

リフレクションを用いた 拡張可能なオペレーティングシステムの構成法

岡坂 史紀 上野 球
任天堂株式会社 技術研究グループ
2007年1月30日

発表の概要

- ・ 研究の背景
- ・ 拡張可能なオペレーティングシステム
- ・ コンポーネントオペレーティングシステム
- ・ es オペレーティングシステム
- ・ まとめと今後の課題

研究の背景

“Components: a revolution in application development. But most operating systems aren't designed to host component based applications.”

– Noah Mendelsohn, 1997

“There has been much talk about component architectures but only one true success: Unix pipes. It should be possible to build interactive and distributed applications from piece parts.”

– Rob Pike, 2000

“With the enormous growth of the worldwide web and the growth of standards for it like HTTP, HTML, XML, etc., the importance of a new OS structure was considerably diminished.”

– Jim Mitchell, 2001

拡張可能なオペレーティングシステム

```
Handle<IStream> stream;  
char buf[SIZE];  
  
stream = directory->bind("file", 0);  
stream->write(buf, sizeof buf);
```

```
[object, uuid(032c16c8-25db-11db-9c02-0009bf000001)]  
interface IStream : IInterface  
{  
    long getPosition();  
    void setPosition(long pos);  
    long getSize();  
    void setSize(long size);  
    int read([out, size_is(count)] void* dst, int count);  
    int read([out, size_is(count)] void* dst, int count, long offset);  
    int write([in, size_is(count)] void* src, int count);  
    int write([in, size_is(count)] void* src, int count, long offset);  
    void flush();  
}
```

```
IBinding, ICache, IClassFactory,  
IClassStore, IContext,  
ICurrentProcess, ICurrentThread,  
IFile, IInterface, IInterfaceStore,  
IIterator, IMonitor, IPageable,  
IPageSet, IProcess, IRuntime, IStream,  
IThread, and so on.
```

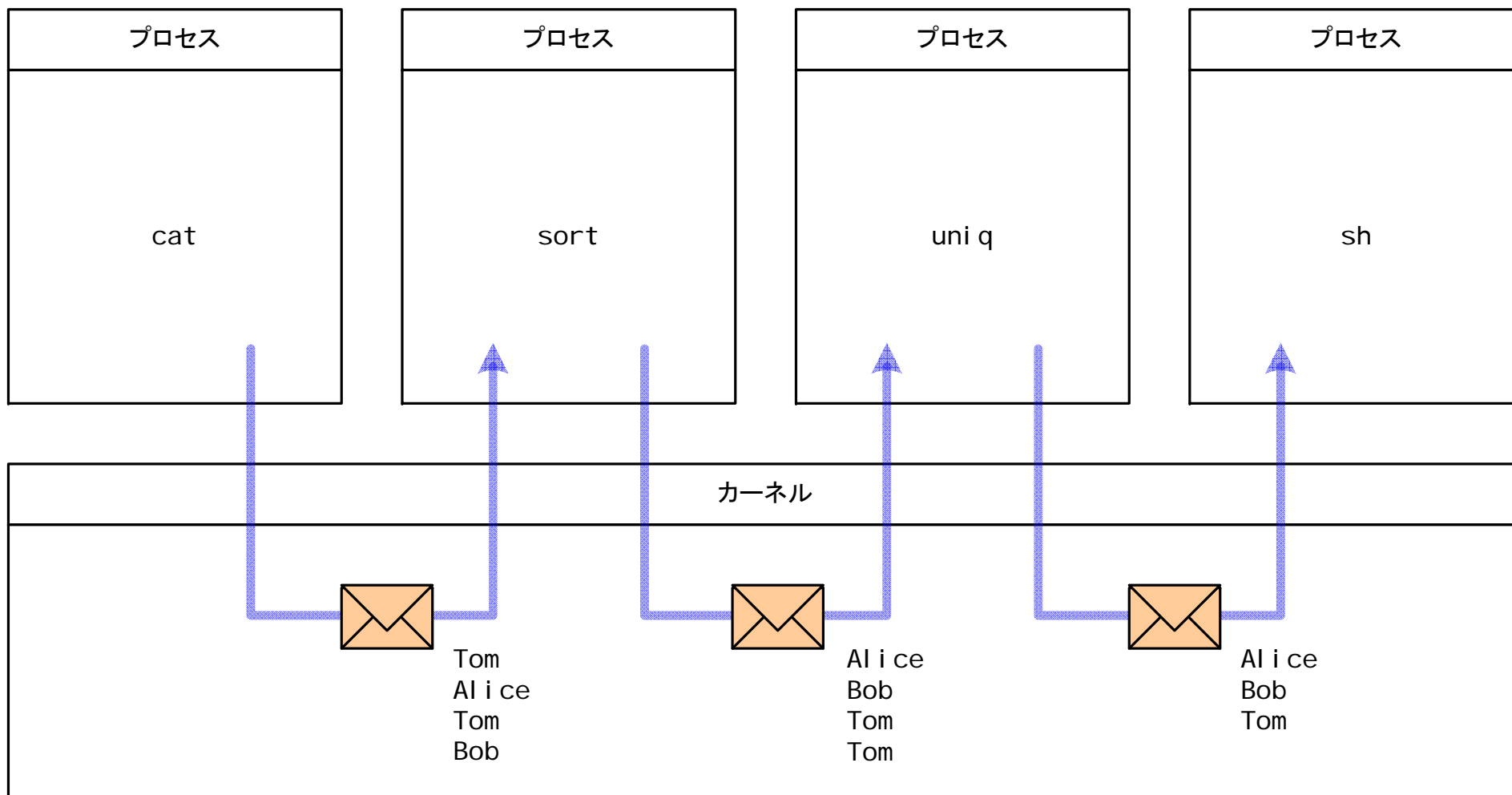
```
creat();  
create();  
NewWindow();  
NewCWindow();  
CreateWindow();  
CreateWindowEx();
```

```
int fd;  
char buf[SIZE];  
  
fd = creat("file", 0700);  
write(fd, buf, sizeof buf);
```

If I had to do it over again? Hmm... I guess I'd spell 'creat' with an 'e'.

