

# カーネルサイズ・使用メモリ検証／改善PJ (Status update)

2006年3月24日

NEC Linux推進センター

池田 宗広

m-ikeda(( at mark ))ds.jp.nec.com



1. プロジェクトの目的
2. 実行項目
3. 実行ステップ
4. 現在のStatus
5. 自動検証ツール クイックガイド
6. Step1 結果の見方
7. Step 2
8. 今後の方針
9. まとめとお願い



# 1. プロジェクトの目的

- **組み込み領域におけるLinuxカーネルへの要求**
  - サイズ・使用メモリを極力最小化したカーネルを作成したい。
  - カーネルのサイズ・使用メモリ量を改善したい。
- **要求実現へのアプローチ**
  - サイズ・使用メモリを最小化したカーネルを作成するためには、カーネルconfigの絞りこみが最重要。
  - サイズ・使用メモリを改善するためには、改善すべき部分の明確化が必要。  
「改善すべき部分」の大部分はconfig項目にマッピングできる。

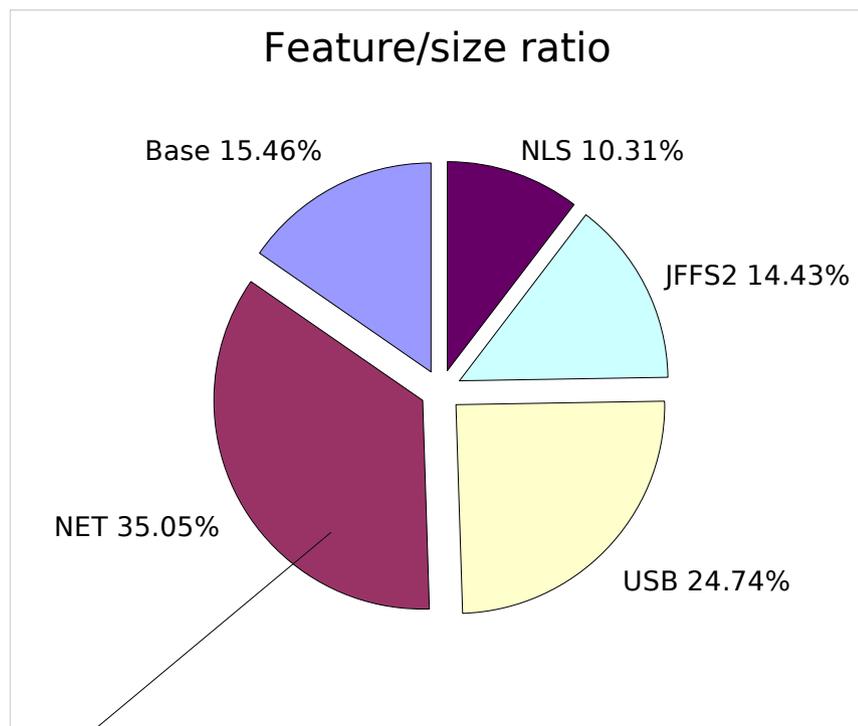
→本プロジェクトの目的は、

各config項目がサイズ・使用メモリに及ぼすインパクトを検証・公開し、  
サイズ・使用メモリ最適化構成および改善の材料を提供すること。  
実際の改善まで行えればなお良し。

