

THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS

電子情報 通信学会誌

THE JOURNAL OF THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

本会のWWWホームページ
<http://www.ieice.org/>

特別小特集 Cool Japan を支える ICT 技術

- ・電磁波工学教育におけるシミュレータ利用の可能性
- ・オープンソースソフトウェア無線プラットフォーム
GNU Radio
- ・画像・映像意味解析技術の過去、現在と今後の可能性

EIC

一般社団法人

電子情報通信学会

平成25年1月
JANUARY
Vol.96 No.1

1

信学誌 / Vol.96 No.1 pp.1-68 2013/1/1 東京

4

ゲーム開発を支えるミドルウェア

Middleware for Game Development

新井タヒル

1. はじめに

近年、コンシューマゲームの開発規模は急速に拡大し、開発スタッフの分業化も進んでいる。各社が競い合い、新しい手法を生み出してきたが、共通化できる部分も多く見られ、それらを専門とする会社が、ミドルウェアと呼ばれるライブラリ製品を販売するようになった。ミドルウェアの中でも、特にグラフィック等のゲーム開発の中心部分に位置するものをゲームエンジンと呼ぶ。海外では優秀なミドルウェアが多く生み出され、開発コストを抑えつつ良い製品を生み出す流れにシフトしているが、日本の開発現場ではミドルウェアの導入が遅れている。その背景には、日本人の職人のような作品へのこだわりから、機能が制限されているミドルウェアへの抵抗があったと考えられる。しかしそれが技術進歩の遅れや開発コストの高騰につながり、結果的に作品のクオリティが下がってしまうことも多く、ミドルウェアが見直されるようになってきた。本稿では日本製ゲームエンジン OROCHI を例に、ミドルウェアの基本技術を紹介する。

2. ミドルウェアの位置付けと役割

図1のように、各プラットフォームにはそれぞれのSDKによるAPIが存在しているが、それらのインタフェースを共通化し、機能を追加してゲーム開発者に提供するのが、ミドルウェアの主な役割である。(中間に

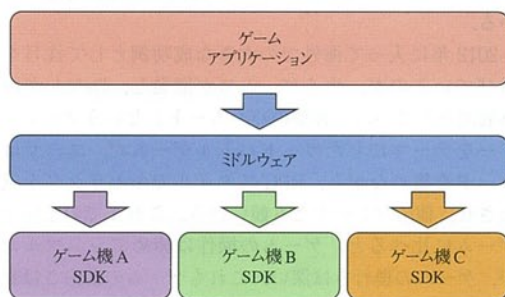


図1 インタフェースの共通化

位置するため、ミドルウェアと呼ばれる。) ミドルウェアの種類としては、システム、グラフィック、物理アニメーション、AI (人工知能)、ネットワーク、サウンド等がある(詳細は後述)。これらはランタイム上で動作するものだが、ゲーム開発ではランタイムに読み込ませるためのデータ(主にデザイナーが作成する)が必要であり、ミドルウェアにはそれらのデータを製作するためのツールも含まれる(図2)。

3. ミドルウェア選定のポイント

ミドルウェアを使う上で重要視されるのが、性能(処理速度、消費メモリ)、コスト(開発効率)、柔軟性(拡張性)である。

3.1 性能

携帯電話やPCと違い、ゲーム専用機ではCPUやメモリ等の性能が固定されており、それらの性能を最大限に引き出せるかが重要になる。複数のプラットフォームに

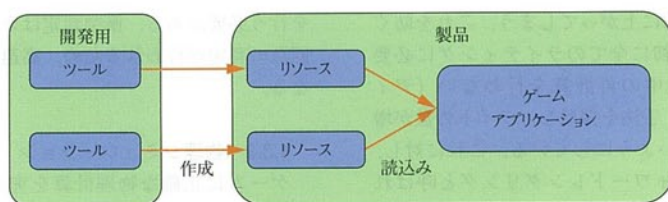


図2 データ作成フロー

向けて開発する場合は最も性能の低い機種に合わせて開発すれば問題はないが、最近ではそれぞれの機種に特化した機能もあり、一概に性能を比較できなくなっている。そういった条件の中では、それぞれの機種向けに最適化するという作業が必要となり、ミドルウェアがそこを担うことで、ゲーム開発者の負担を削減できる。複数プラットフォームで製作すると性能が落ちてしまいがちだが、ミドルウェアは最適化されているため、性能の低下を軽減してくれることになる。

