

IIJによるmruby事例紹介

株式会社インターネットイニシアティブ

@rev4t (Takashi Sogabe)

自己紹介

- 曽我部 崇 (Takashi Sogabe)
 - 企業向けルータ SEIL(ザイル)の開発
 - 「戦略的開発部」..という部署で日々面白そうなサービスを考えています

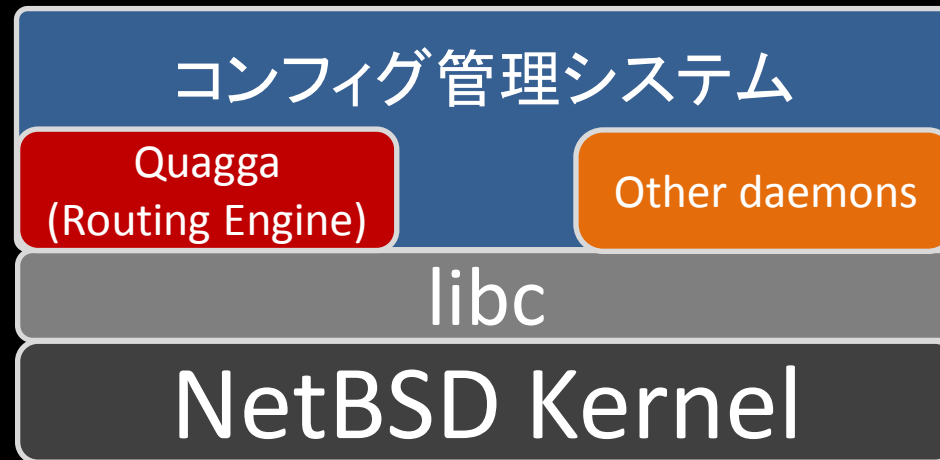
SEIL(ザイル)作っています

- ISPのノウハウを結集してIIJが開発した企業向けアクセスルータ
- Firmwareは**独自開発**、Hardwareは国内製造
- 仮想化環境(VMWare, KVM, Hyper-V)対応



SEILのアーキテクチャ

- NetBSDをベースに、ルーティングエンジンやコンフィグ管理システム(CLI, WebUI)を組み込んでいる



組込ソフト開発者の苦悩

```
static int
load_from_file(char *fname, char **bufp)
{
    FILE *fp = NULL;
    char *buff = NULL;
    size_t len;
    int nread = -1;

    fp = fopen(fname, "r");
    if (fp == NULL) {
        fprintf(stderr, "failed to open¥n");
        nread = -1;
        goto failure;
    }
    fseek(fp, 0, SEEK_END);
    len = ftell(fp);
    fseek(fp, 0, SEEK_SET);
    buff = malloc(len + 1);
    if (buff == NULL) {
        fprintf(stderr, "malloc failed");
        nread = -1;
        goto failure;
    }
}
```

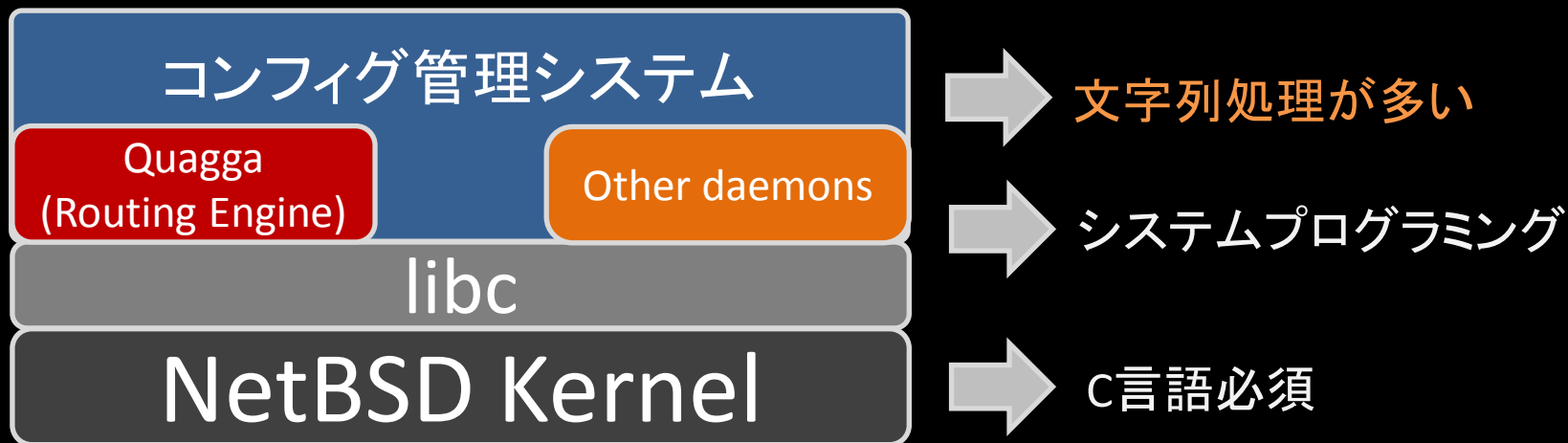
```
    nread = fread(buff, 1, len, fp);
    if (nread < len && !feof(fp)) {
        fprintf(stderr, "fread() failed¥n");
        nread = -1;
        goto failure;
    }
    *(buff + len) = '¥0';
failure:
    if (buffp)
        *buffp = buff;
    if (fp)
        fclose(fp);

    return nread;
}
```