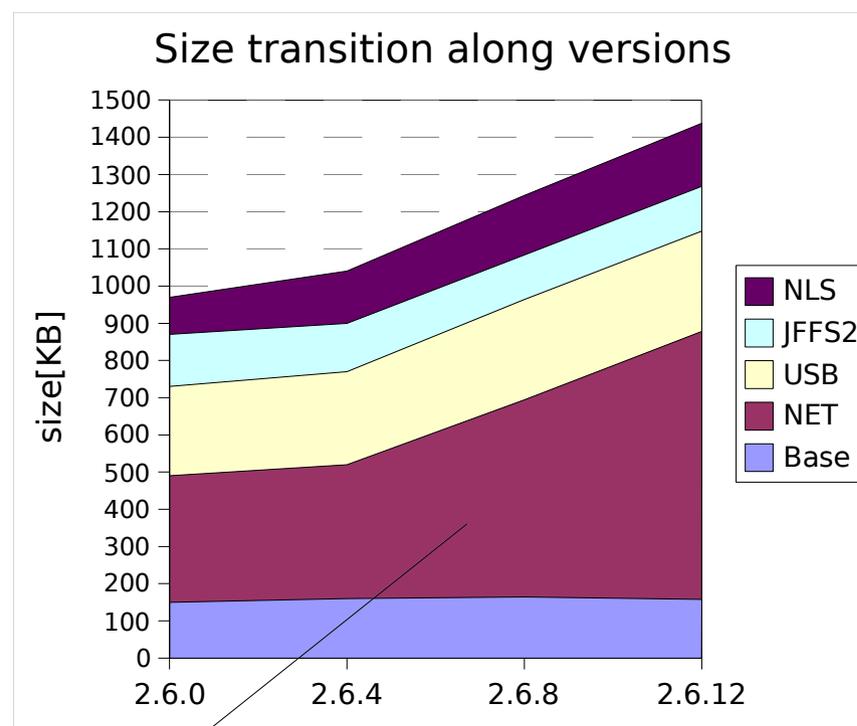


# 1. プロジェクトの目的(補足)

## • 提供する情報のイメージ



この部分のインパクトが大きいことがわかる



この部分が肥大化傾向にあることがわかる

※ 実測値ではありません。あくまでイメージです。



## 2. 実行項目

- サイズ・使用メモリに対するインパクトの大きい部分を明確化する。
- バージョン間で肥大化した部分を明確化する。
- 提供するデータは、組み込み用途で 사용되는様々なアーキテクチャを網羅する。

→ 要改善箇所を特定する。

- **実行のために必要な条件**
  - アプローチの妥当性、測定条件についての十分な検討・議論。
  - 対象アーキテクチャに対する知識。
  - 測定データとカーネルコードの突合せおよび議論。
  - 検証用の機材とオペレーションタイム。

皆さんの協力が必要です！



### 3. 実行ステップ

- 自動検証ツールの作成

- Step 1

- ツール動作確認、測定条件検討ステップ

i386, arm

2.6.12

- Step 2

- 対象バージョンを拡大、バージョン間推移データを取得

current status

i386, arm

2.6.0, 2.6.4, 2.6.8,  
2.6.10, 2.6.12, 2.6.15

- Step 3

- 対象アーキテクチャを拡大、アーキテクチャ間比較データを取得  
... CEL F Test Lab.の活用

i386, arm, mips, ppc, sh, ...

2.6.0, 2.6.4, 2.6.8,  
2.6.10, 2.6.12, 2.6.15, ...

→カーネル改善開発

反復して実行

注

※対象アーキテクチャ

※対象バージョン



### 3. 実行ステップ

**過程と結果は以下のWikiページにて全て公開。**

<http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>

**議論はcelinux-dev ML および SystemSizeWG-MLにて。  
皆様のご意見・ご参加を熱望します！**



## 4. 現在の Status

- **自動検証ツール作成・公開**

- <http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>
- 指定したカーネルコンフィグレーション項目とサイズ・メモリ使用量との関係を自動的に測定するツール(群)。
- 同ページにマニュアルあり。(まずは英語で作成しました)

