ox_mathへの計算中断機能の実装

小原功任

ohara@OpenXM.org

金沢大学理学部計算科学科

ox_math とは

- OpenXM プロトコルをサポートするための Mathematica Kernel の wrapper
- MathLink (Mathematica の通信ライブラリ) と OpenXM プロトコルの両方を扱う
- 今回、ox_math に計算中断機能を実装した

ox_math とは

- OpenXM プロトコルをサポートするための Mathematica Kernel の wrapper
- MathLink (Mathematica の通信ライブラリ) と OpenXM プロトコルの両方を扱う
- 今回、ox_math に計算中断機能を実装した

  (MathLink の非公開機能を用いる)

ox_math とは

- OpenXM プロトコルをサポートするための Mathematica Kernel の wrapper
- MathLink (Mathematica の通信ライブラリ) と OpenXM プロトコルの両方を扱う
- 今回、ox_math に計算中断機能を実装した

  (MathLink の非公開機能を用いる)
- テスト環境

  Mathematica 4.x (Linux, Windows),
  Mathematica 3.x (Solaris)

OpenXM と MathLink の比較

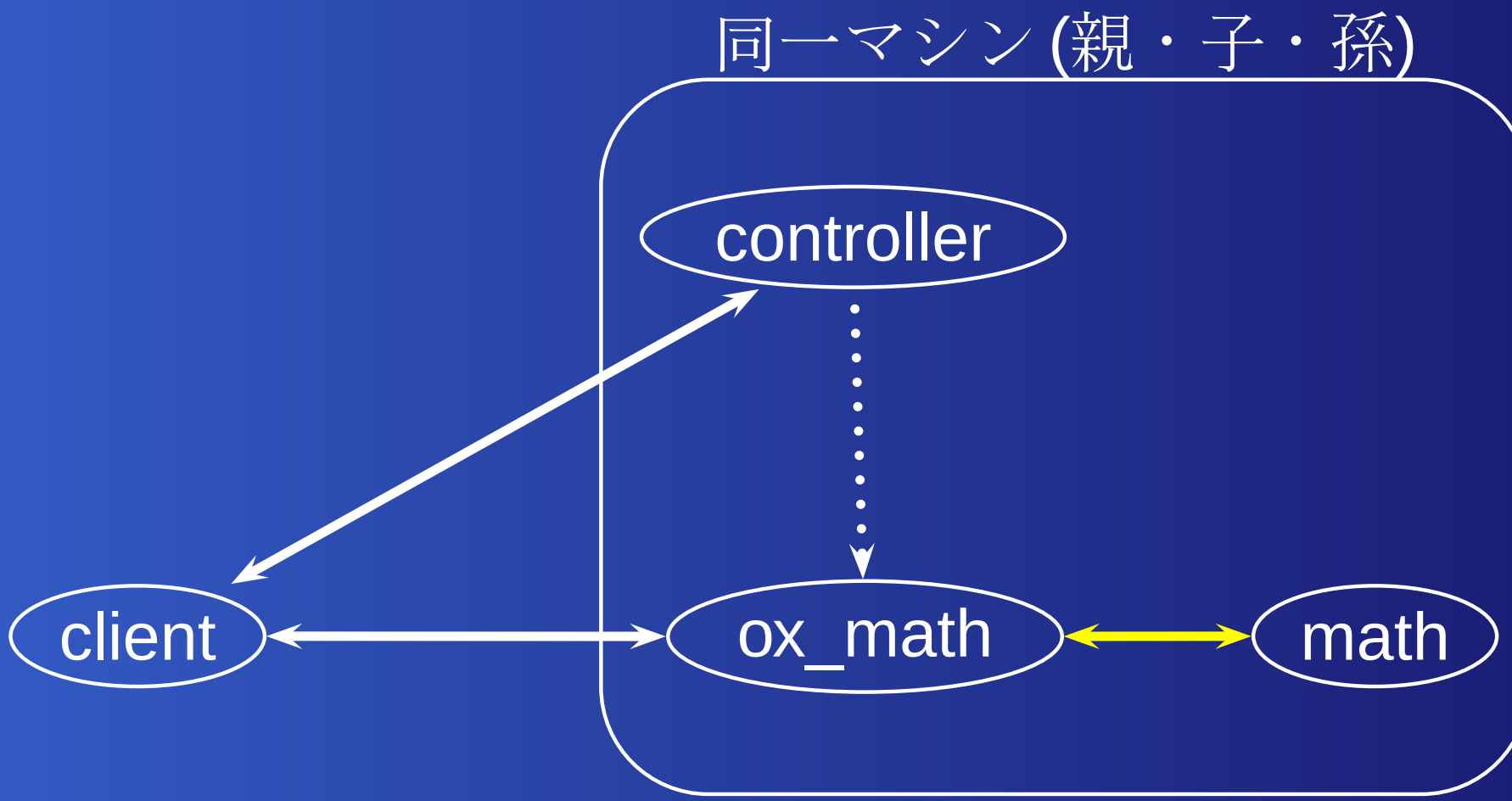
- OpenXM
  プロトコルを公開。仕組みが明らかだが自分で実装しなければならない。
- MathLink
  ライブラリを公開。実装しなくてよいが中身が分からない。またソースが公開されていない。

