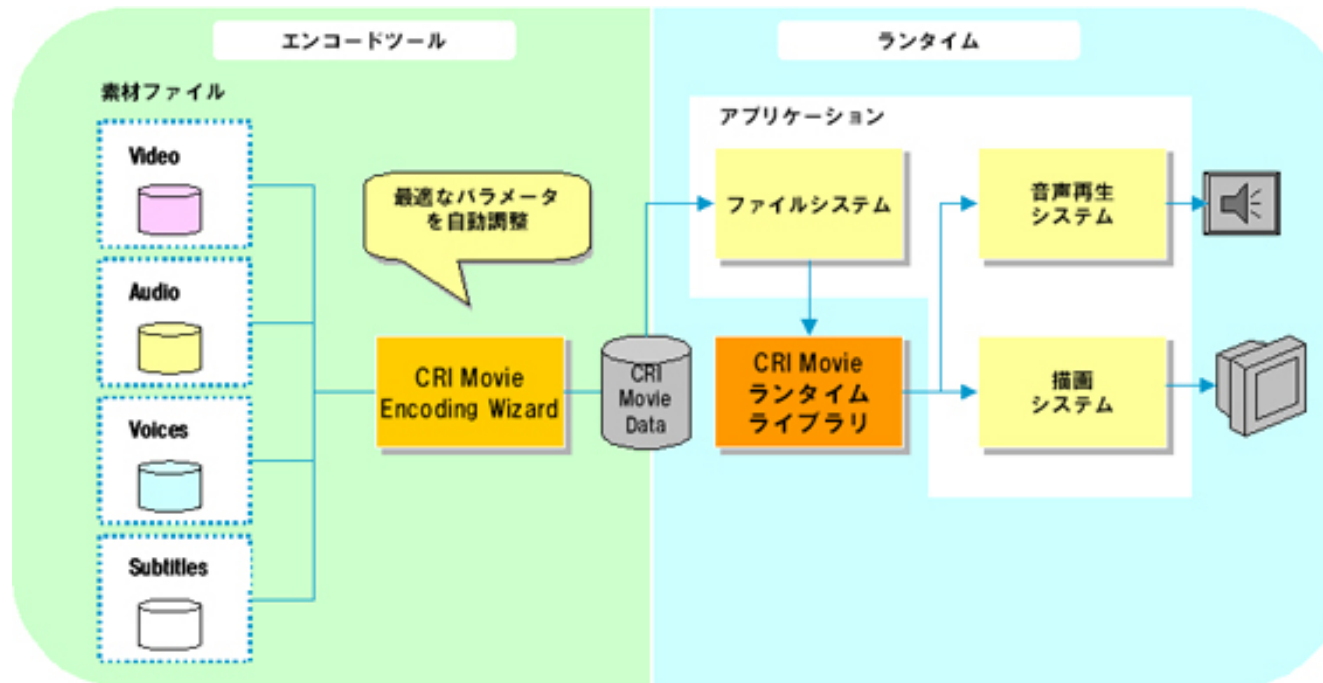
■ CRI Sofdec

- マルチプラットフォーム対応。
- CRI ADX との連携で全自動なファイル読み込み／サウンド処理。
- アルファムービーや連結。

両者は互換性の無い別ライブラリですが、ビデオの圧縮コーデックは共通のエンジンを使い、高画質のエンコード／再生を実現しています。

シネマスタジオのワークフロー

- ウィザード形式エンコーダでAVIから簡単ムービー作成
- 『CRI Movie』ランタイムライブラリでムービーをデコード



15: ムービのファイルサイズを節約する秘技

圧縮する前のムービファイルサイズ

■ 非圧縮素材のデータ量

圧縮前の動画素材(RGB888)のデータサイズは...

640x480, 30fps で1分の動画の場合、約1582Mbyte。

ビットレートで言うと約221Mbpsです。

■ 画質の大半はエンコードで決まる！

ムービの再生画質は、その大半がエンコード段階で決まります。

高周波成分を捨てれば圧縮はしやすくなりますが、ぼけた映像になってしまい、例えばCGのシャープ感は出なくなってしまう。

目標ビットレート計算式 (1)

■ 目標のファイルサイズに抑えるにはビットレートいくつ？

ムービの総尺T[秒]とエンコード後の目標ファイルサイズS[バイト]から計算。

$$\text{ビットレート[bps]} = \text{S[byte]} \times 8 \div \text{T[sec]}$$

(※可変ビットレートエンコードの場合は、正確にこの数値とはならない。)

このビットレートでエンコードしてみて、画質を評価する。

目標ビットレート計算式 (2)

(例)

