

luatexja-ruby.sty

2014年3月19日

概要

luatexja-ruby.sty は, LuaTeX-j_a の機能を利用してルビの組版処理を行う追加パッケージである. LuaTeX, LuaTeX-j_a の内部処理に割り込むことにより, 熟語ルビ中の行分割や, 行頭形・行中形・行末形の自動検出, また進入許容量の自動設定などを可能とした.

目次

1	使い方	2
1.1	用語	2
1.2	命令	2
2	注意点	6
3	実装の大まかな方法	7
4	いくつかの例	10
5	「日本語組版処理の要件」20120403 の例	12

1 使い方

パッケージ読み込みは、`\usepackage{luatexja-ruby}` で良く、必要ならば自動的に LuaTeX-ja 本体を読み込む。plain LuaTeX でのロードはまだサポートしておらず、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ のみサポートしている。

1.1 用語

「進入 (intrusion)」「突出 (protrusion)」という用語は、ZR さんによる `pxrubrica` パッケージでの用法に準ずる。

進入あり：と^{あかつき}暁の
進入なし：と^{あかつき}暁の
突出あり：聴^{ちようしゆう}衆
突出なし：聴^{ちようしゆう}衆

1.2 命令

■`\ltjruby` ルビ出力用命令の本体。 `\ruby` という別名を定義している。

`\ltjruby[option]{親|文|字}{おや|も|じ}`

のように使用する。

第 2・第 3 引数内の「|」はグループの区切りを表す。グループの数は両者で一致しなければならないが、`\ltjruby{紋章}{もん|しょう}` のようにはできない。

- 1 グループのみのルビ (単純グループルビ) はグループルビとして組まれる。そのため、もしモノルビを使いたければ、面倒でも

^{もんしょう} の紋章が	の <code>\ltjruby{紋}{もん}\ltjruby{章}{しょう}</code> が
--------------------------	--

のように、複数回使用すること。

- 全てのグループにおいて「ルビ文字列の長さは親文字列以下」であれば、単純グループルビの並びとして扱われる。すなわち、次の 2 行は全くの等価となる。

`\ltjruby{普|通|車}{ふ|つう|しゃ}`

`\ltjruby{普}{ふ}\ltjruby{通}{つう}\ltjruby{車}{しゃ}`

- 複数グループかつ上記の条件を満たさない場合は、所謂「可動グループルビ」であり、グループの切れ目で改行が可能となる。例えば

…の\ltjruby{表|現|力}{ひょう|げん|りよく}は…

という入力からは得られる組版結果は、次のいずれかになる。

改行なし（行中形）	ひょうげんりよく …の表 現 力 は…
直前で改行	ひょうげんりよく 表 現 力 は…
	ひょうげんりよく …の表 現 力 は…
	ひょうげんりよく …の表 現 力 は…
直後に改行	ひょうげんりよく …の表 現 力

上記の例で見られるように、2ブロック以上をまとめて組むときは、全体を1つのグループのように組版する（JIS X 4051と同様）。「日本語組版処理の要件」では、附属書Fに「熟語の構成、さらにその熟語の前後にくる文字の種類を考慮して配置する方法」として別の方法を解説しているが、こちらの方法は現時点ではサポートしていないので、面倒でも手動で `pre`、`post` などを使って頑張ってください。

さて、`<option>` には以下の内容を key-value リストで指定可能である：

`pre=<real>` 前進入許容量をルビ全角単位で指定。負の長さを指定した場合は、ルビの状況や直前の文字に応じた自動指定を意味する。既定値は負（つまり、自動指定）。

`post=<real>` 同様に、後進入許容量を指定する。既定値は負（自動指定）。

`mode` 進入処理のモードを表す bit vector。下位 2 bit は、`pre` や `post` が負である場合にしか効力を発揮しない。既定値は $(0001)_2 = 1$ 。