- **Step1実施・データ公開**

- 検証のアプローチ、測定条件議論のためのパイロット的ステップ。
- データ公開場所はツールと同じページ。

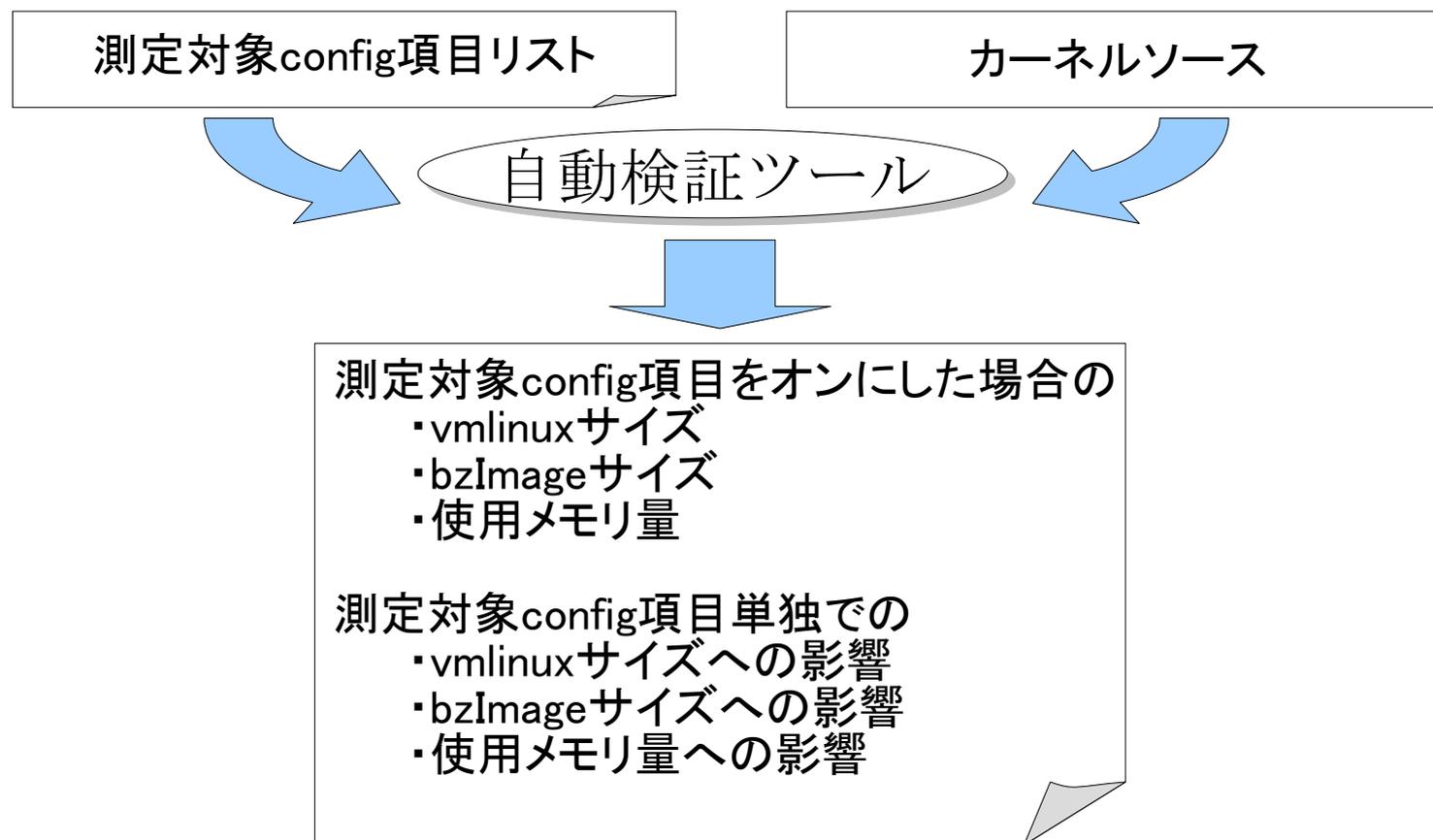
- **Step2実施中**

- バージョン間比較データ取得。
- しかし問題が・・・(後述)



## 5. 自動検証ツール クイックガイド

### ・ 自動検証ツールの概要



ツール:

<http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>  
kconfigs\_size\_rev0.1.2.tar.bz2



## 5. 自動検証ツール クイックガイド

- 測定対象config項目リスト

```
CONFIG_GSC=b
CONFIG_SERIO_GSCPS2=b
CONFIG_SERIO_LIBPS2=b
CONFIG_X86=b
CONFIG_MMU=b
CONFIG_SYSCTL=b
...
CONFIG_EXPERIMENTAL=f
CONFIG_SWAP=f
CONFIG_SYSVIPC=f
...
```

全てのカーネルで  
enableにするconfig項目

一つ一つ  
enableにするconfig項目

### 参考:

<http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>  
i386-up\_conf.tar.bz2 ⇒ i386-up/i386-up.lst