1分で60Mbyteが目標なら、

$$\text{ビットレート} = 60 \times 1024 \times 1024 \div 60 \times 8 = 8,388,608$$

→約8.4Mbps

もちろん単純にサイズの問題ではなく、綺麗に再生できてかつファイルサイズを抑えるというバランスを考えることが重要。

ビットレートの目安



■ シネマスタジオ: ビットレートと画質の目安

- 高精細 10Mbps 激しい動きがあってもほとんどノイズが見られない。
- 高画質 8Mbps シャープな質感なども損なわずに再現できる。
- 普通 6Mbps 通常の映像であれば、ほぼ問題ない。
- 高圧縮 4Mbps 部分的にノイズが発生するが許容できる。

ファイルサイズ節約のために: 解像度 (1)

■ 限られたファイルサイズで少しでも画質をよくしたい。

ファイル容量の制限が厳しいと、ビットレートが低すぎて画質が出せないことがある。

■ 解像度を適度に落としてみる。

- 同じビットレートなら解像度が小さいほうが有利。

640x480を3Mbpsでエンコードするより、480x480を3Mbpsでエンコードして、表示で横方向に拡大したほうが綺麗になることは十分ありえます。

見栄えは再生時の拡大の綺麗さにも依存するので、解像度を落とす場合は、拡大に都合がいいようにキリのいい倍率にすることをオススメします。

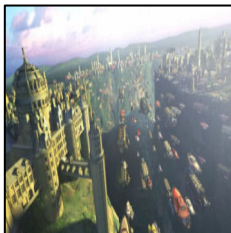
ファイルサイズ節約のために: 解像度 (2)

■ 解像度を適度に落としてみる。

- 解像度が小さいほうが圧縮には有利。最終的な見栄え勝負。



640x480を3Mbpsでエンコードして、1倍表示。



480x480にリサイズしてから、3Mbpsでエンコード。拡大表示。



シネマスタジオのエンコーダ



■ シネマスタジオのエンコーダ

- ウィザード形式のツールで、絵・音・字幕を一括エンコードします。
- 圧縮関連のパラメータは高画質／標準／高圧縮でテンプレート化。
- ファイルサイズまたはビットレートを直接指定することも可能。
- (今後) リサイズ機能等是对应検討中。
→現時点では事前に素材レベルでリサイズが必要。

■ CRIエンコードサービス

- エンコード専門スタッフが条件下でのベストクオリティを求めます。
テストエンコードやエンコードアドバイスもお気軽にお問い合わせください。

16: ワイドテレビ用ムービエンコードの秘技

ワイドテレビ

■ Wiiの映像出力は 480i/480p

ワイドテレビに対応(画面いっぱいに表示)する場合、
テレビによる横方向の拡大を使用すると、Wiiの画面解像度をフル活用できます。(以降、本資料ではテレビによる横方向のみの拡大を前提とします。)



テレビの拡大使用しない
(画面左右は非表示=黒)