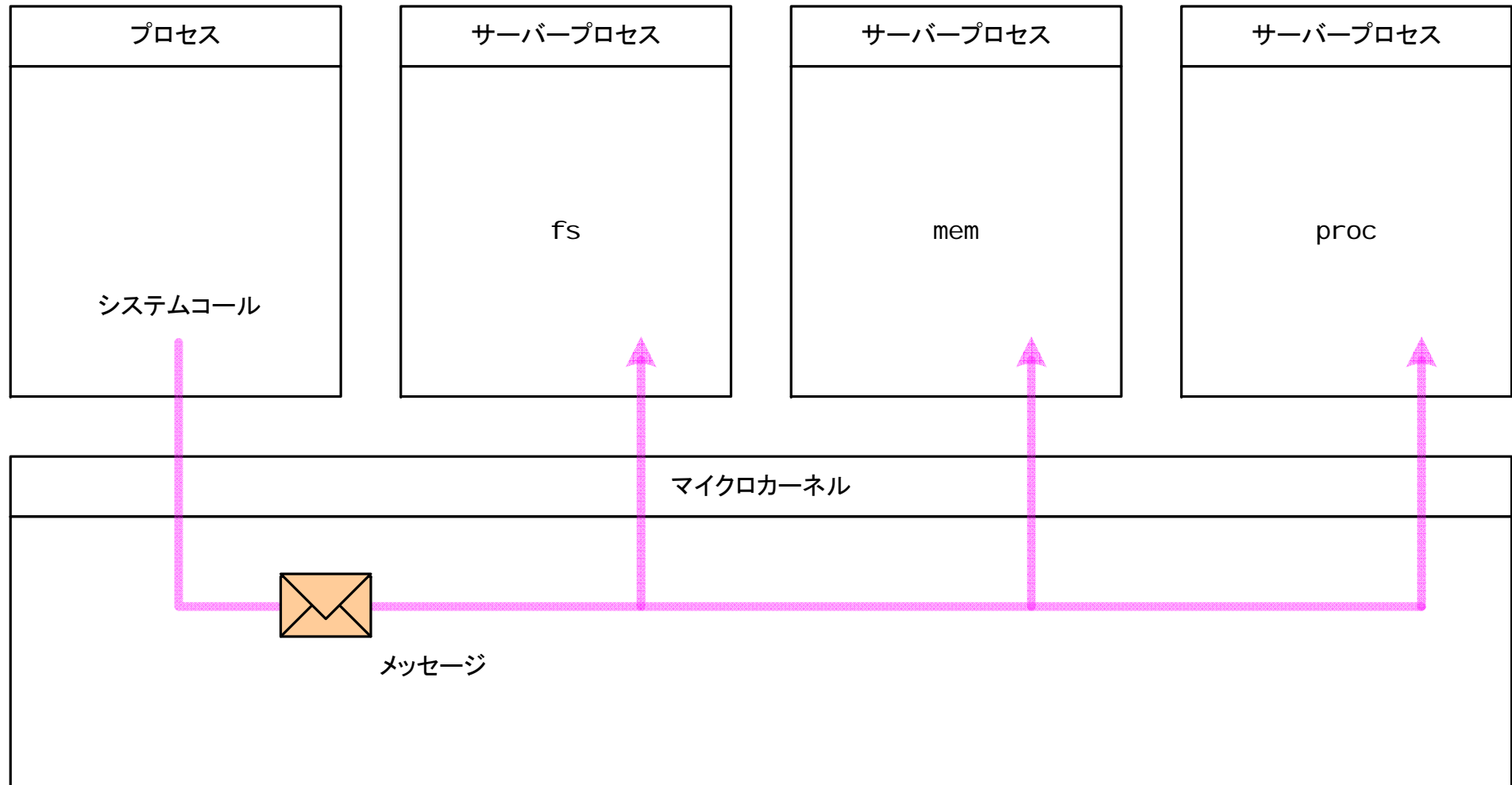
– Ken Thompson

パイプ



```
$ cat list | sort | uni q
```

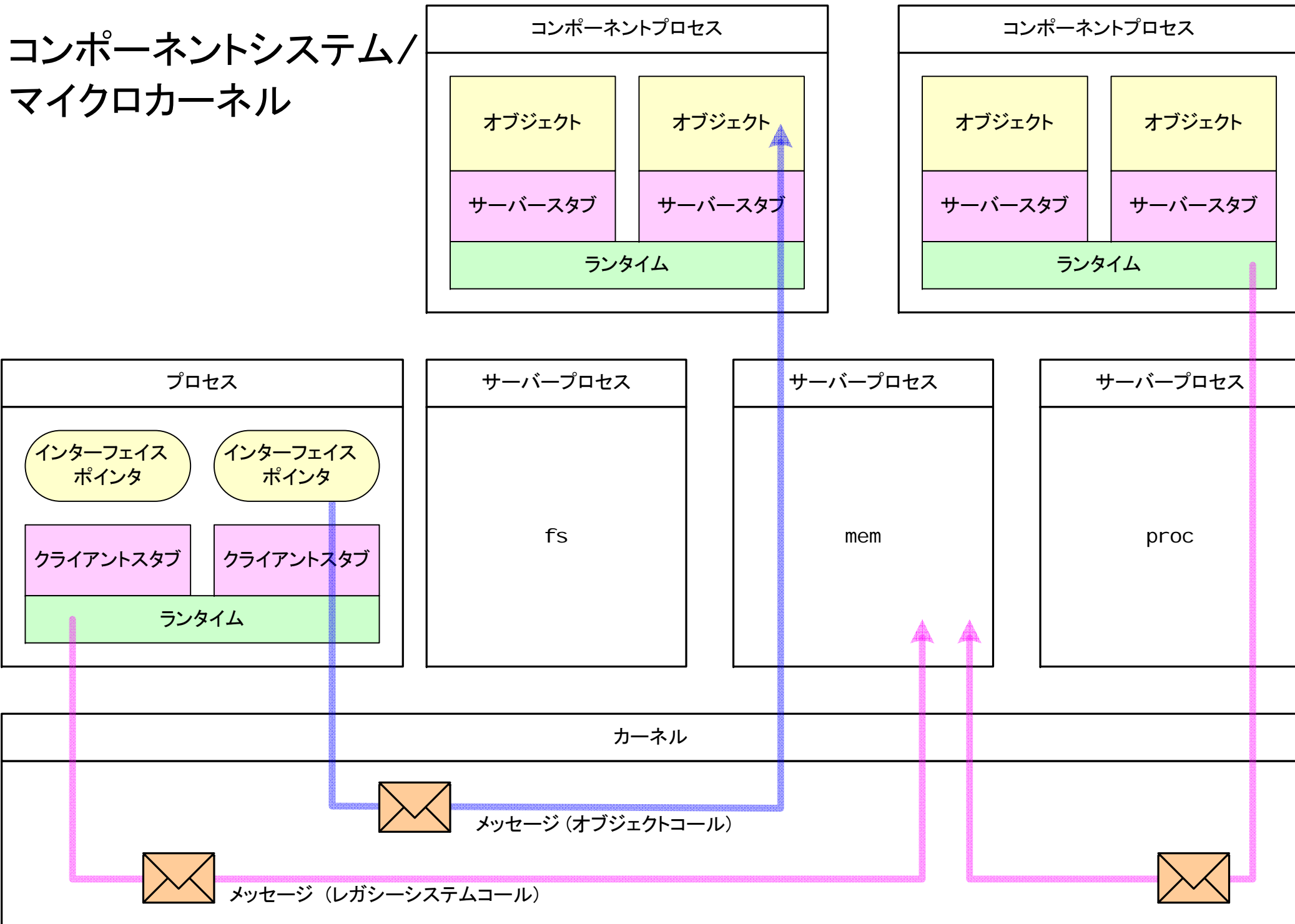
マイクロカーネル



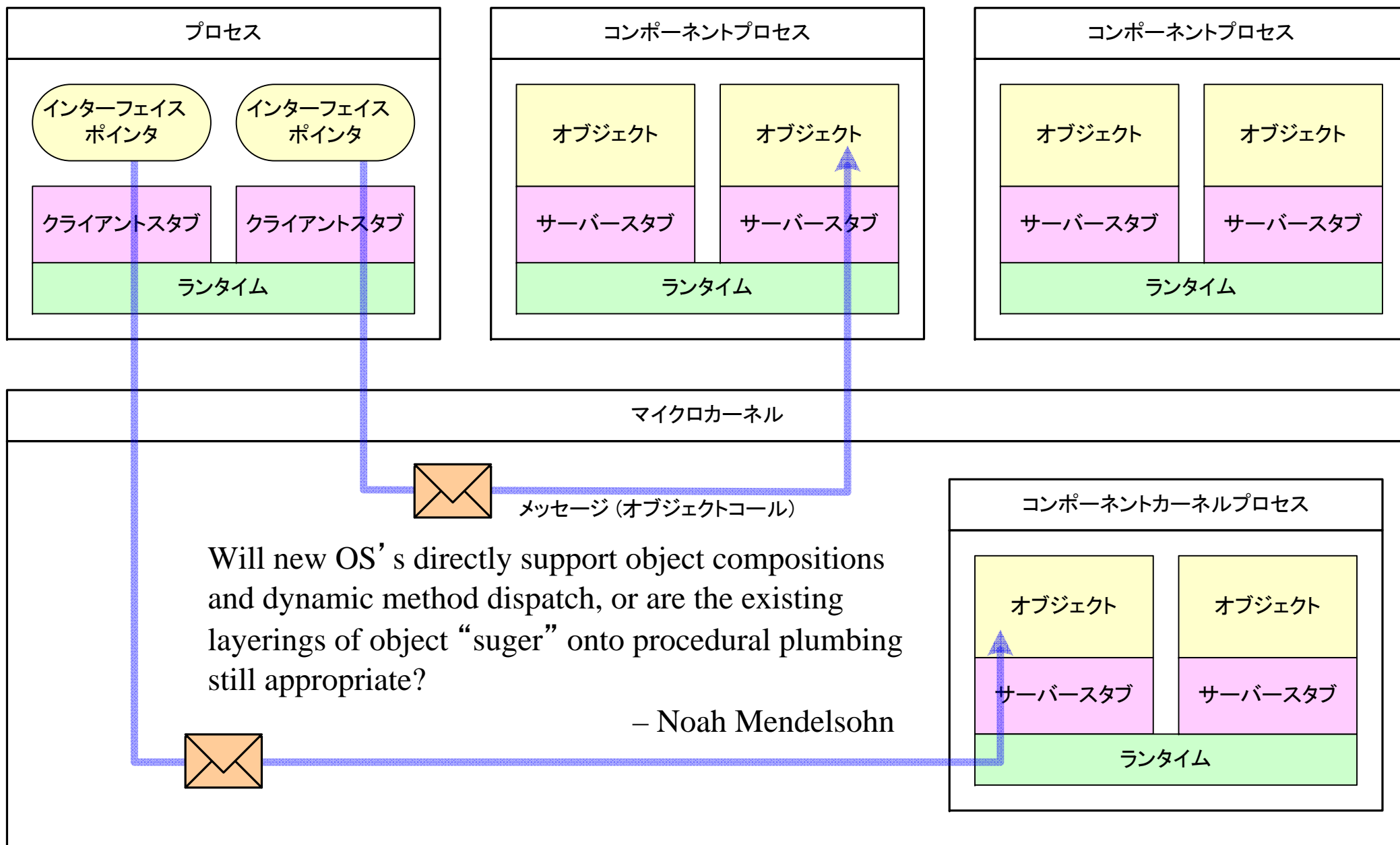
New operating systems today tend to be just ways of reimplementing Unix. How can operating systems research be relevant when the resulting operating systems are all indistinguishable?