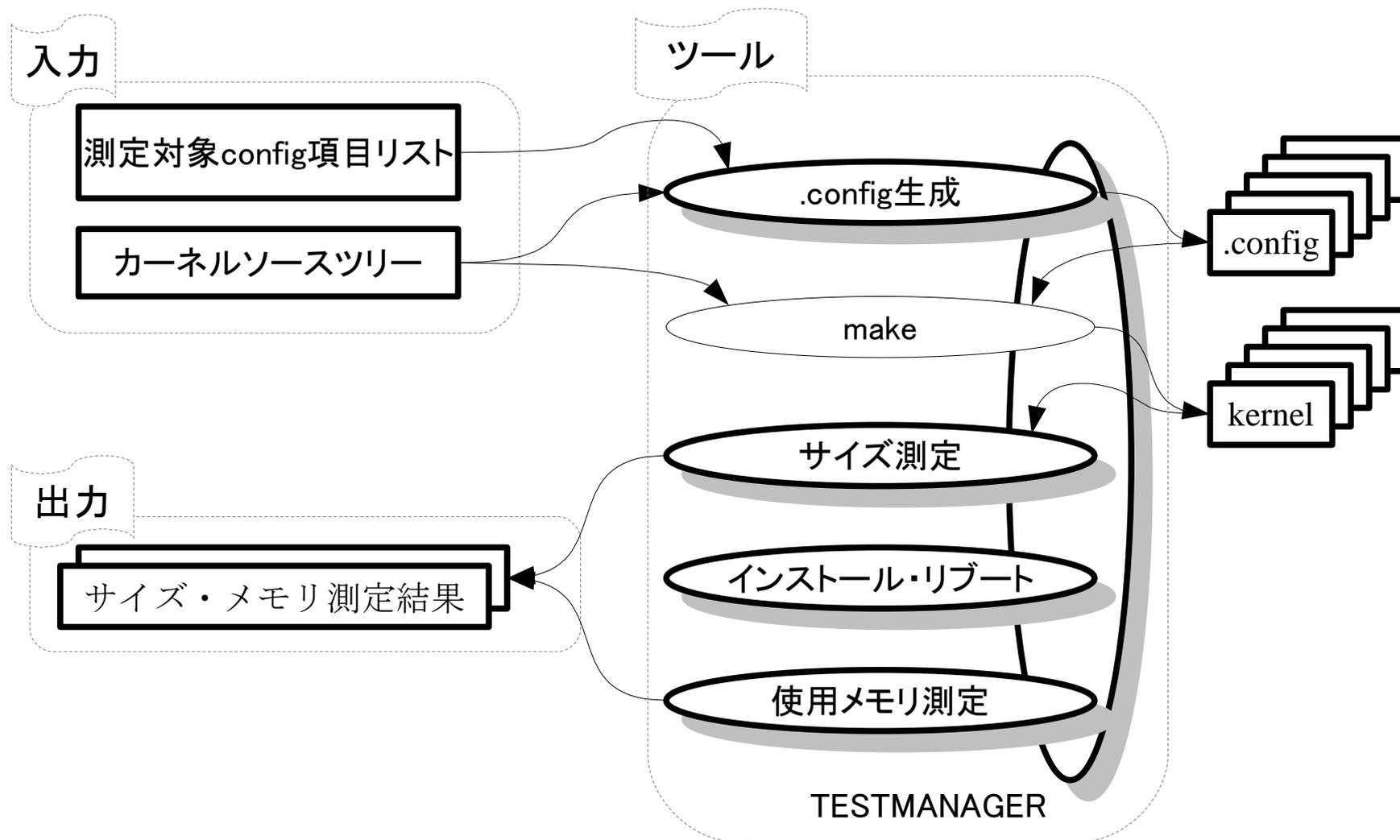


# 5. 自動検証ツール クイックガイド

- コマンド

  - # kconfigsize/scripts/TESTMANAGER <your\_setting\_file>

- 動作概要



## 6. Step1 結果の見方

参考: <http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>  
i386-up\_size.ods ... Kernel2.6.12.3, i386での結果

### • raw\_data シート ... 測定データ

Category abst.	mid.	index	objective item	Kernel image file size [byte]		enabled items
				bzImage	vmlinux	
(base)		0	(base)	592974	1618310	(nothing)
System configuration	General setup	207	EXPERIMENTAL	592974	1618310	EXPERIMENTAL
		206	SWAP	601901	1633546	SWAP
		205	SYSVIPC	605552	1637175	SYSVIPC
...	...	...	...	...	...	...
	DOS/FAT/NT Filesystems	21	VFAT_FS	617705	1671602	VFAT_FS+NLS+FAT_FS

測定対象config項目の  
カテゴリ

通し番号

測定対象config項目

サイズ測定結果

depends/selectによって  
実際にenableにされた  
config項目

VFAT\_FSと(base)との差分が、VFAT\_FSを有効にするために必要なカーネルサイズの増分。

bzImage :  $617705[\text{byte}] - 592974[\text{byte}] \doteq 24[\text{KB}]$

vmlinux :  $1671602[\text{byte}] - 1618310[\text{byte}] \doteq 52[\text{KB}]$

VFAT\_FS=y の場合 NLS, FAT\_FS もselectされて=yとなる。

この増分にはこれらconfigによって有効になるコードのサイズも含んでいる。



## 6. Step1 結果の見方

参考: <http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>  
i386-up\_size.ods ... Kernel2.6.12.3, i386での結果