テレビで横拡大

ワイドテレビ用ムービ

■ スクイーズ方式でのムービエンコード

横方向に拡大して16:9フル画面で正しく見えるということは、ムービ自体はスクイーズ方式(横に縮小)でエンコードする必要があります。



ムービは640×480スクイーズ

TVで横拡大



正しい縦横比

ワイドテレビ用ムービ作成のポイント

(1) ワイド表示したときに映像の縦横比は正しいか？

最終的にテレビにワイド表示したときに、正しい縦横比で見えるようにエンコード時から注意する必要があります。

(2) ワイドテレビ左右の黒帯はどうするか？

Wiiの画面を普通にワイドテレビに拡大表示すると、画面の左右に黒い帯(映像を表示しない領域)ができます。

この領域をどうするかはアプリケーションで決める必要があります。

上記(1)のためにも、黒帯をどうするかは重要な情報です。

そのままにするか、映像を横にさらにのばして黒帯を減らすか？

ワイド表示でムービの縦横比がおかしくなるケース

■ 縦横比がおかしくなるケース

1280x720のムービを単純に640x480にリサイズしてエンコードすると、テレビで横拡大したときに、少し縦長の映像になってしまいます。



ムービは640×480

TVは720x480
の領域を横拡大



縦長に見える



正しい縦横比

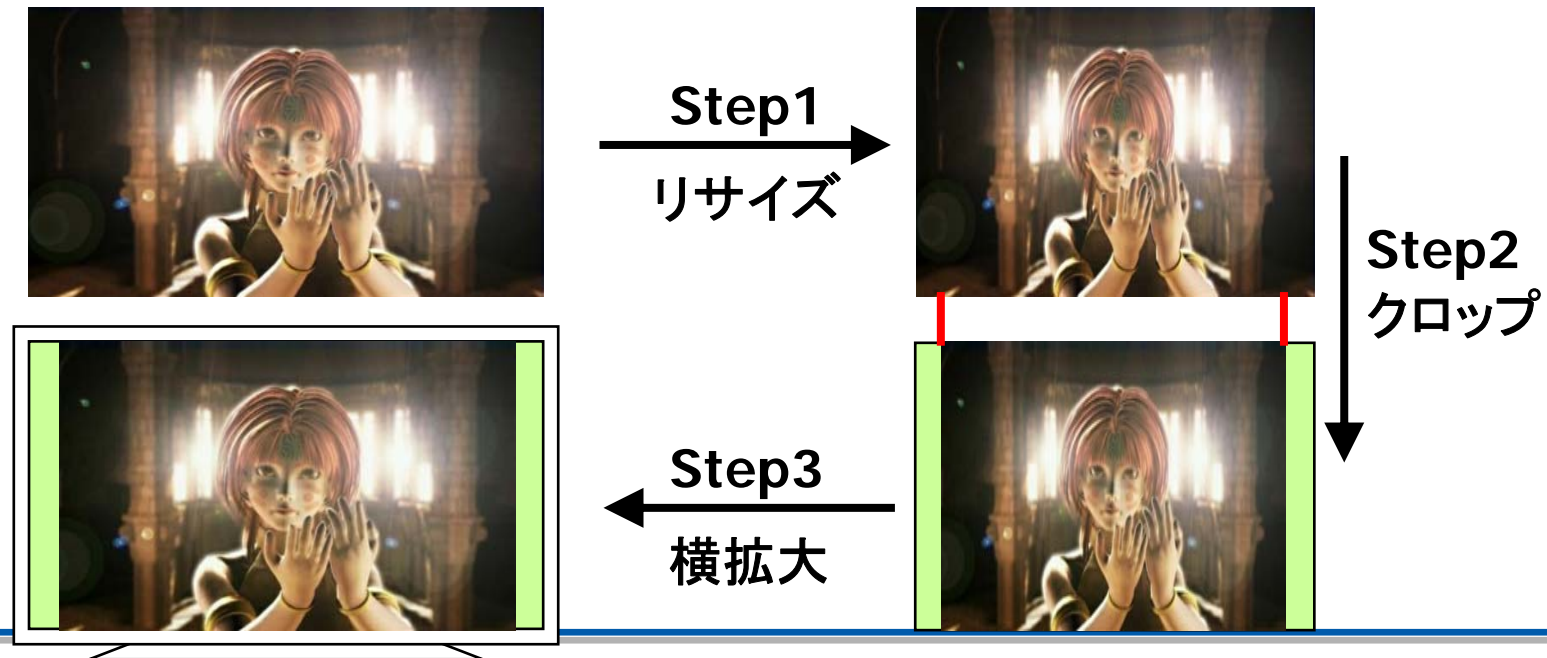
ワイド用ムービエンコード手順(基本)

■ ワイドテレビ用のリサイズとクロップ(基本)

Step 1. ワイド素材を720×480にリサイズする。

Step 2. ムービサイズ640×480に左右をクロップする。

Step 3. エンコード。再生時にテレビで横拡大する。



ワイドテレビ左右の黒帯 (1)

