

MKng プロジェクトにおける携帯端末サポート

5B-5

黒岩 実 石井 秀浩 高野 陽介 横田 実
NEC C&C 研究所

1 はじめに

最近、マルチメディアやインターネットを駆使する高度なアプリケーションを扱う携帯端末の実現が期待されている。我々は、MKng プロジェクト [1] の一環として、そのような携帯端末の実現を目指した小型で高機能な OS プラットフォームの研究開発を行っている。本稿では、RT-Mach のマイクロカーネルに関して、携帯端末に適用する上での課題を整理し、その解決方針について提案する。

2 携帯端末の特徴

携帯端末は、デスクトップ機と比較すると以下のような特徴を持つ。

- 実メモリベース
可搬性の面で重量やサイズに制約があることや、アプリケーションやデータをネットワークに接続されたサーバで管理できること、などの理由から、ハードディスクなどの二次記憶に依存しない、実メモリベースのものが主流になると考えられる。
- 高機能なアプリケーション
マルチメディア処理やインターネットアクセスなど、機能的にデスクトップ機と同程度のサービスを要求される。
- 限定されたアプリケーション構成
携帯端末では、限られた資源を有効利用するために、アプリケーション構成が利用環境に応じて限定化されるのが普通である。

このため、携帯端末では、二次記憶を仮定しない実メモリベースでの動作が OS に要求され、また、アプリケーション構成を限定することによって OS の提供する機能も限定されたものに固定化することができる。

3 RT-Mach を携帯端末に適用する上での課題

本節では、RT-Mach をベースとして、携帯端末の制御に適した OS プラットフォームを構成する上での課題について述べる。

OS design for mobile computer in MKng research project
Minoru KUROIWA, Hidehiro ISHII, Yosuke TAKANO and
Minoru YOKOTA

C&C Research Laboratories, NEC Corp.

4-1, Miyazaki, Miyamae-ku, Kawasaki, Kanagawa 216, Japan

3.1 RT-Mach のメモリ消費

図 1 は、i486 ベース¹の AT 互換機²上での、RT-Mach ロード直後と Lites 起動直後における RT-Mach のメモリ消費量を示している。RT-Mach は、豊富な機能を持つプロセス間通信機構や仮想記憶機構を実現するために、メモリの消費量が多い。特に、カーネル内に動的に確保されるメモリ消費が大きな割合を占めているが、これは、多量のポートやスレッドを作成するアプリケーション構成法に因るものと考えられる。実メモリベースのメモリ管理を必要としている携帯端末では、RT-Mach が消費するメモリを低減することが必要である。

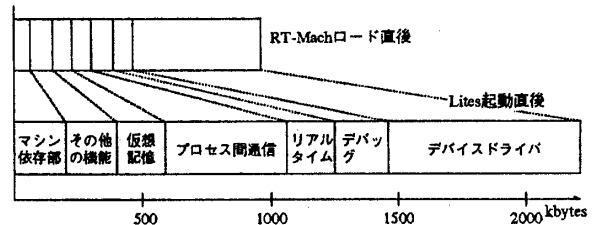


図 1: RT-Mach の消費メモリ

3.2 メモリ枯渇時の対応

RT-Mach では、通常の仮想記憶機構に加えて、アプリケーションのアドレス空間に実メモリを固定するメモリ割り当て機能を提供している [2]。しかし、メモリが枯渇した時にはマイクロカーネルやアプリケーションが回復不可能な状態に陥る可能性がある。そのため、このような事態を回避するための機能を RT-Mach に付加することが必要である。

3.3 イベント処理時のオーバーヘッド

手書き入力やディスコネクト可能なネットワーク機能を提供する携帯端末では、デバイスドライバの割り込み処理ルーチンから、ユーザレベルのイベントハンドラを実行するまでの遅延を短くすることが必要である。RT-Mach は、デバイスドライバからユーザレベルのハンドラにイベントを通知するメカニズムを提供しているが、カーネルからユーザタスクに切り替えるオーバーヘッドが大きい。このオーバーヘッドを低減するためには、カーネル内サーバ [3] やスレッド・マイグレーション

¹i486 はインテル社の商標です

²IBM, Personal Computer AT は IBM 社の商標です

ン[4]に見られるような、より効率的な方式を検討する必要がある。

3.4 サーバレイヤのモジュール化

携帯端末では、サーバレイヤをモジュール化し、それを組み合わせることによってアプリケーション環境に最適なサーバ構成を実現できることが必要である。デスクトップ機ではUNIX³の全ての機能をエミュレートするUNIXサーバを実行して、互換性を維持する場合が多いが、アプリケーション構成を限定できる携帯端末では、UNIXとの互換性は必ずしも必要ではない。例えば、ハードディスクを持たずUNIXの高機能なファイルシステムが必要ない場合は、ROMやフラッシュメモリに特化した小型のファイルシステムを提供できることが必要である。