- Rob Pike

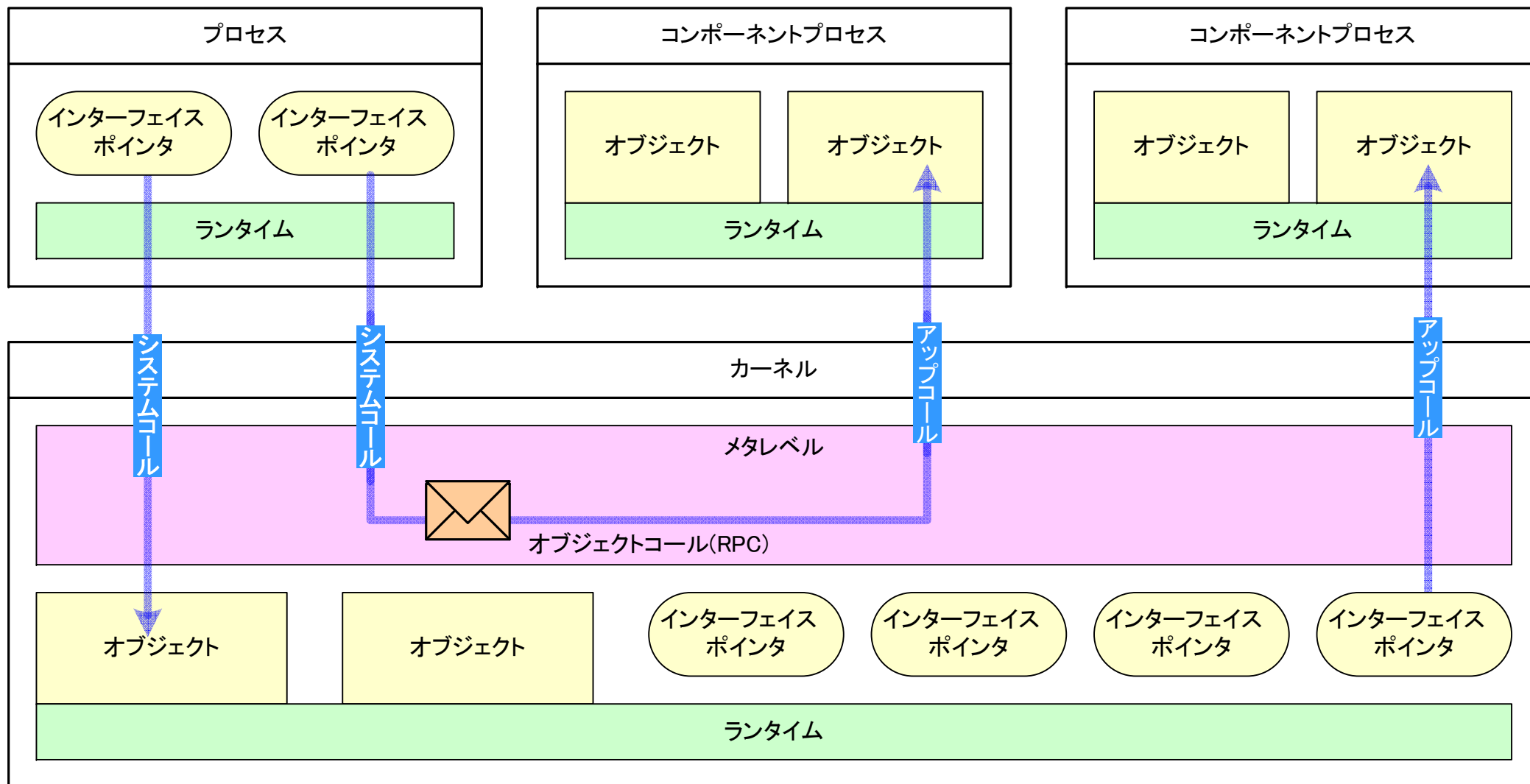
コンポーネントシステム/ マイクロカーネル



カーネルをコンポーネントフレームワークに統合するには？



コンポーネントオペレーティングシステム



C++の発展。C++0xにむけて。

199x

```
void setChannels(IStream* stream, int channels)
{
    IAudi oFormat* dsp;

    stream->queryInterface( IID_I Audi oFormat,
                          reinterpret_cast<void**>(&dsp));
    dsp->setChannels(channels);
    dsp->release(); // この行がないとメモリリークする
}
```