■ ワイドテレビ左右の黒帯

単純にテレビの拡大機能だけを使った場合、画面左右に黒い領域(何も表示されない)ができます。



この  の領域

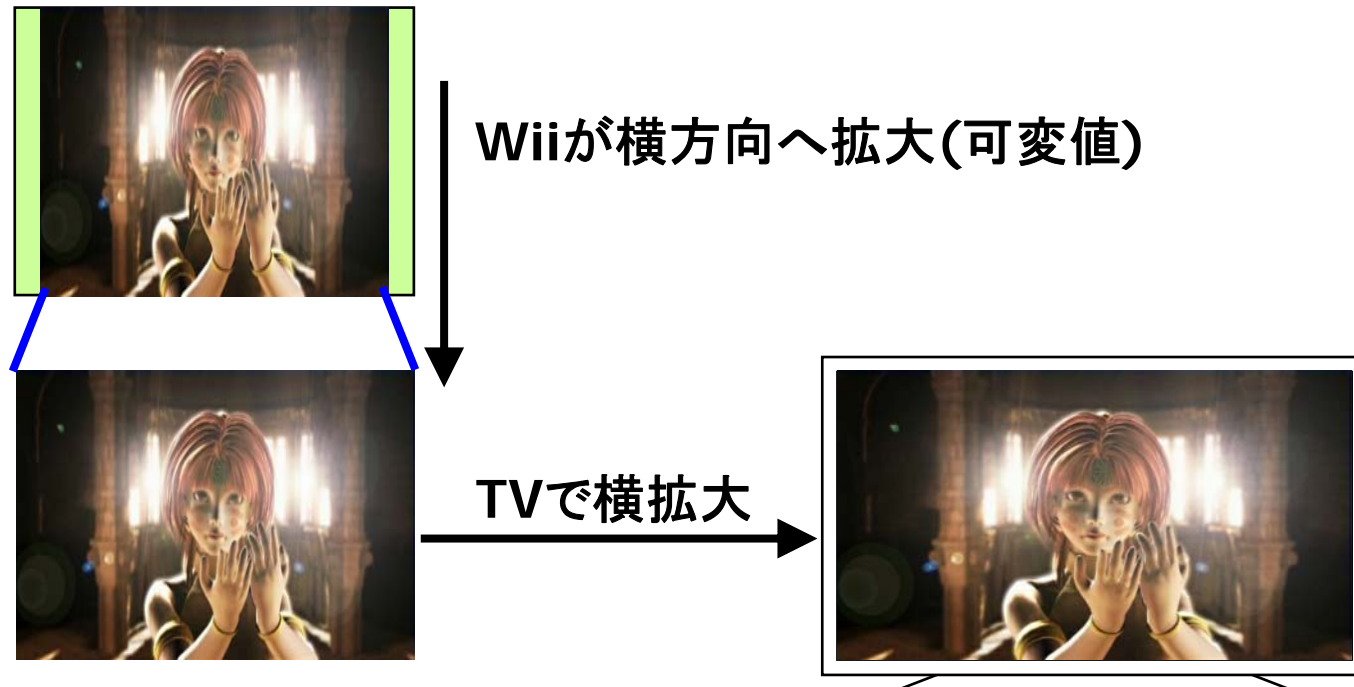
これを避けるために、テレビへの映像出力信号を横方向へ拡大します。

ワイドテレビ左右の黒帯 (2)

■ Wii+ワイドテレビによる横拡大

Wiiが拡大した映像をさらにテレビが拡大します。

これにより、テレビ左右の黒帯を減らすことができます。



Wiiによる横拡大

■ Wiiの横拡大について

ここでは説明のため、Wiiで横に拡大する値を「**スクリーン幅**」と名付けます。



スクリーン幅は**アプリ依存の値**です。

ムービをエンコードする際にアプリ開発者からこの情報をもらってください。

スクリーン幅は、ゲームの画面デザインやテレビのセーフティエリアを考慮して決められています。

ワイド用ムービエンコード手順

■ ワイドテレビ対応(改)

映像信号を横拡大する場合の手順です。

Step 1. ワイド素材を720×480にリサイズする。

Step 2. **スクリーン幅XXX**×480に左右をクロップする。

Step 3. ムービサイズ640×480にリサイズする。

Step 4. エンコード。再生時にWii+テレビで横拡大する。

(スクリーン幅を640とすると、基本手順と実質同じになります。)



Step2



Step3



Step4



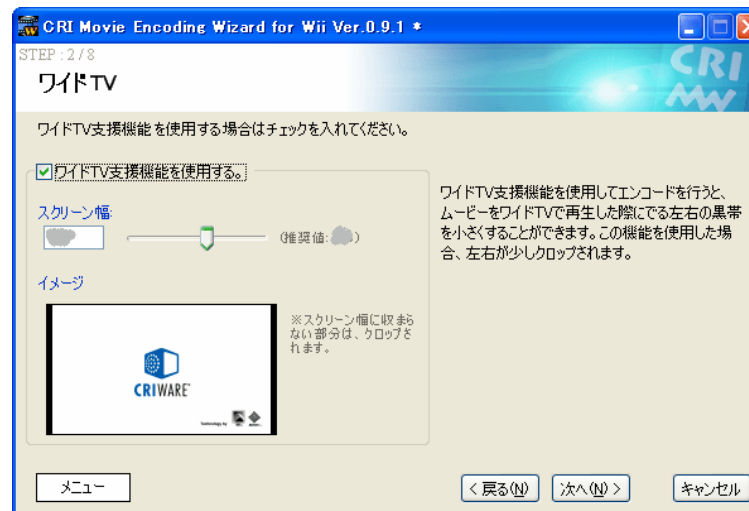
シネマスタジオによるワイド用エンコード



■ シネマスタジオの場合

ワイドテレビ支援機能があります。(素材が16:9用の場合のみ)
ここまで説明してきたクロップやリサイズを自動で一括処理できます。
「スクリーン幅」はエンコード時に指定します。