OpenXM と MathLink の比較

- OpenXM
  プロトコルを公開。仕組みが明らかだが自分で実装しなければならない。
- MathLink
  ライブラリを公開。実装しなくてよいが中身が分からない。またソースが公開されていない。

  文書化されていないことをするときに困る

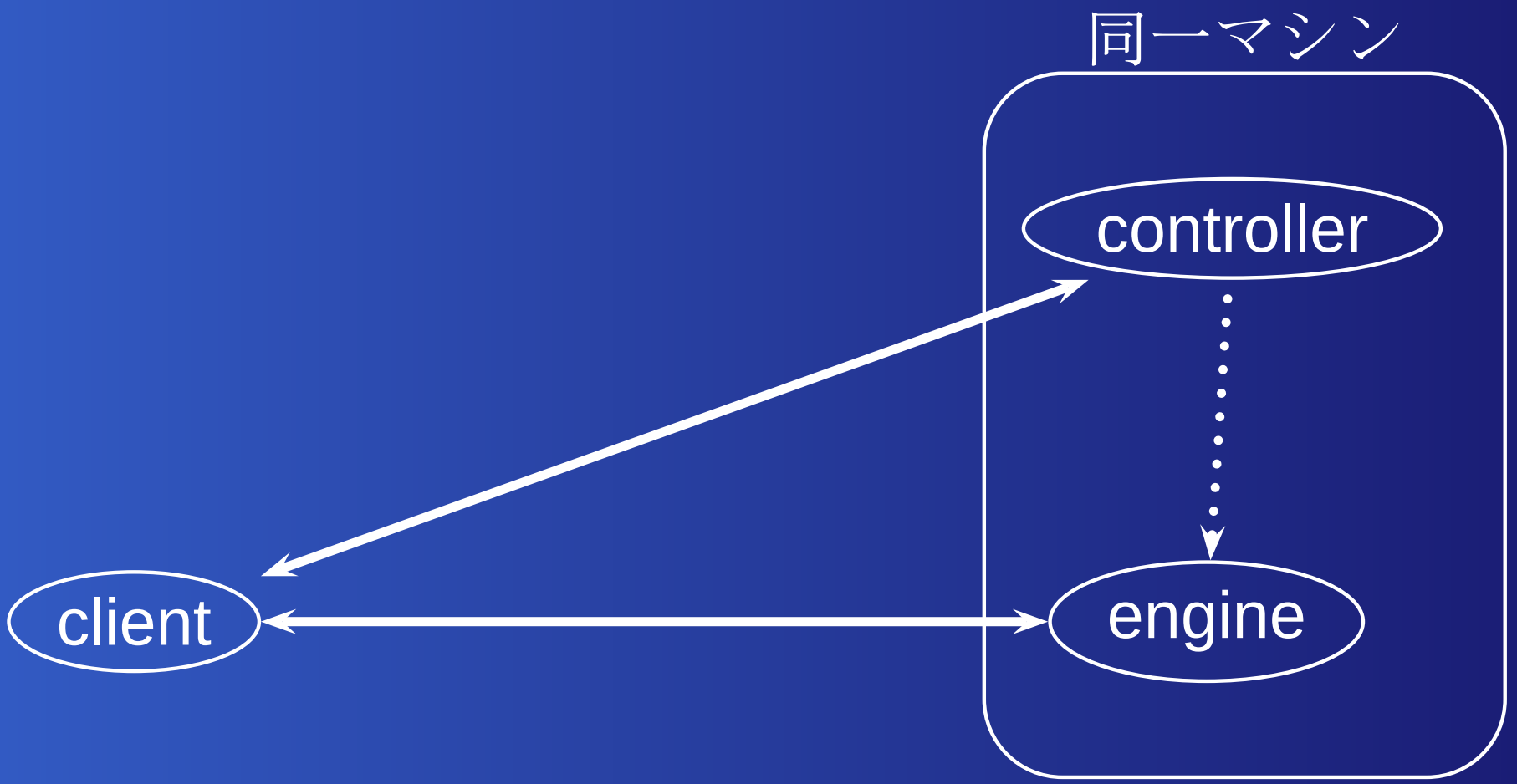