- ISO C++: ISO/ICE 14882: 1998

200x

```
void setChannels(Handle<IStream> stream, int channels)
{
    Handle<IAudi oFormat> dsp = stream;
    dsp->setChannels(channels);
}
```

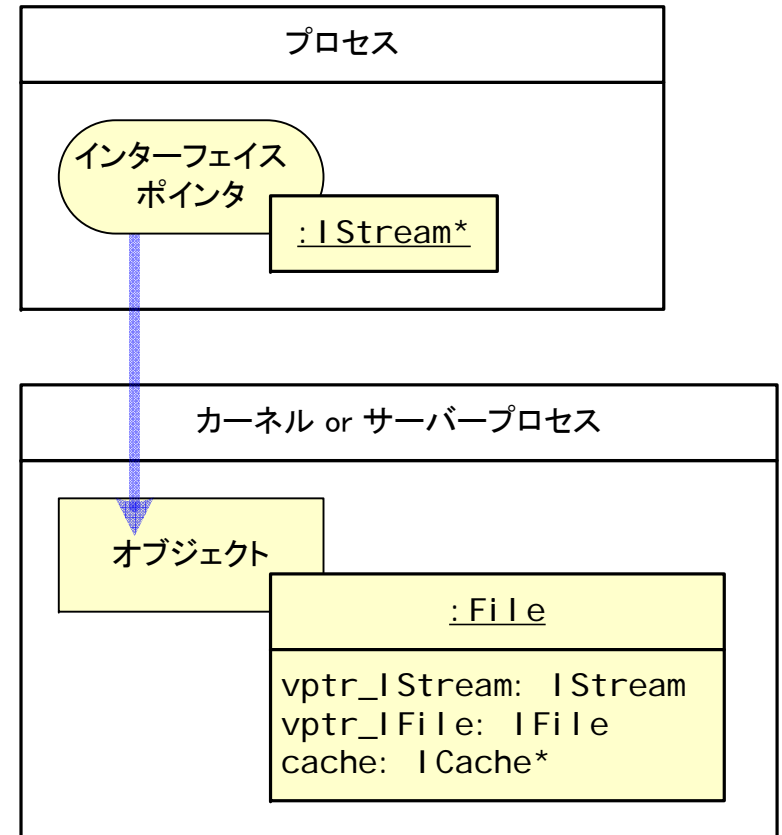
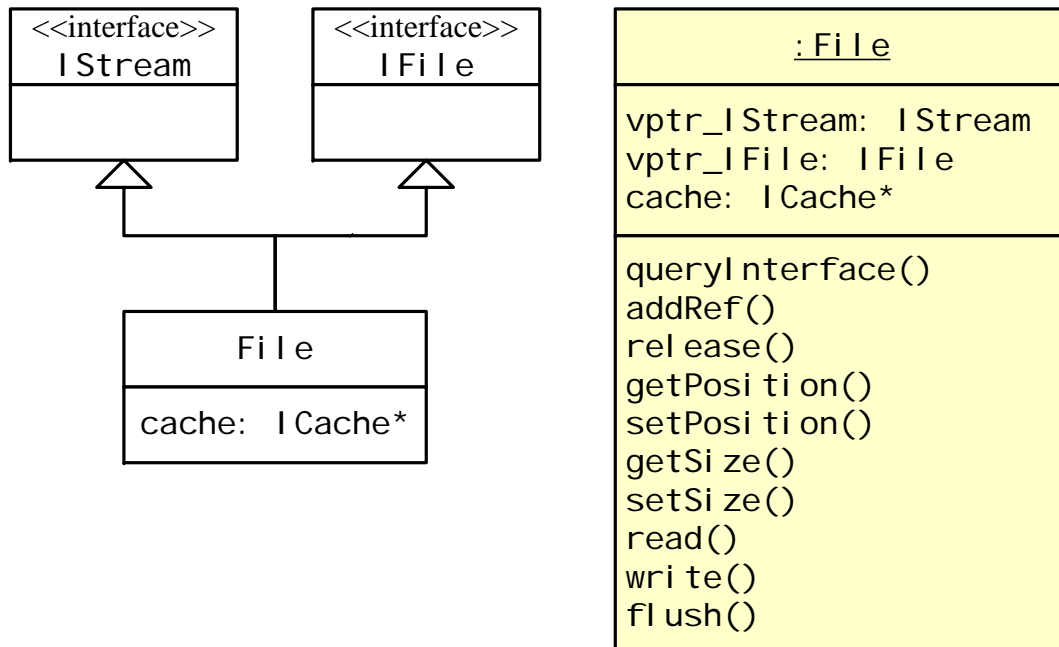
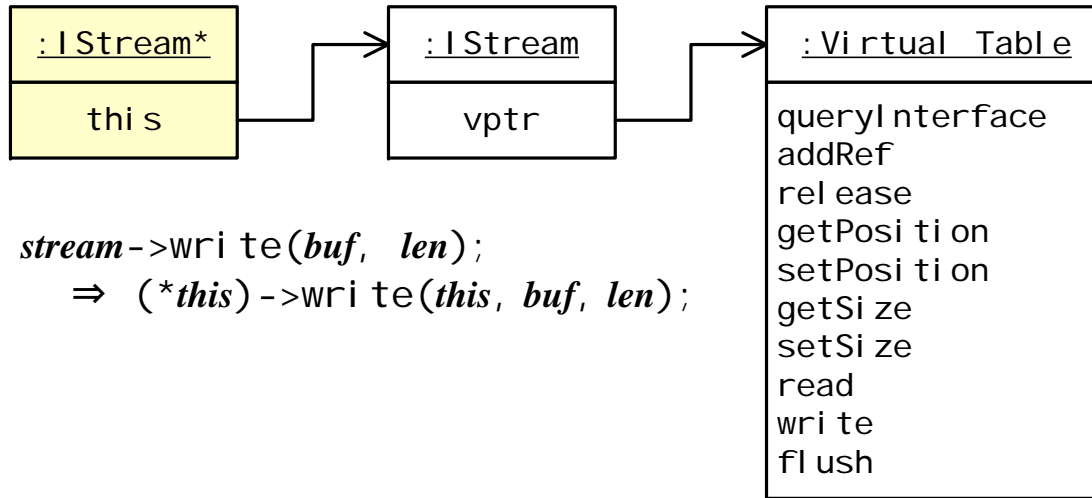
- Itanium C++ ABI: 2001
- ISO C++ 2nd ed.: ISO/ICE 14882: 2003
- C++ TR1: ISO/IEC DTR 19768: 2005

```
template<class I>
class Handle
{
    ...
    template<class J>
    Handle(const Handle<J>& comptr)
    {
        IInterface* unknown = comptr.get();
        object = cast(unknown, IInterface<J>());
    }
    ...
};
```