1280x720の素材も、簡単にワイド用エンコードができます。



ワイドテレビとノーマルテレビ

■ ワイドテレビとノーマルテレビ

ここまではワイド素材をワイドテレビ用にエンコードする説明でした。

実際は、

- ムービファイルはワイドテレビ用／ノーマルテレビ用？
- 使用しているテレビはワイドテレビ／ノーマルテレビ？

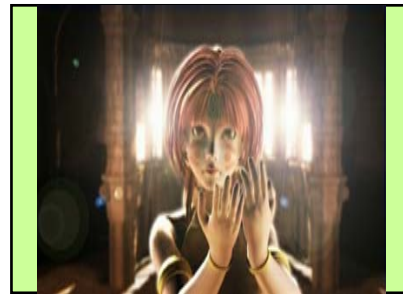
この2つの条件の組み合わせを考えなければいけません。

どういう表示方法にするかはアプリ側に選択の余地があります。

ワイド用ムービのノーマルテレビ表示

■ ワイド用ムービ → ノーマルテレビ

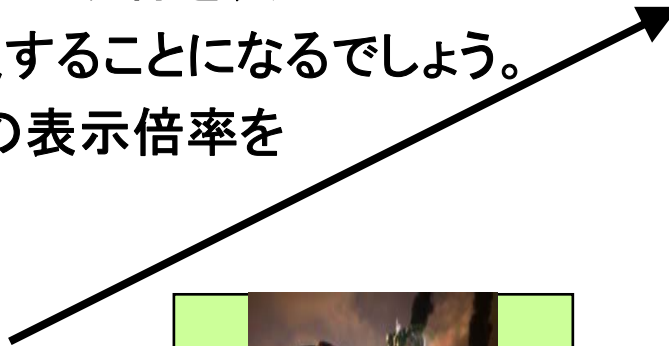
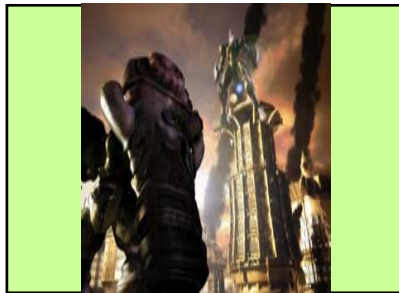
- 映像の左右を切って大きく表示
 - 上下に黒を入れて全映像を表示
- のどちらかを選択することになるでしょう。
それぞれで縦横の表示倍率を調整します。



4:3ムービのワイドテレビ表示

■ 4:3用ムービ → ワイドテレビ

- 映像の上下を切って大きく表示
 - 左右に黒を入れて全映像を表示
- のどちらかを選択することになるでしょう。
それぞれで縦横の表示倍率を調整します。



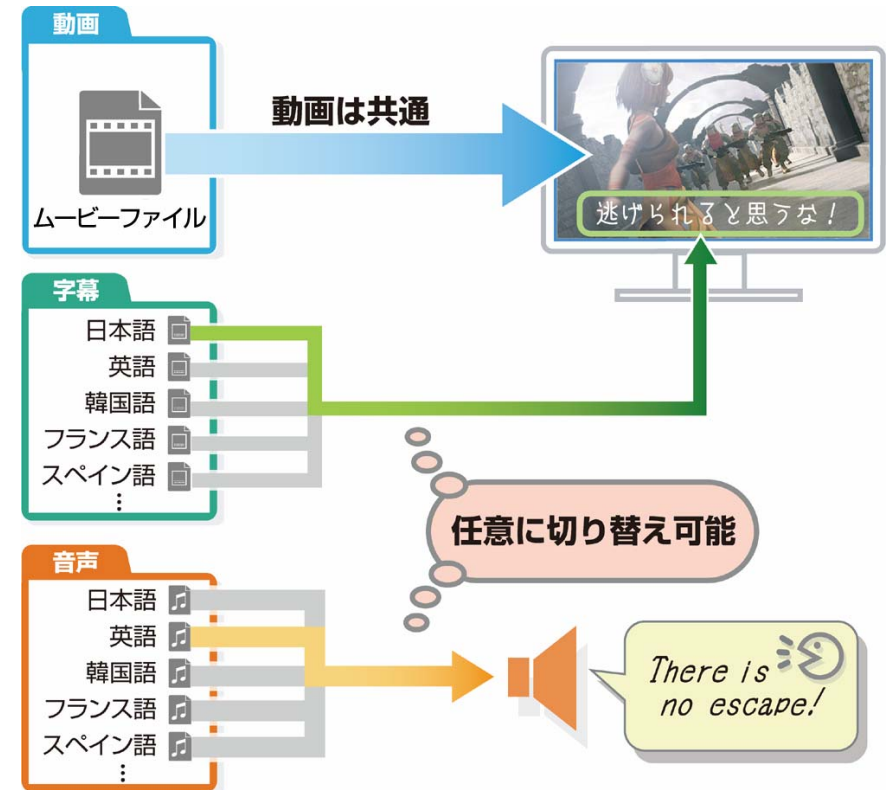
17: 多言語対応ムービー作成の秘技

多言語対応ムービ

■ 複数の音声と字幕を入れる。

ゲームを世界各地で発売する場合、ムービファイルのなかに必要な音声や字幕情報が入っていることはデータの取り回しを簡単にします。

もちろん音声を複数入れることはファイルサイズの増加につながるなので、空き容量との相談にもなります。



言語ごとにムービを作る場合

■ もしムービに複数の音声や字幕を入れられないと...