概念図



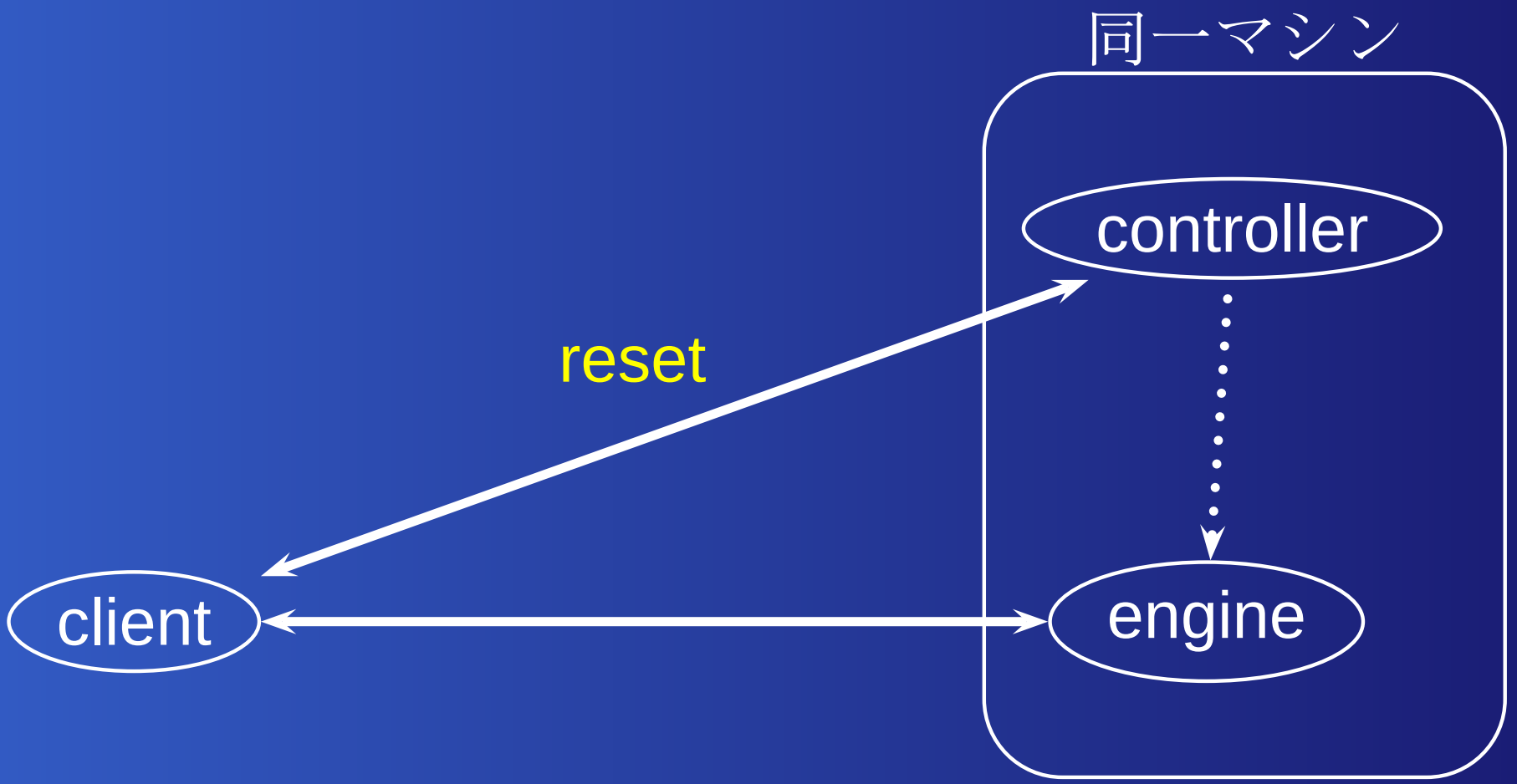
OpenXM の仕組み

- サーバ・クライアント方式
- ネットワーク透過性がある
- サーバはコントローラとエンジンの 2 プロセスからなる
- エンジンはスタックマシン
- 計算を途中で中断できる