- impact シート ...  
config項目単独でのサイズ・使用メモリインパクト計算結果  
– 単独でのインパクトが計算できない場合もある(次々ページ参照)

Category		index	objective item	effective items	size impact [byte]		enabled items
abst.	mid.				bzlimage	vmlinux	
(base)			0(base)	(nothing)	0	0	(nothing)
System configuration	General setup	207	EXPERIMENTAL	EXPERIMENTAL	0	0	EXPERIMENTAL
		206	SWAP	SWAP	8927	15236	SWAP
		205	SYSVIPC	SYSVIPC	12578	18865	SYSVIPC
...	...	...	...	...	...	...	...
	Linux/FAT/NT Filesystems	21	VFAT FS	VFAT FS+FAT FS	21252	51942	VFAT FS+NTFS+FAT FS

測定対象config項目の  
カテゴリ

通し番号

測定対象config項目

インパクト計算対象config項目  
(単独計算できない場合は  
複数のconfig項目を含む)

サイズ  
インパクト

depends/selectによって  
実際にenableにされた  
config項目

サイズ・メモリインパクトは、バージョン間・アーキテクチャ間の比較に用いる。





## 6. Step1 結果の見方(補足)

- config項目単独のインパクトが計算できない場合
  - CONFIG\_A は CONFIG\_B と CONFIG\_C に依存している。  
⇒ CONFIG\_A=y の場合の測定結果S1は、CONFIG\_BとCONFIG\_Cの寄与を含んでいる。
  - CONFIG\_B=y、CONFIG\_C=y の場合の測定結果S2がない。

→ CONFIG\_A単独のインパクトは計算できない

測定結果S1      −      測定結果S2      =      ?

CONFIG_A
CONFIG_B
CONFIG_C
(base)

CONFIG_B
CONFIG_C
(base)

CONFIG\_A...?



## 7. Step 2

### • 2.6シリーズ バージョン間推移検証

- 測定対象config項目 : 組み込み用途でメジャーと考えられる約200項目
- 対象カーネルバージョン : 2.6.0, 2.6.4, 2.6.8, 2.6.10, 2.6.12.6, 2.6.15.6
- 対象アーキテクチャ : i386, ARM



## 7. Step 2

- データ分析に問題が発生。
- バージョン間のサイズ・使用メモリ比較は、「インパクト計算対象config項目が同じもの同士を比較」という構想だったが...

config項目の依存性が変更された結果、バージョン間の比較が行えないケースがある。

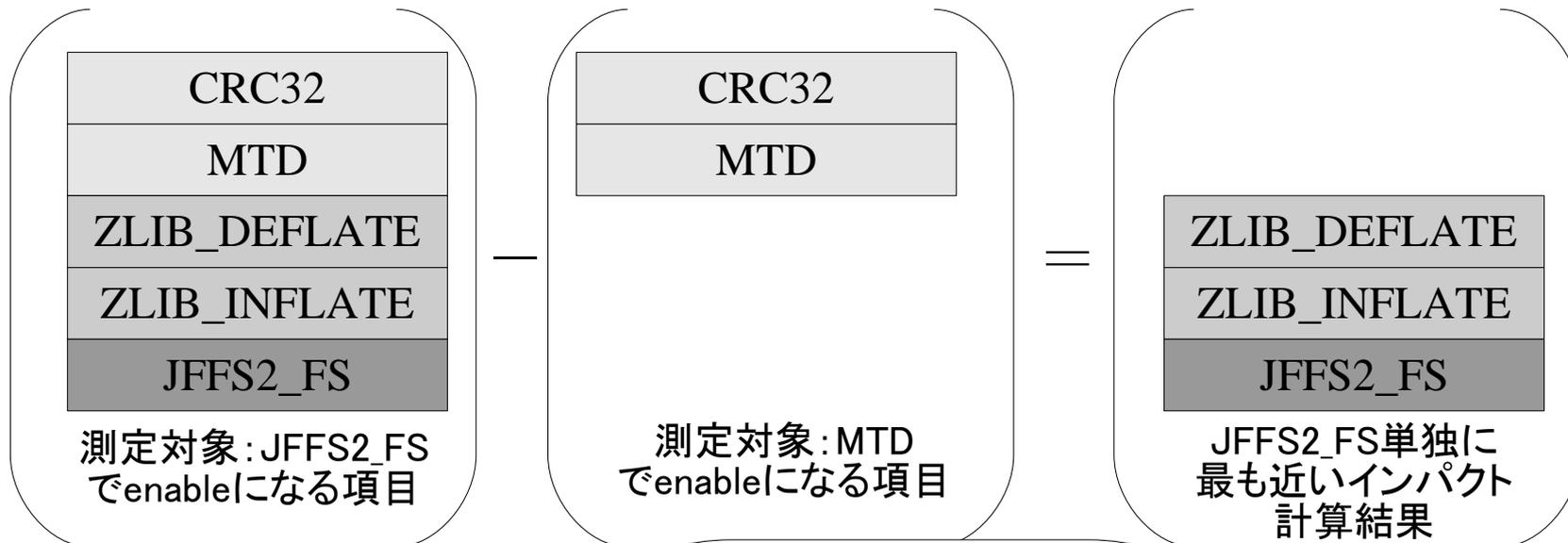
比較バージョン		比較不能項目		総項目数
		割合	項目数	
2.6.0	2.6.4	4%	7	194
2.6.4	2.6.8	13%	26	208
2.6.8	2.6.10	9%	19	213
2.6.10	2.6.12.6	16%	37	231
2.6.12.6	2.6.15.6	6%	14	223
2.6.0	2.6.15.6	35%	87	246