- 音声についての問題

絵と音を別ファイルにして同時再生するなら、音をオンメモリに乗せない限りマルチストリーム読み込みとなる。絵にたいする転送レートが落ちることと、AV同期など扱いにくい問題が発生する。

- 字幕についての問題

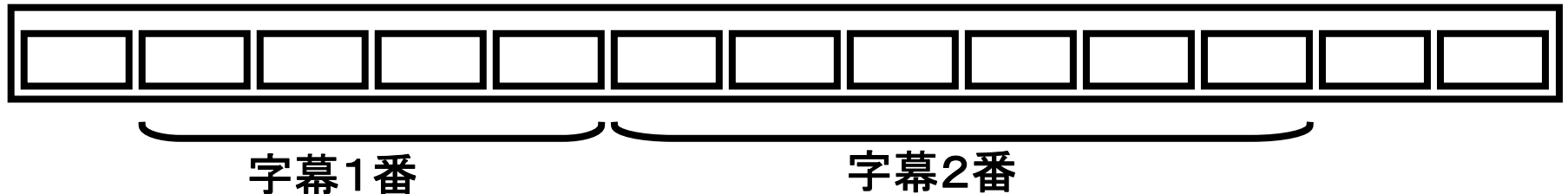
字幕情報を別ファイル化して、ムービの再生時刻を監視しながら字幕表示をすることになる。アプリの手間が増える。

字幕情報と識別子

■ 字幕情報にタイミングだけを入れる

字幕については表示タイミングと識別子だけをムービに埋め込む方法もあります。ムービからは「何番を表示するか」という情報だけを得て、文字列自体はアプリが別管理しておいて、識別子から変換するということです。

文字列をアプリ管理する手間はありませんが、ムービファイルに手を入れずに字幕内容を簡単に書き換えられるメリットがあります。

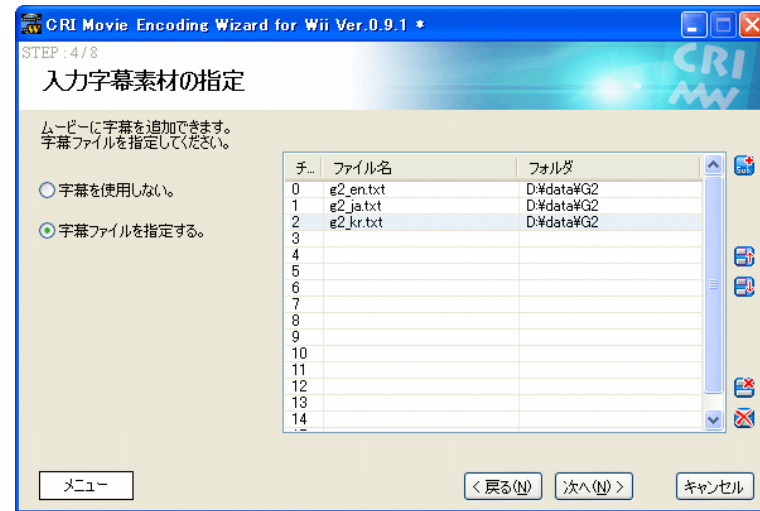
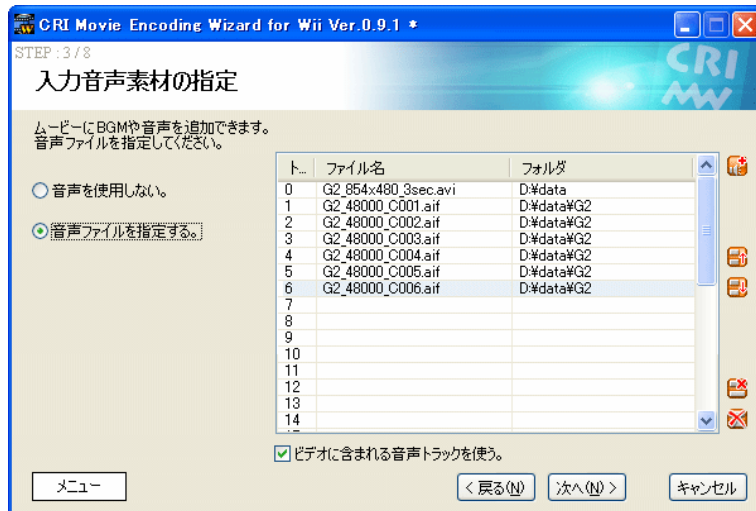


シネマスタジオによる多言語対応ムービ作成



■ シネマスタジオの場合

複数のオーディオ／字幕をムービに簡単に入れることが可能。
一つのエンコードツールで一括処理が可能。



字幕表示タイミングの編集方法

■ そもそも字幕の表示タイミングをどうやって編集するか？

字幕付きムービを作るということは、

- どういう文字列を
- いつからいつまで

表示するかをあらかじめ決める必要があります。

どんなツールで字幕編集を行うかは、各プロジェクトごとに環境が違うことと思いますが、ここではフリーソフト SubtitleWorkshop を使った例を簡単に紹介します。