3.2 コスト

費用対効果を考える上で、ミドルウェアの製品自体の機能、価値だけではなく、開発チームとの相性も考慮する必要がある。ゲーム開発のコストのほとんどは人件費であり、デザイナー等のデータ制作者が約7割を占める。データ制作者の作業効率はツール・ミドルウェアへの依存度が高く、使用するミドルウェアがプロジェクトコストに大きく影響することになる。

3.3 柔軟性

開発効率と柔軟性は相反するものである。設定項目を減らし簡略化することでツールの使いやすさは改善されるが、開発の自由度は失われる。ゲーム開発現場のニーズを理解し、開発効率と柔軟性のバランスを意識して対応することが、ミドルウェアに求められている。

4. 技術詳細

4.1 システム

4.1.1 メモリ管理

ゲーム機は要求レベルと比較するとメモリ量は小さく、常に開発者を悩ませる。更に複数のオブジェクトが複雑に動作するため、メモリの破壊や漏れを検出するためのシステムが必要となる。OROCHIではメモリ管理システム、それらを確認するためのツールや、漏れ検出機能を提供している。OROCHIには含まれていないが、定期的にメモリの断片化を整理するガベージコレクションを実装しているエンジンもある。

4.1.2 ファイル読み込み

昨今ではゲーム媒体が光ディスクになり、データ読み込み時のディスクアクセス時間がストレスと感じられるようになった。そのストレスを少しでも軽減するために、ゲームのメイン処理とは別に非同期でファイルを先読みしたりする手法がある。今では当たり前のように実装されているが、読み込みタイミングによってバグが発生しやすく、安定したミドルウェアに依存したい部分である。また、ディスクに対して同時にアクセスはできないため、できるだけファイルを圧縮して読み込み時間を減らし、展開に時間がかかる場合は複数スレッドで対応するのが望ましい。

4.2 グラフィック

4.2.1 マテリアル

3Dモデルがどのような質感で表現されるかを決めるのが、マテリアルと呼ばれるデータである。マテリアルは複数のシェーダー（GPU用のプログラム）と、シェーダーに必要なパラメータを保持している。シェーダーには基本となるテクスチャの色情報以外に、法線、正反射等の情報が使われ、それらを組み合わせて個性的な質感を生み出すことが可能である。マテリアルの調整は従来はDCC（Digital Contents Create）ツール上で行われていたが、シェーダーの種類が増え、開発機上でしか確認できない表現もあることから、独自のツールでレビューしながら調整するケースが増えてきた。

OROCHIでは開発機上で調整が可能となっている。

4.2.2 光と影

ゲームのリアリティを急速に向上させたのが、ライティングとシェーディングの技術である。それらはブリレンダーのムービーであればレイトレーシングという技術で現実極めて近い表現が可能だが、リアルタイムのゲームではとても処理が間に合わないため、低負荷で高クオリティの表現が求められる。ライティングは、メッシュの頂点の法線や、法線マップによる面のピクセルの法線から計算される。光源計算はゲーム上に配置されているライトからそれぞれのオブジェクトに対して行う必要があるが、ライトやオブジェクトの数が増えてくる

6. ま と め

このように、ゲーム開発現場では表現力の広がりとともに、多くの分野からの技術が取り込まれている。それぞれに深い専門知識が求められ、専門分野に特化した技術者が開発するミドルウェアへの依存度は、今後も高まっていくことだろう。ただし、ミドルウェアはゲーム開発におけるツールにすぎないので、開発者にとっては使いやすさが重視されることを忘れてはならない。日本の開発現場に適したミドルウェアを充実させることが、次世代のゲーム開発における重要な課題となるだろう。

文 献

- (1) ジェイソン・グレゴリー、ゲームエンジン・アーキテクチャ、ソフトバンククリエイティブ、2010。
- (2) 安田絹子、小林林広、飯塚博道、阿部貴之、青柳信吾、マルチコア CPU のための並列プログラミング、フィックスターズ(編)、秀和システム、2006。
- (3) 西川善司、3D ゲームファンのためのグラフィックス講座、インプレスジャパン、2010。
- (4) Game Programming Gems, M. Deloura (監修)、ボーンデジタル、2001。

(平成 24 年 7 月 17 日受付 平成 24 年 8 月 9 日最終受付)



あらに
新井 タヒル

平 17 阪大・基礎工・情報卒、同年、(株)ゲームリパブリック入社、平 23 シリコンスタジオ株式会社入社、ゲームエンジン OROCHI の開発チームリーダー。