`bit 0` 進入を無効にするならば 0、有効にするならば 1。

`bit 1` 前進入許容量 B と後進入許容量 A が異なった場合、そのまま処理する場合は 0、小さい方に揃えるならば 1。

`bit 2-3` ルビ文字の突出量 x から実際の前進入量 b 、後進入量 a の計算方法を指定する。親文字の文字数が $k+1$ 、親文字の前に入る空白量・間の空白量・後ろの空白量の比が $p : q : r$ のとき、

$$00 \quad b = \min\{B, xp/[p+kq+r]\}, \quad a = \min\{A, xr/[p+kq+r]\}$$

$$01 \quad b = \min(B, x), \quad a = \min[A, \max(x-b, 0)]$$

$$10 \quad a = \min(A, x), \quad b = \min[B, \max(x-a, 0)]$$

11 $M = \min(B, A)$ とおく。もし $x \leq 2M$ ならば $b = a = x/2$ 。そうでなければ

$$b = \min\left(B, M + \frac{(x-2M)p}{p+kp+r}\right), \quad a = \min\left(A, M + \frac{(x-2M)r}{p+kp+r}\right)$$

組み方の具体例を実際に示す。例示のため、平仮名にはルビが1字まで、「立」にはルビを0.5字分までかけてよいことにしている。

00	は美しい	は聴衆と	は暁立	は聴衆立
01	は美しい	は聴衆と	は暁立	は聴衆立
10	は美しい	は聴衆と	は暁立	は聴衆立
11	は美しい	は聴衆と	は暁立	は聴衆立

`stretchruby={⟨left⟩⟨middle⟩⟨right⟩}` 親文字の合計長がルビ文字の合計長より長い時に、ルビ文字間に入れる空白の割合であり、それぞれ0-7の自然数で指定する。既定値は {1}{2}{1} である。⟨left⟩はルビ文字の先頭までの空き量、⟨middle⟩はルビ文字間の空き量、⟨right⟩はルビ文字の末尾からの空き量（の比）を表す。以下が例である。


`1 \ltjruby[stretchruby=123,maxmargin=2]%`
`2 {○○○○}{◆◆}`

`stretch={⟨left⟩⟨middle⟩⟨right⟩}` 行中形でルビ文字の方が長い場合、親文字の前・中・後に入れる空白の割合。既定値は {1}{2}{1} である。「親文字均等割禁止」を行う場合は、⟨middle⟩を0にした値、例えば {1}{0}{1} を指定することになる。

`stretchhead={⟨left⟩⟨middle⟩⟨right⟩}` 行頭形～。既定値は {0}{1}{1} である。

`stretchend={⟨left⟩⟨middle⟩⟨right⟩}` 行末形～。既定値は {1}{1}{0} である。

`maxmargin=⟨real⟩` 親文字の方がルビより長い時に、ルビの先頭と親文字の先頭、及びルビ末尾と親文字の末尾の間に許される最大の空白量。親文字全角単位で指定し、既定値は0.5。

`size=⟨real⟩` ルビ文字の親文字に対する大きさ。既定値は0.5。

`intergap=⟨real⟩` ルビ文字と親文字との空きを親文字全角単位で指定。既定値は0。

`rubysmash=⟨bool⟩` ルビの高さを0にするか。既定値は偽。次が例である。


`1 \vrule width 0pt height 2\zw depth 1\zw`
`2 \frame{\ltjruby[rubysmash=false]{本}{ほん}}\`
`3 \frame{\ltjruby[rubysmash=true]{本}{ほん}}\`
`4 \frame{\ltjruby[rubysmash=false,intergap=0.2]{本}{ほん}}\`
`5 {本}{ほん}}\`
`6 \frame{\ltjruby[rubysmash=true,intergap=0.2]{本}{ほん}}\`
`7 {本}{ほん}}\`
`8 \frame{\ltjruby[rubysmash=false,intergap=-1.5]{本}{ほん}}\`
`9 {本}{ほん}}`

`kenten=⟨command⟩` 各文字につく圏点の出力命令を指定する。既定値は「`\ltjalchar‘・’`」であり、U+2022 (Bullet) を **ALchar** として出力することを指定している。

`fontcmd=⟨command⟩` ルビ用のフォント切り替え命令を指定する。

次の2つは、以上で説明した複数のオプションを一度に設定するためのものである。普通はこの2つのうちいずれかを設定するだけで足りるだろう。

naka 以下のオプションを同時に設定する。主に中付きルビを組むときに用いる。

```
mode=1, stretch={1}{2}{1}, stretchruby={1}{2}{1}
```

kata 同様に、肩付きルビ用に、次を設定する。

```
mode=9, stretch={1}{2}{1}, stretchruby={0}{0}{1}
```