字幕編集例: SubtitleWorkshop (1)

■ SubtitleWorkshop

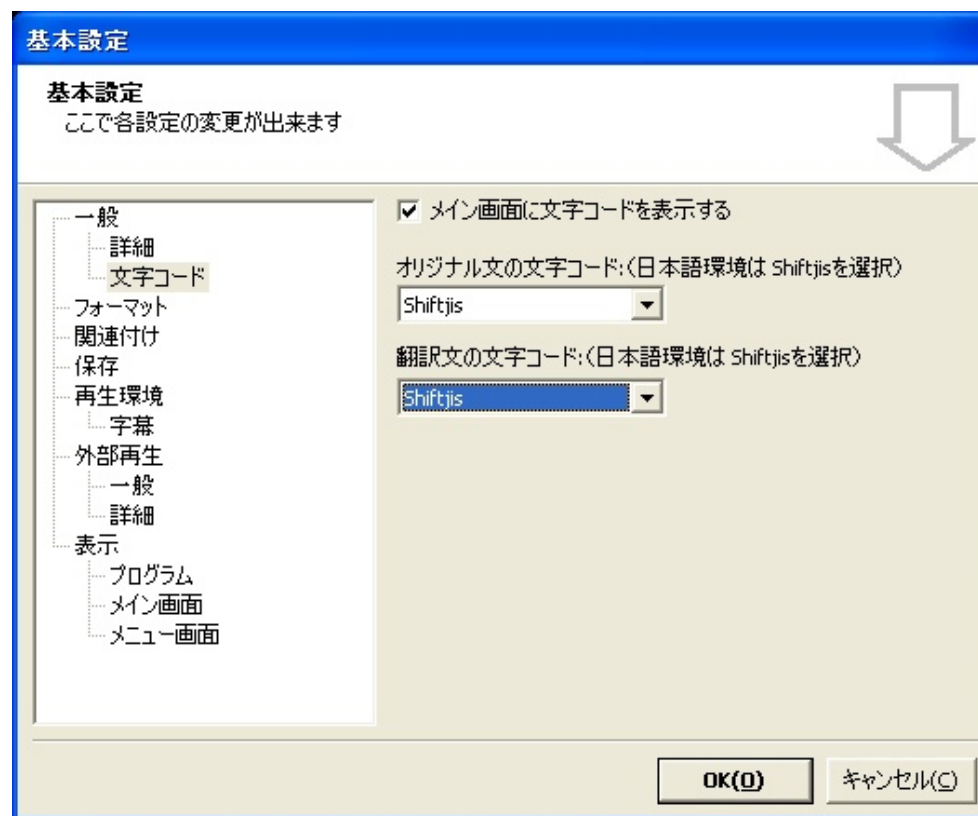
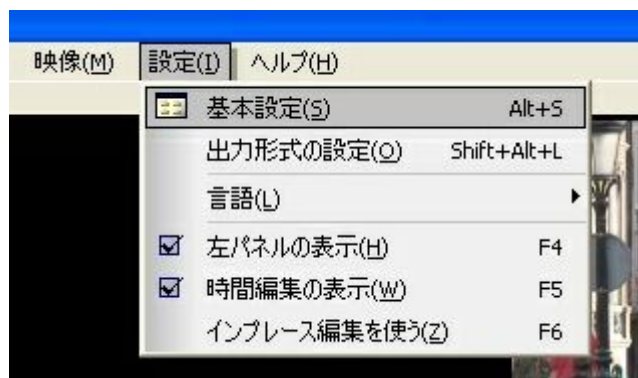
字幕情報を作るためのフリーソフトです。

AVIファイルを指定し、絵を見ながらタイミングの調整を行うことができます。

字幕の文字コードは Shift JIS で説明を行います。

字幕編集例: SubtitleWorkshop (2)