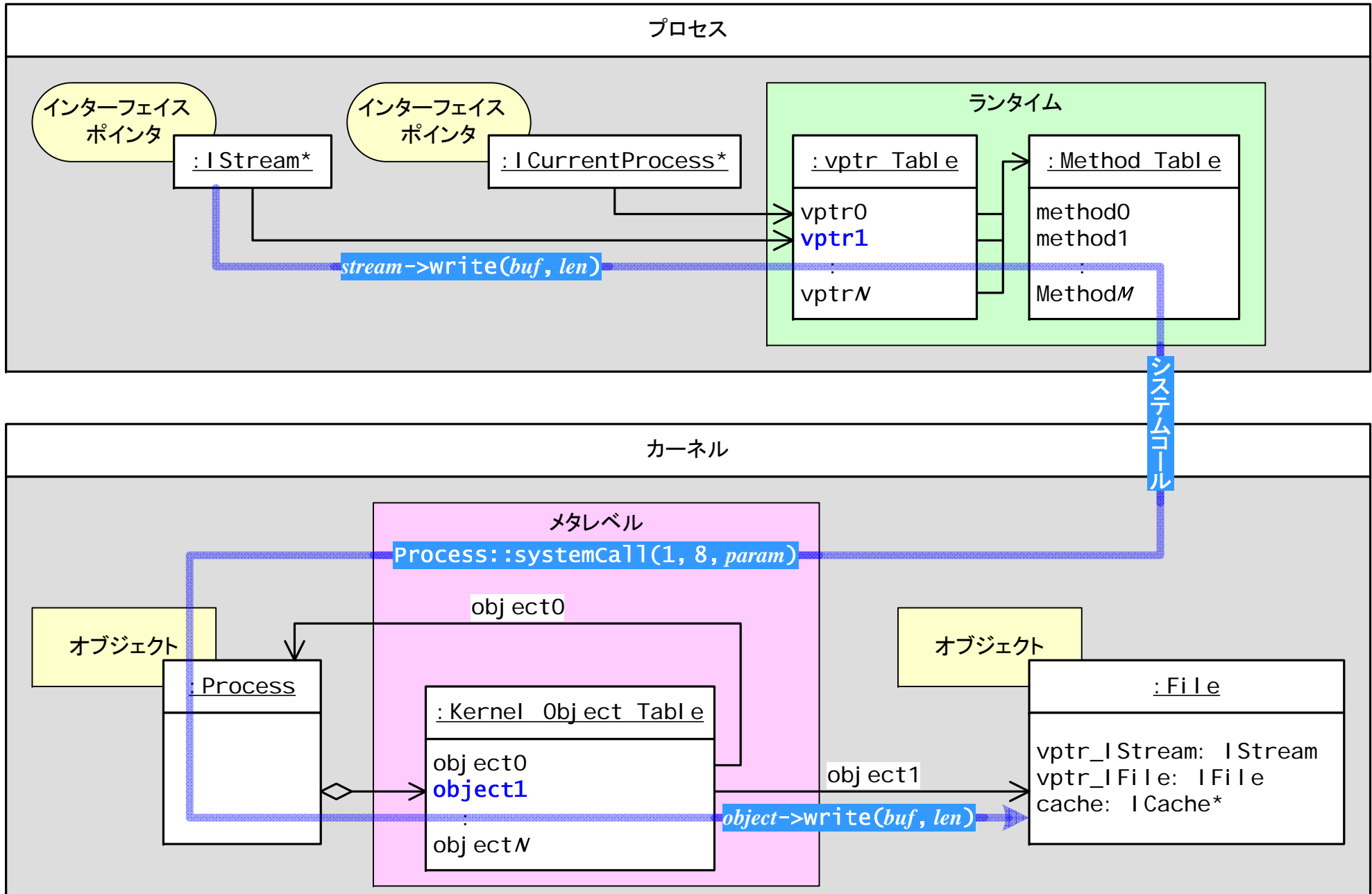
メンバーテンプレート. 1994.

型特性 (type traits). C++ TR1, 2005.