■`\ltjsetruby{<option>}` `<option>` の既定値を指定する。luatexja-ruby 読み込み時の値は各項目の所で既に説明してあるが、

```
pre = -1, post = -1, mode = 1,  
stretchruby={1}{2}{1}, stretch = {1}{2}{1},  
stretchhead = {0}{1}{1}, stretchend = {1}{1}{0},  
maxmargin=0.5, size = 0.5, intergap=0, rubysmash = true,  
kenten=\ltjalchar‘・’, fontcmd=\relax
```

であり、中付きルビで組まれることになっている。

■`\ltjsetparamater` に追加されるキー

`rubypreintrusion={<chr_code>, <pre_int>}` 文字 `<chr_code>` に、その直後のルビによって掛けられるルビ文字列の最大長をルビ全角単位で指定。

`rubypostintrusion={<chr_code>, <post_int>}` 文字 `<chr_code>` に、その直前のルビによって掛けられるルビ文字列の最大長をルビ全角単位で指定。

既定値は、`rubypreintrusion`, `rubypostintrusion` とも以下の文字に対しては 1, その他の文字については 0 である：

平仮名 (U+3040–U+309F), カギ括弧「」, 読点「,」「、」, 中黒「・」

■`\ltjkenten` 圏点を出力する命令で、`\kenten` という別名を定義している。

`\ltjkenten[<option>]` 親文字

のように使用する。複数文字に圏点をつける場合でも、`\ltjruby` のように「|」を使って文字を区切る必要はないことに注意してほしい。

圏点として出力される文字は `kenten` キーによって指定し、圏点自身の大きさは (ルビと同様に) `size` キーで指定する。

2 注意点

ルビ文字のはみ出し 「日本語組版処理の要件」の図 3.82 のように、ルビ文字のはみ出しが繋がらないようにする処理について述べる：例えば、

りよう みささぎ
 陵◆陵

において、後者の「陵」のルビが前の「◆」にかかる量は次のように決まる：

- 1 回目の実行では、行分割前に「陵」の後側進入量は前もって知ることはできない。なので、「陵」は行中形で組まれるものとして「陵」前側進入許容量は

$$\underbrace{0.5 \text{ zw}}_{\text{元々の許容量}} - \underbrace{0.25 \text{ zw}}_{\text{前のルビの後側進入量 (行中形)}} = 0.25 \text{ zw}$$

となる。なお、行分割後、「陵」が実際に組まれた時に使われた後側進入量は aux ファイルに記述される。

- 2 回目以降の実行では、aux ファイルに保存された「陵」の後側進入量を用いて、「陵」前側進入許容量を

$$\underbrace{0.5 \text{ zw}}_{\text{元々の許容量}} - \underbrace{0.25 \text{ zw}}_{\text{前のルビの後側進入量 (from aux)}} = 0.25 \text{ zw}$$

と計算する。

aux ファイルに保存する際、各 `\ltjruby` 命令の呼び出しを識別するキーが必要になるが、そのキーとしては単純に「何個目の `\ltjruby` 命令か」である。

可動グループルビの制限 実装方法の都合上、ルビの直前・直後・途中で 2 箇所以上の改行が起きる場合（以下のパーツの組み方が出てくる）に対応できない。

組み方	サンプル
単独 1	りゆう 流
単独 2	ちよう 暢
単独 (1 + 2)	りゆう ちよう 流 暢

段落末尾のルビ

- 段落がルビで終わった場合、そのルビが行末形で組まれることはない。これは、段落の「本当の」末尾には `\penalty10000\parfillskip` があるため、ルビ処理用に作った最後のグルー（下の説明では g_2 ）が消去されないことによる。

`\parfillskip` の長さ（や、場合によっては `\rightskip`）を実測し、それによって処理を変えるのも可能だが、そのようなことはしなかった。段落がルビで終わることは普通ない（最低でも句点が続くだろう）と思うからである。