4 携帯端末サポートの方針

4.1 RT-Machの小型化

RT-Machが消費するメモリを低減するためには、RT-Machが備える機能を更にモジュール化して、アプリケーションに必要な機能だけを組み合わせた小型のマイクロカーネルを構成できることが必要である。それによって、静的メモリ領域であるオブジェクトサイズを低減させられるとともに、スレッドやポートなどに関連した構造体の小型化によって動的メモリ領域も低減することができる。

プロセス間通信機構(IPC)を例にすると、RT-MachのIPCは様々な形態の通信モデルを実現するために、豊富な機能を提供している(図2)。携帯端末では、アプリケーション構成を限定でき、また、シングルユーザを仮定できるので、アプリケーション構成の動的な変化を検出するポート状態変化の通知や、ポートに対する不当なアクセスを防止するためのポート権などの機能は必要ない場合もある。

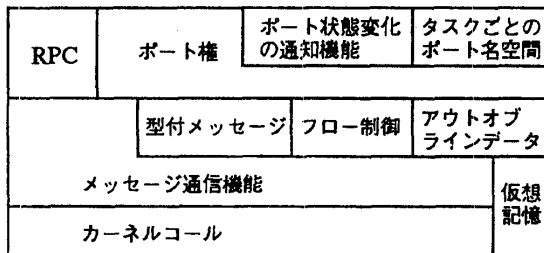


図2: IPCの機能ブロック図

また、現状のOSサーバではスレッドやポートを多用するために、それらに関連した構造体によって多くのメモリが消費される。携帯端末では、アプリケーション構成が限定できるため、スレッドやポートの使用を最小限にとどめたサーバを実現でき、マイクロカーネルのメモリ消費量を低減することも可能である。

³UNIXはX/Open Co. Ltd.がライセンスしている登録商標です。

4.2 実メモリベースのメモリ管理

実メモリベースでメモリ管理される携帯端末ではメモリをより有効に利用するため、以下のようなメカニズムを実現すべきであると考えている。

- **メモリ枯渇時の縮退動作**
メモリ枯渇時にシステムが回復不能状態に陥ることを回避するために、適切なアプリケーションを強制終了して縮退動作することによってシステムを維持するメカニズムが必要である。例えば、文書編集中にメモリが枯渇した場合には、時計などのデスクトップアクセサリなどを強制終了することによって文書編集が継続される方が望ましい。
- **マイクロカーネルとアプリケーションの協調制御**
RT-Machでは、マイクロカーネルが管理するメモリを有効に利用するメカニズムは組み込まれているが、アプリケーションレベルのメモリ管理は各アプリケーションに任されている。メモリ枯渇をできる限り防ぐには、マイクロカーネルとアプリケーションがメモリを協調制御するメカニズムが必要がある。例えば、ディスプレイサーバが管理するフォントキャッシュでは、メモリ不足時にキャッシュサイズを縮小することが望ましい。現状では、そのメモリをマイクロカーネルから制御することはできないが、ディスプレイサーバとマイクロカーネルが協調することによって、より効率的なメモリ管理が可能となる。

5 おわりに

本稿では、RT-Machを携帯端末に適用する上での課題と解決方針を述べた。現在、IPCのモジュール化と実メモリベースでのメモリ管理について、検討と設計を進めている。今後は、これらのメカニズムを、モバイルIP機能などのサービス面での充実を含めて、携帯端末上で実装・評価する予定である。

参考文献

- [1] 徳田, 追川, 西尾, 萩野, 斎藤: "MKng: 次世代マイクロカーネル研究プロジェクト," 第53回情処全国大会論文集, 5B-4 (1996).
- [2] H.Tokuda, T.Nakajima and P.Rao: "Real-Time Mach: Towards a Predictable Real-Time System," USENIX Mach Workshop, pp.73-82 (1990).
- [3] Jay Lepreau, Mike Hibler, Bryan Ford and Jeffrey Law: "In-Kernel Servers on Mach 3.0: Implementation and Performance," Proceedings of the Third USENIX Mach Symposium, pp.39-55 (1993).
- [4] Bryan Ford, Jay Lepreau: "Evolving Mach 3.0 to a Migrating Thread Model," Proceedings of the Winter USENIX Conference, pp.97-114 (1994).