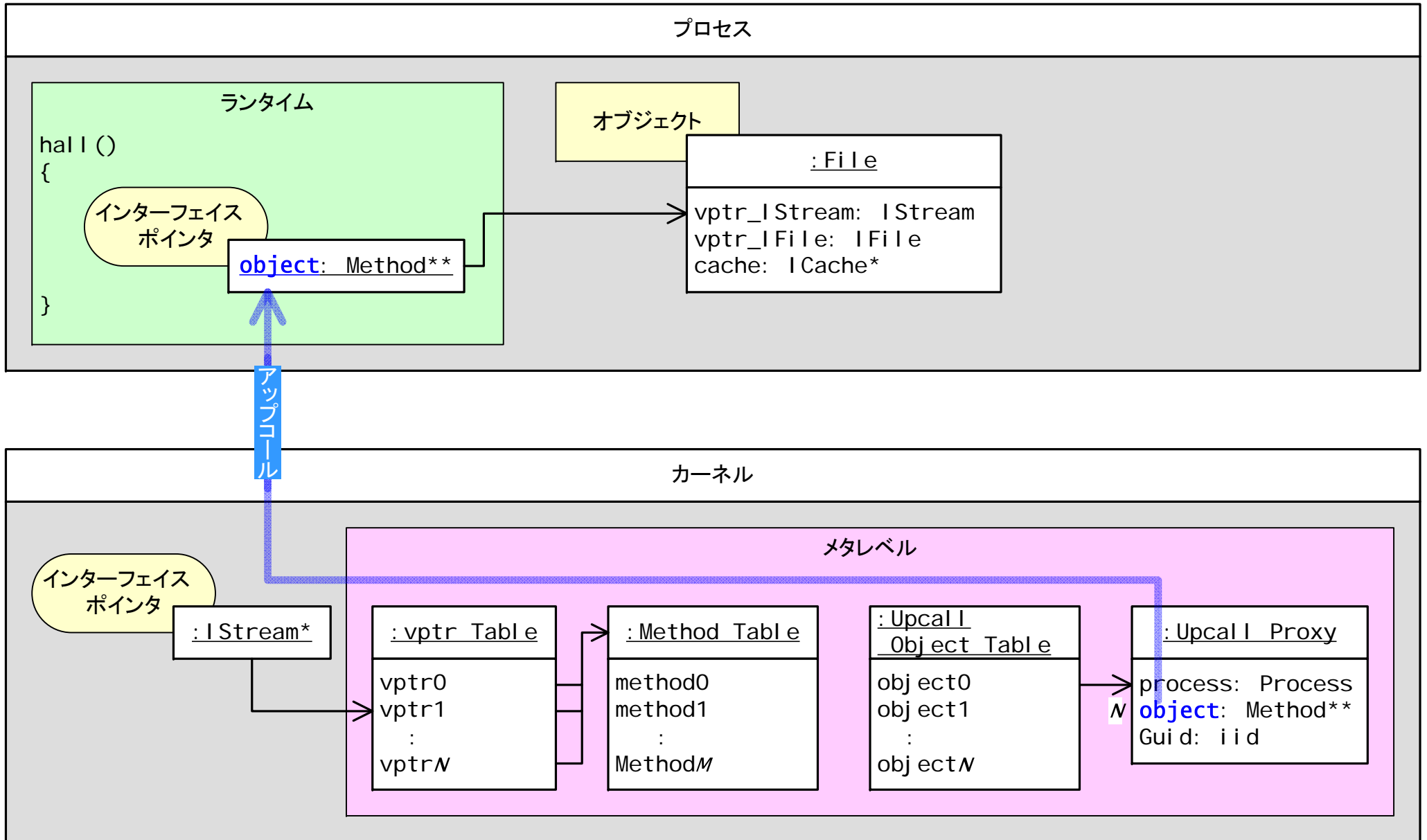
C++ オブジェクトモデル



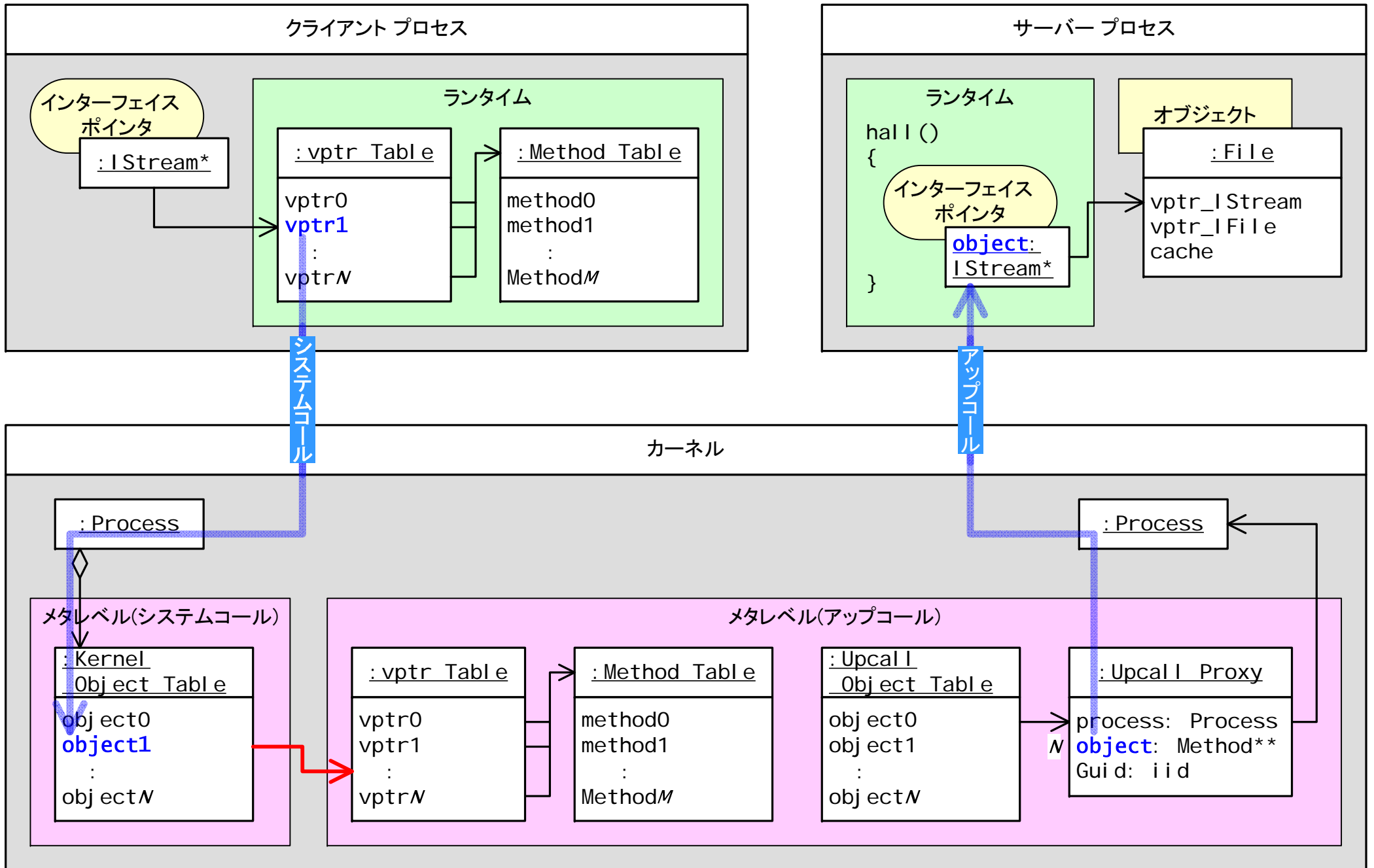
システムコール



アップコール



オブジェクトコール = システムコール + アップコール



es オペレーティングシステム

オープンソース:

- <http://sourceforge.jp/projects/nes/>

標準のサポート:

- Intel MultiProcessor Specification Version 1.4
- FAT ファイルシステム
- ISO 9660 ファイルシステム
- TCP/IP プロトコル

ビルドシステム:

- gcc-4.1.1
- binutils-2.6.1
- newlib-1.14.0

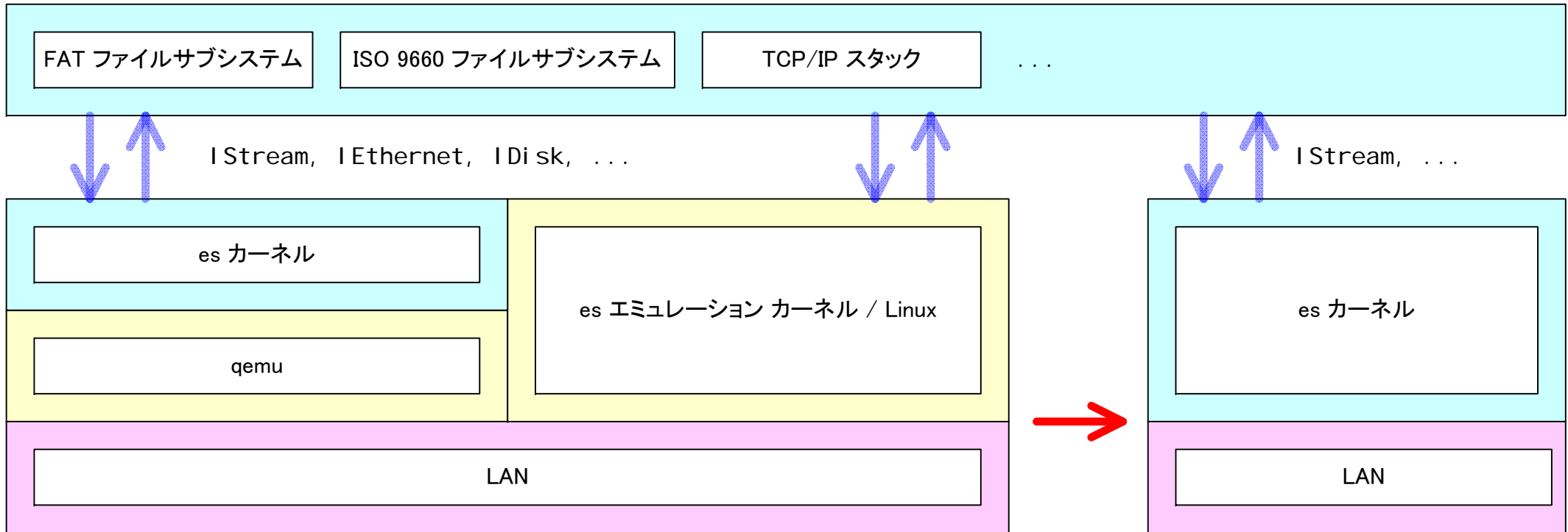
テストシステム

- qemu, Microsoft Virtual PC
- Linux

Copyright © 2006, 2007
Nintendo Co., Ltd.

Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Nintendo makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

es カーネル / Linux



まとめと今後の課題

es オペレーティングシステム

- Xen, PC実機等への対応

標準のインターフェイスセットの作成

- OSKit, MMLite, BREW, XPCOM, etc.

スクリプト言語との統合

- ECMAScript, etc.

分散環境への適用

- 3Dコンピューティング環境(cf. Croquet, etc.)