■ SubtitleWorkshop 初期設定: 文字コードの指定



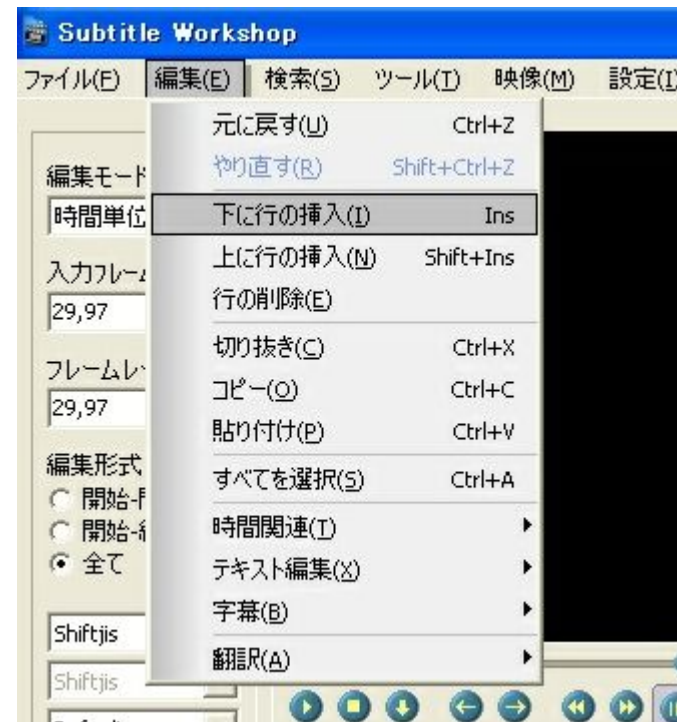
字幕編集例 : SubtitleWorkshop (3)

■ SubtitleWorkshop : 映像素材の指定



字幕編集例 : SubtitleWorkshop (4)

■ SubtitleWorkshop : 字幕の新規作成 / 追加



字幕編集例 : SubtitleWorkshop (5)

■ SubtitleWorkshop : 字幕情報の編集 / プレビュー

開始時刻: 終了時刻: テキスト文 (46 文字):

00:00:01,998 00:00:05,624

表示時間: 00:00:03,626

This is Utrecht, a city famous for its canals.



CRI字幕情報フォーマット (1)



■ CRI の字幕情報フォーマットで保存する

SubtitleWorkshopの字幕を保存するときに、カスタムフォーマットを使うことでシネマスタジオのエンコーダにそのまま入力できる CRI フォーマットで直接保存することができます。

- 時間単位の編集のみ
- ShiftJIS か 英語半角(UTF8)のみ

CRI字幕情報フォーマット (2)



■ CRI カスタムフォーマットのスクリプト

SubtitleWorkshopでCRI用に使うカスタムフォーマットの内容は次のようになります。時間単位の編集にのみ対応します。

```
[Information]
Name=CRI Subtitle (msec)
Extension=*.txt
Time structure=MS
Time=True
New line char=

[Format text]
1000
{RepeatSub}
{swStart}, {swEnd}, {swText}
{EndRepeat}
** This is a blank line that CRI Subtitle format needs: !*
```

もちろん、任意のフォーマットに保存してからCRI フォーマットに自前で変換するのもあります。

字幕の文字コード

■ 文字コードについて

たいていの場合、字幕は「絵」としてではなく「文字列」として管理することになるでしょう。

日本語と英語だけなら Shift JIS という選択肢があります。

ヨーロッパ各国や韓国語なども扱う場合はUnicode (UTF8, UTF16) がありますし、言語別に専用の文字コードもあるでしょう。

アプリの使用しているフォントが使用している文字コードを使うのが、最も簡単な選択になると思います。

18: 演出ムービの秘技

演出ムービ

■ 演出や飾りとしてムービを使う

オープニングやデモシーンのようなフル画面シーンだけでなく、メニュー画面などで演出や飾りにムービを使う方法もあります。

演出ムービのメリットは、演出内容の表現をプログラマではなく、デザイナーが自由に調整できることにあります。

演出ムービに重要な機能

■ 演出ムービに必要な機能

メモリ使用量やデコード負荷という条件以外に、ムービ再生エンジンに次のような機能があるかどうか演出ムービのポイントになるでしょう。

- ループ再生
- 連結再生
- アルファムービ

デモシーンと違い、演出ムービはユーザ操作があるまで延々と再生を続ける可能性があります。ループはもちろん、連結やアルファが可能かどうかは、演出の幅につながります。

このほか、オンメモリ再生が欲しい場合などもあるでしょう。

シネマスタジオによる演出ムービ



■ シネマスタジオの機能

シネマスタジオの場合、演出ムービ向けの機能は...

● シームレスループ再生可能

オンメモリ、ファイル読み込みともループ再生可能です。

● シームレス連結再生可能

連結再生のファイル登録は、コールバック関数方式です。

オンメモリムービの連結再生は未対応。(今後予定)

● アルファムービ未対応 (今後対応予定)

デモプログラムのメニュー



■ デモプログラムのメニュー画面

シネマスタジオデモのメニュー画面背景は、ムービのループ再生を使っています。

例えばこのデザインを変える場合、プログラムは再生するファイル名を変えるだけでいいというわけです。



お問い合わせ先

URL

<http://www.cri-mw.co.jp/inquiry>

メール

ファイルマジック	: FileMajik @cri-mw.co.jp
CRI Audio	: CriAudio @cri-mw.co.jp
シネマスタジオ	: CinemaStudio @cri-mw.co.jp

CEDEC当社講演内容に関するご質問やお問い合わせは、
お気軽に下記メールアドレスまでご連絡下さい。

cedec2007@cri-mw.co.jp