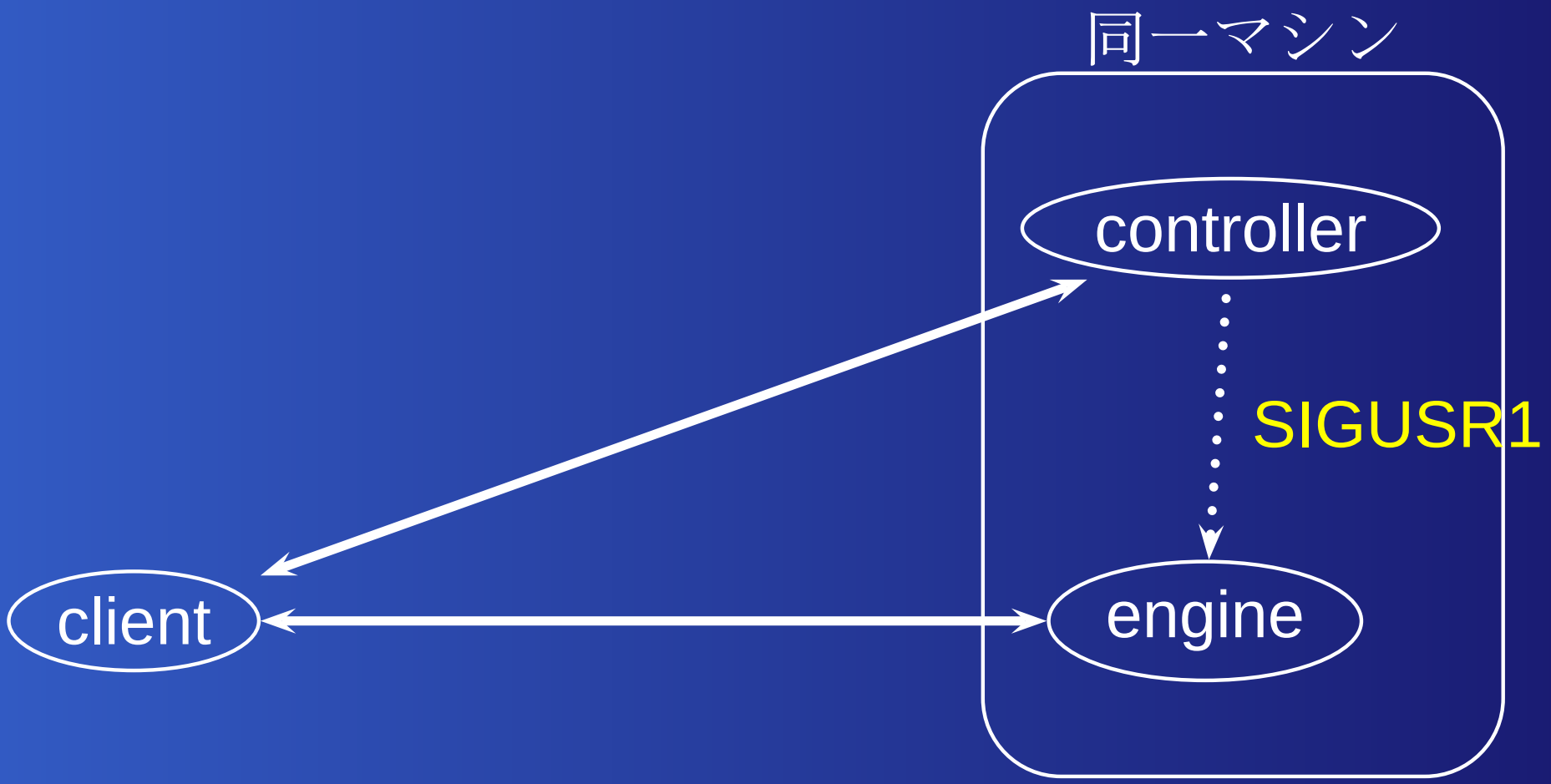
OpenXM の計算中断機能



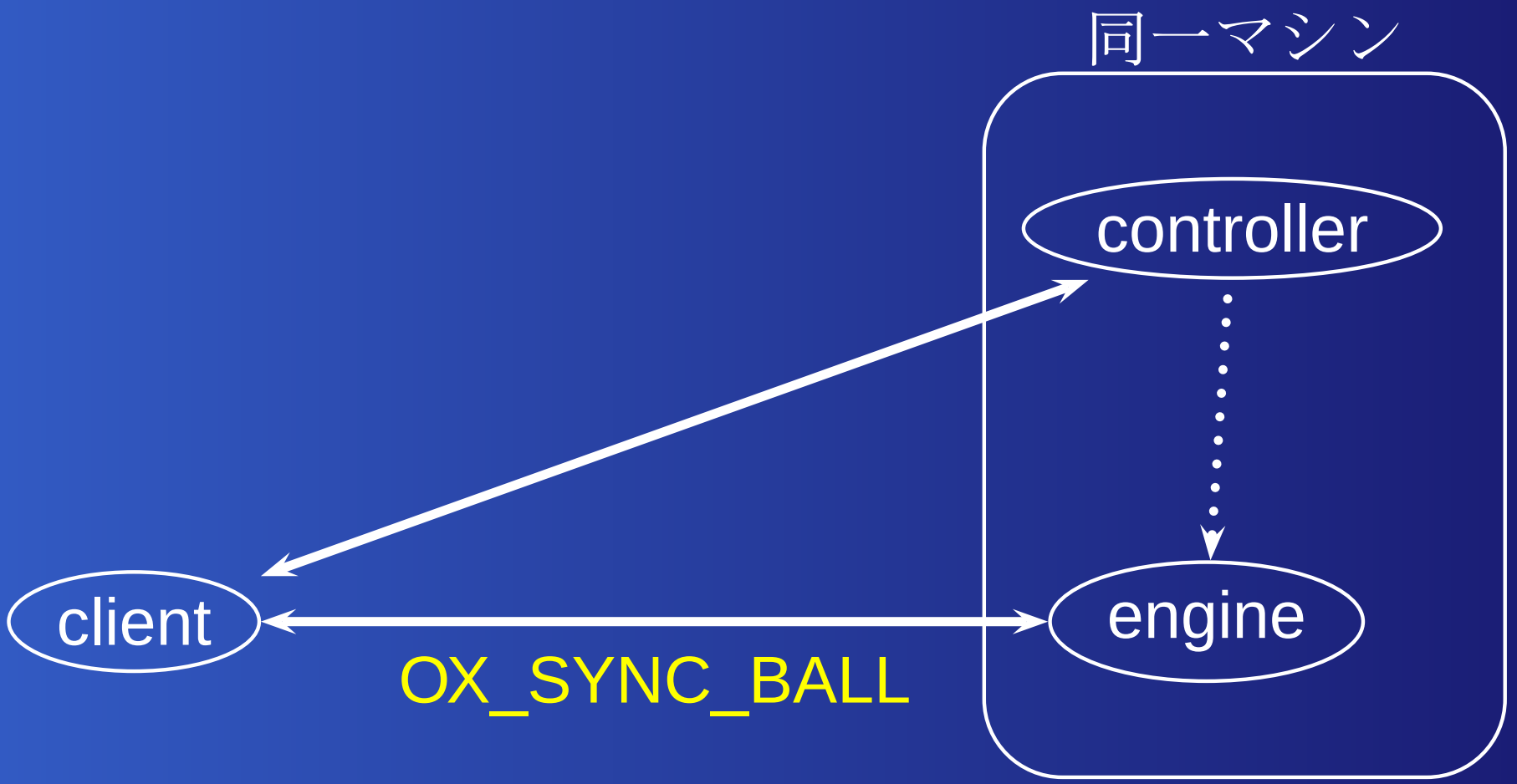
OpenXM の計算中断機能



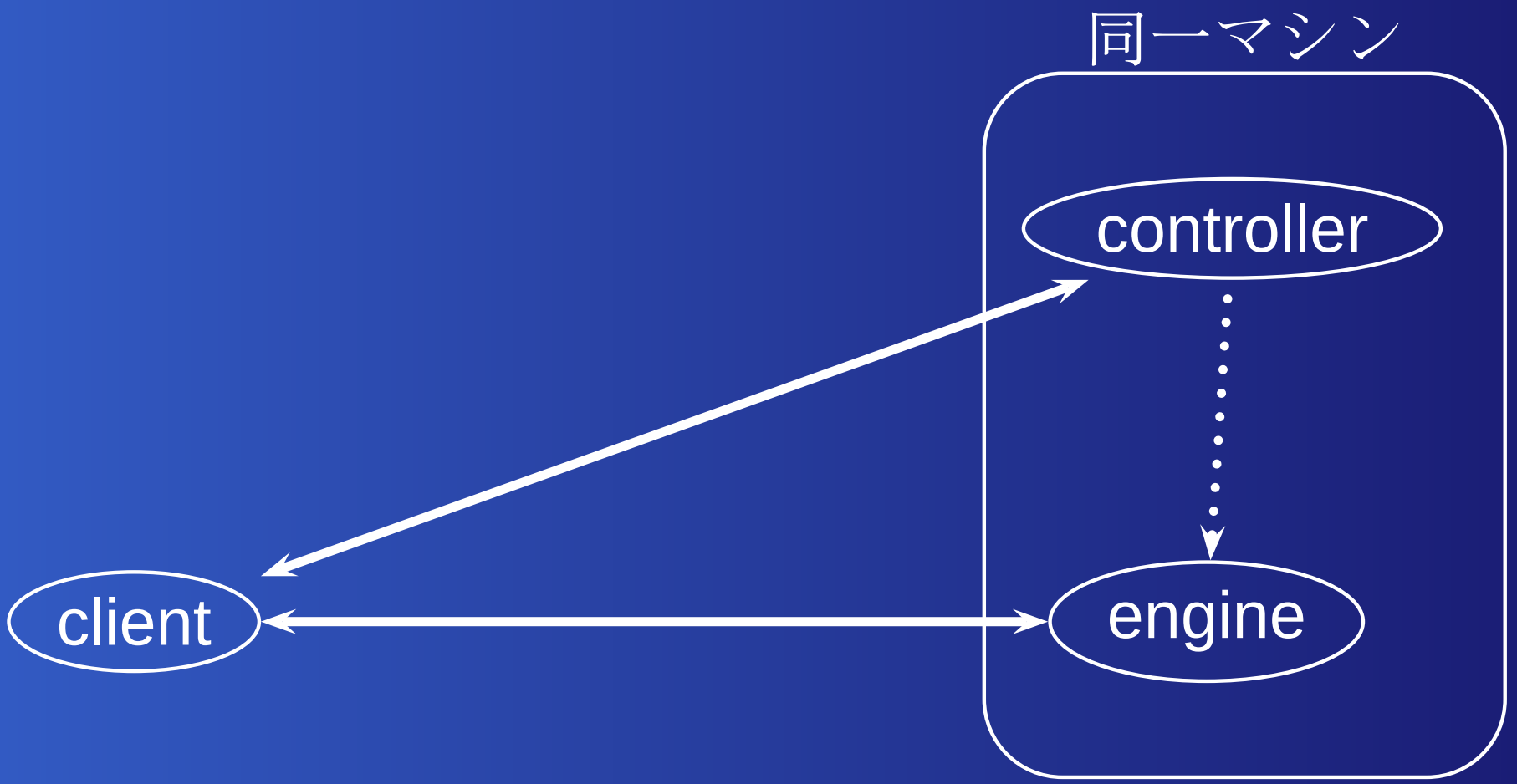
OpenXM の計算中断機能



OpenXM の計算中断機能



OpenXM の計算中断機能



MathLink の仕組み

- (ほとんど) ネットワーク透過性がある

MathLink の仕組み

- (ほとんど) ネットワーク透過性がある
- 通信路で送られるのは Mathematica の式

MathLink の仕組み

- (ほとんど) ネットワーク透過性がある
- 通信路で送られるのは Mathematica の式

例:
```
EvaluatePacket[Sin[$VersionNumber]]
ReturnPacket[Sin[x]]
InputNamePacket["In[1]:= "]
MenuPacket[1,"Interrupt> "]
```