## 7. Step 2

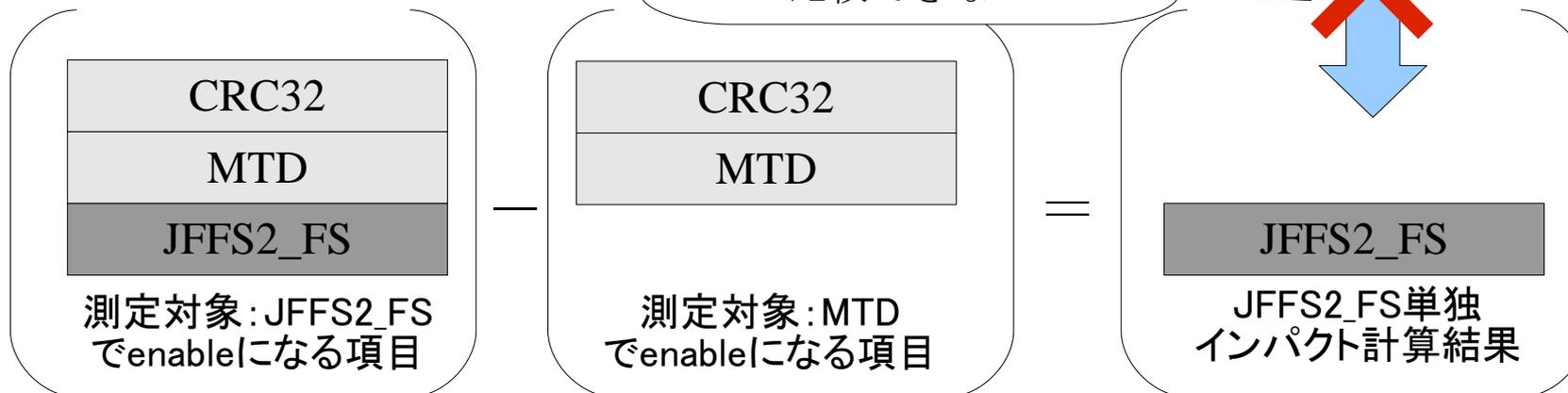
- 例: JFFS2\_FS のサイズをバージョン間で比較したい場合

2.6.0



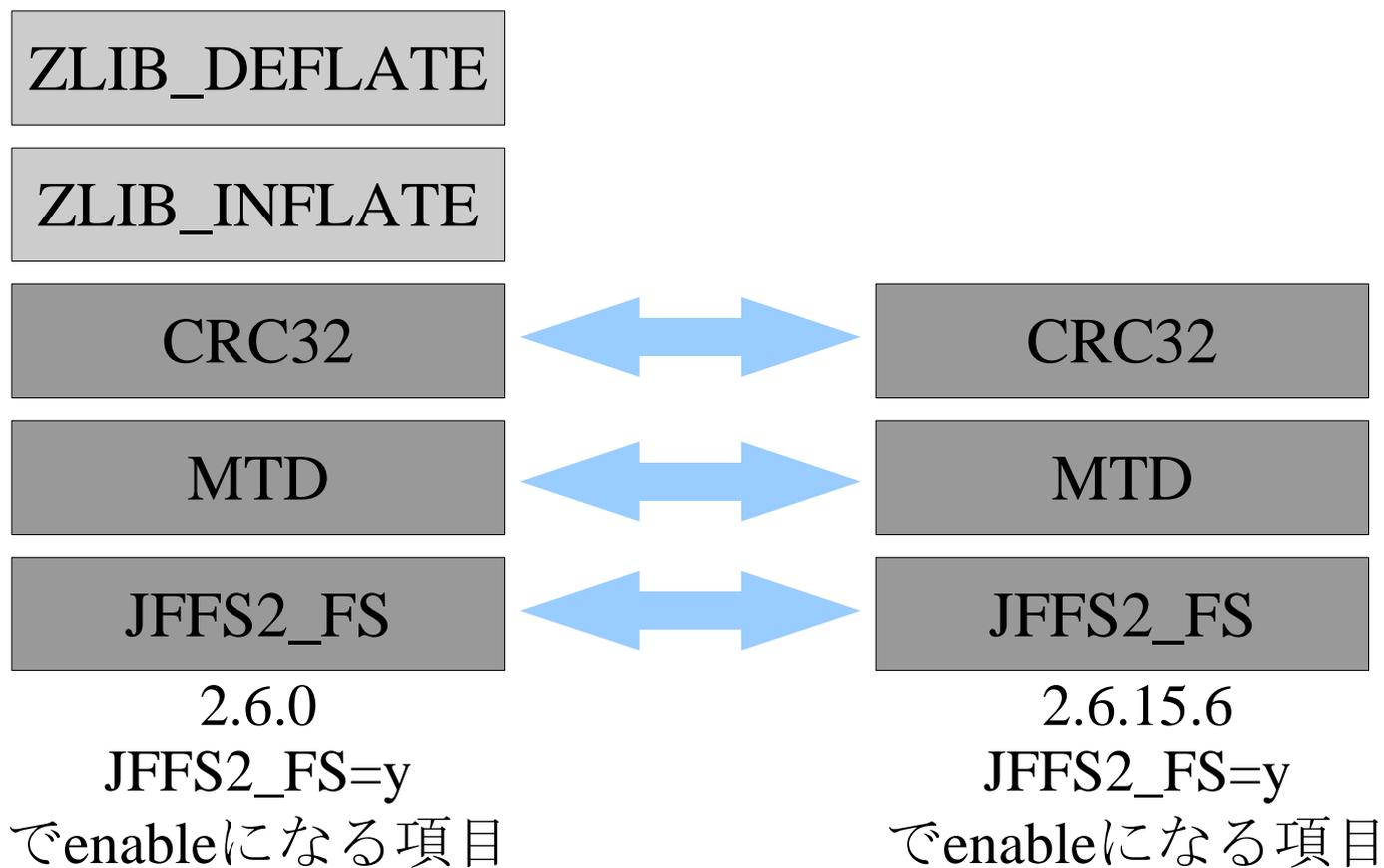
インパクト計算対象  
config項目が異なるため、  
比較できない

2.6.15.6



## 7. Step 2

- 対応案1  
全てのconfig項目単独のサイズを計算して比較する。
  - ツールの改造が必要。

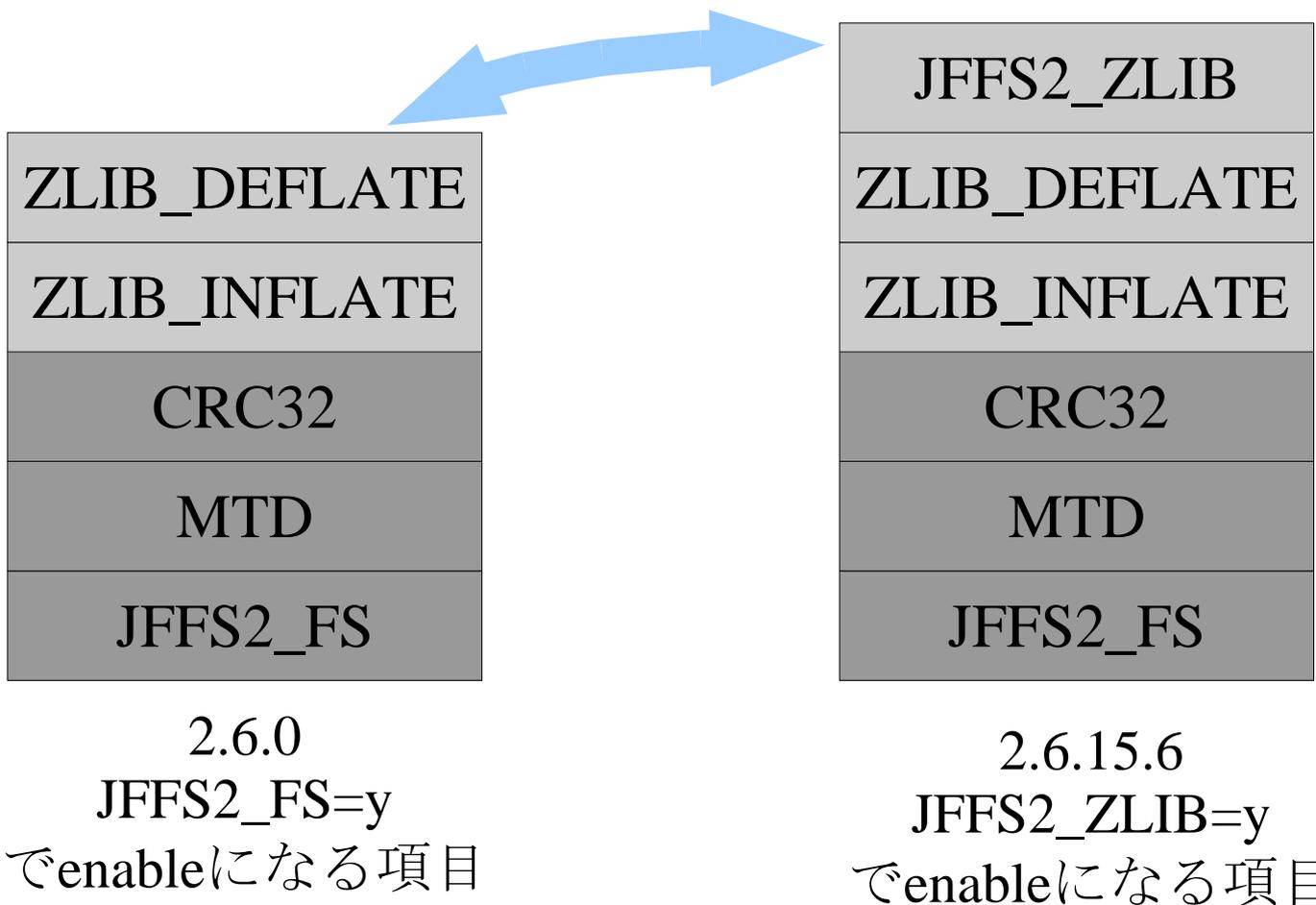


## 7. Step 2

- 対応案2

この2つは含まれる機能が同じ。  
よってこの2つを比較する。

– ただし、config項目の依存関係を人手で確認する必要あり。