C言語でコード書くのは結構大変

- エラー処理のコードが煩雑
- 特に文字列処理が大変
 - 文字列領域の踏み抜き..
 - 確保したメモリの解放忘れてメモリリーク..
 - FDの解放し忘れてFDリーク..
 - etc...

mrubyを使えば開発効率が上がる?



Config Management System is written in ruby and development efficiency will be higher

事例紹介 1

バックボーン遅延測定
「Quill」

SEIL/X1 ハードウェア構成

- CPU: Cavium Octeon (SoC)
 - MIPS64アーキテクチャ
- Memory: 256MB
- Flash: 16MB
- USB: 1ポート
- Ethernet: GbE 3-port

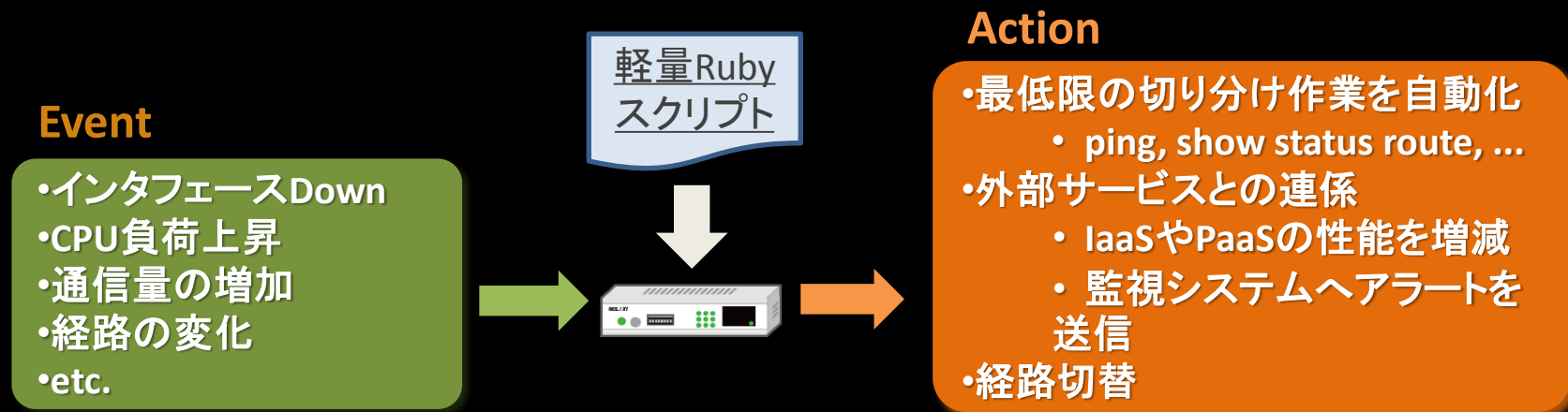


事例紹介 2

柔軟なルータ
「SEIL(ザイル)」

もっと柔軟なルータが欲しい

- イベントスクリプト機能 開発中
 - SEILの機能をmrubyスクリプトから操作
 - 経路障害、インタフェースのUp/Downなどのイベントをトリガになる