MathLink の仕組み

- (ほとんど) ネットワーク透過性がある
- 通信路で送られるのは Mathematica の式

例:
```
EvaluatePacket[Sin[$VersionNumber]]
ReturnPacket[Sin[x]]
InputNamePacket["In[1]:= "]
MenuPacket[1,"Interrupt> "]
```

- *Packet[] を Mathematica Book ではパケットと呼ぶ

MathLink プログラムの書き方 (1)

```
char *s = "Mathematicaの式";
MLPutFunction(link, "EvaluatePacket", 1)
MLPutFunction(link, "ToExpression", 1);
MLPutString(link, string);
MLEndPacket(link);
```

MathLinkプログラムの書き方(2)

```
char *s;
while (MLNextPacket(link) != RETURNPKT)
    MLNewPacket(link);
switch(MLGetNext(link)) {
MLTKSTR:
    MLGetString(link, &s);
    ...
MLTKINT:
}
MLNewPacket(link);
```

Mathematica の割り込み

Mathematica の割り込み

Mathematica の割り込み

Mathematica の割り込み

MLPutMessage

- MathLink の非公開関数
- ネットワーク透過性はない

MLPutMessage

- MathLink の非公開関数
- ネットワーク透過性はない
- MLPutMessage(link, MLInterruptMessage) で link の指すプロセスに SIGINT を送る。(Unix の場合)

MLPutMessage

- MathLink の非公開関数
- ネットワーク透過性はない
- MLPutMessage(link, MLInterruptMessage) で link の指すプロセスに SIGINT を送る。(Unix の場合)
- 割り込んだあとの後始末は、相手方のプログラムの作りに依存する

割り込みの後始末

- MenuPacket[1,"Interrupt> "] を受け取れば計算が中断されている

割り込みの後始末

- MenuPacket[1,"Interrupt> "] を受け取れば計算が中断されている
- MLPutString("\n")

割り込みの後始末

- MenuPacket[1,"Interrupt> "] を受け取れば計算が中断されている
- MLPutString("\n")
- MenuPacket[0,"Interrupt> "] を受け取る

割り込みの後始末

- MenuPacket[1,"Interrupt> "] を受け取れば計算が中断されている
- MLPutString("\n")
- MenuPacket[0,"Interrupt> "] を受け取る
- MLPutString("a")

割り込みの後始末

- MenuPacket[1,"Interrupt> "] を受け取れば計算が中断されている
- MLPutString("\n")
- MenuPacket[0,"Interrupt> "] を受け取る
- MLPutString("a")
- TextPacket["..."] を受け取る

計算結果 $Aborted

- ReturnPacket[$Aborted] が返ってくるか?
  3.x ⇒ ○,　　4.x ⇒ ×

計算結果 $Aborted

- ReturnPacket[$Aborted] が返ってくるか?
  3.x ⇒ ○,　　4.x ⇒ ×
- 4.x では、次の計算を行うと、$Aborted が返ってくる!!

計算結果 $Aborted

- ReturnPacket[$Aborted] が返ってくるか?
  3.x ⇒ ○,　　4.x ⇒ ×
- 4.x では、次の計算を行うと、$Aborted が返ってくる!!
- “割り込みの後始末” のあと、EvaluatePacket[0] を送って、ReturnPacket[...] をふたつ受け取るとよい。最初のものが $Aborted

参考文献

[1] Mathematica Book, Wolfram Research

[2] MathArchive にあった Todd Gayley のメール ([mg17015], 1999/Apr)

[3] 昔の MathLink にあった MLSignal( ) の解説 (Google のキャッシュにあった)

[4] mathlink.h, nm libMLa の出力, mprep の生成するソース