## 8. 今後の方針

- **Step2の結果分析**

- まずは比較できる部分のデータについて分析を行う。
- 「対応案2」が適用可能なデータがあるかどうかは人手で確認する。

- **Step3**

- 「対応案1」はStep3への課題。
- 測定対象config項目全ての単独サイズ・使用メモリが測定できるよう、ツールを改造する予定。



## 8. 今後の方針

**ご意見をお持ちの方がいらっしゃいましたら、お知恵を拝借頂けないでしょうか？**

- 「config項目単独」という考え方を捨て、あるconfig項目をenableにした場合のサイズ増分を比較すべき？
- config項目を機能切り出しのI/Fとして用いるのがそもそも間違い？



## 9. まとめとお願い

- 本プロジェクトには、皆さんの協力が不可欠です。ぜひともご参加を！
- 皆さんからのご意見は、なんでも大歓迎です。
  - 「もっとこういうアプローチの方が良いのでは？」
  - 「これでは分かん。こういうデータを見せてくれ」
  
  - 検証のアプローチは正しいか？
  - ベースとなるconfig項目は妥当か？
  - 測定対象config項目は妥当か？
  - etc. ...

### Wiki

<http://tree.celinuxforum.org/CelfPubWiki/KernelConfigWeight>

### mail

celinux-dev ML , SystemSizeWG-ML

直接 m-ikeda((at mark))ds.jp.nec.com でもOKです！



Empowered by Innovation

**NEC**