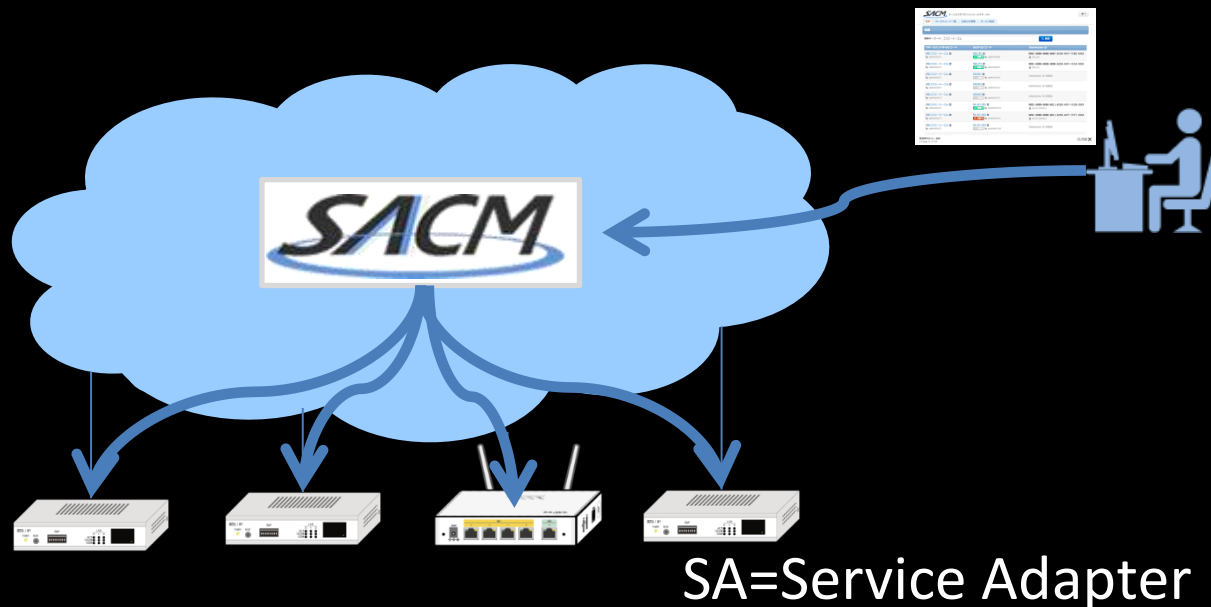


事例紹介 3

ネットワーク管理サービス
「SACM」

SACMとは

- SACM [Service Adapter Control Manager]
 - 機器の自動接続、一元管理ができるサービス
 - <http://www.sacm.jp/>



サービスの特長

- 「富山の置き薬」タイプのネットワーク接続サービス
 - ー 機能自体に課金
 - ・ ネットワーク機器「SA-W1」は無償で提供
 - ・ IPsec-VPNレシピ、又は無線LANレシピ
 - ー 必要なネットワークを迅速に構築できる



SA-W1にmrubyを搭載しました

- コンフィグ管理システム「レシピフレームワーク」はmrubyを用いて開発されています



使ってみた感想

- 開発効率が上がった

- 文字列処理やロジックが簡潔に記述できるようになった
- いちいちfirmwareをビルドしなくてもプログラムを差し替えるのは便利

組込機器の開発プロセスが進化する

- 今までの組み込み機器開発
 - ドライバ、アプリ(ロジック)全てC言語又はアセンブリ言語で開発
- これからの方向性
 - 速度やリソースの制約がある箇所だけC言語やアセンブリ言語で記述
 - その他のロジック部分は、出来る限り開発効率の良い言語で開発
 - テスト駆動開発

スクリプト言語が熱い!

- NetBSDカーネルでLuaが動く
 - Rapid Application Development
 - Better Configuration
 - Modifying software in C is hard for users
 - <http://bit.ly/WD6gzd>
- そのうち、mrubyが OS Kernel上で動くなる
日が来るかも..