段落先頭のルビ 同様に、段落先頭のルビは行頭形にはならず、また `pre` が負（つまり、自動指定）のときは段落最初のインデント部分への進入は行わない。

0	1	2	3	4	5
		みささぎ			
		陵	は	……	
		みささぎ			
		陵	は	……	

```

1 \parindent1\zw\noindent 0 1 2 3 4 5 \par
2 \ltjruby{陵}{みささぎ}は……\par
3 \ltjruby[pre=1]{陵}{みささぎ}は……
4 % 手動で指定するのは OK

```

3 実装の大まかな方法

次の例で説明する。

^{りゅうちよう} ……を流暢に……

```
1 ……を\ltjruby{流|暢}{りゅう|ちよう}に……
```

- `\ltjruby` コマンド自体は、一旦次の `node list` を値とする `whatsit W` を作って、現在の水平リストへと挿入する（必要ならば `\leavevmode` も実行）：



ここで、最初の w の値 2 は、ルビが 2 つのパーツ「流」「暢」からなっていることを表している。この値を cmp とおこう。 s_i 達の中の文字は既にルビの大きさである。

- LuaTeX-ja の和文処理グルー挿入処理において、この `whatsit W` はまとめて「先頭が『流』、最後が『暢』であるような `hbox` を `\unhbox` で展開したもの」と扱われる。言い換えれば、ルビ部分を無視した単なる「流暢」という和文文字の並びとして扱われる^{*1}。次のサンプルを参照

. A
. A

```

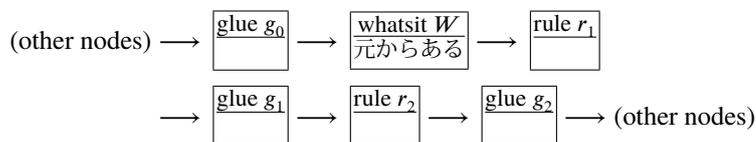
1 \leavevmode\hbox{.}A\
2 % ↑ xkanjiskip
3 \ltjruby{.}{A}
4 % ↑ 2分

```

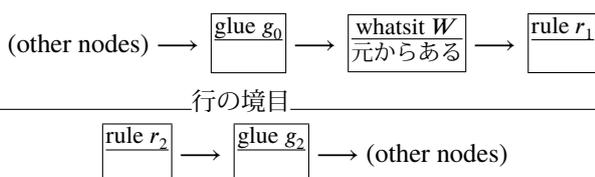
- 和文処理グルーの挿入が終わった後で、可動グループルビのためのノードの挿入に入る。

^{*1} 「流」「暢」の間のグルーは既に入っている、と扱われる。

- (a) W の前後に $2cmp + 1 = 5$ 個のノードが挿入され、 W の周辺は次のようなノード列になる。



- (b) このようにノードを挿入する目的は、 TeX の行分割処理自体に影響を加えずに可動グループルビを実現させることにある。



のようになったとしたら、「流」^{りゅう}「暢」^{ちよう}の間で行分割が起きた、ということがわかり、 g_i, r_i 達のノードを適切に置き換えればよい（後で詳しく説明する）。

- (c) なお、 r_i 達の高さ・深さは組み上がった後のそれである。 g_i, r_i 達の幅は、以下の対応に沿って算出する。

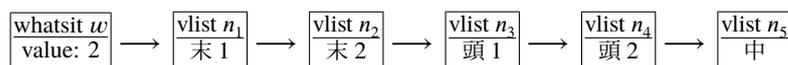
node 名	組み方	サンプル	対応するノード並び
n_1	行末 1 グループ	を流 ^{りゅう}	$g_0 \rightarrow W \rightarrow r_1$
n_2	行末 2 グループ	を流暢 ^{りゅうちよう}	$g_0 \rightarrow W \rightarrow r_1 \rightarrow g_2 \rightarrow r_2$
n_3	行頭 1 グループ	暢に ^{ちよう}	$r_2 \rightarrow g_2$
n_4	行頭 2 グループ	流暢に ^{りゅうちよう}	$W \rightarrow r_1 \rightarrow g_2 \rightarrow r_2 \rightarrow g_2$
n_5	行中	を流暢に ^{りゅうちよう}	$g_0 \rightarrow W \rightarrow r_1 \rightarrow g_2 \rightarrow r_2 \rightarrow g_2$

例えばこの場合、 n_5 に対して

$$g_0 + r_1 + g_2 + r_2 + g_2 = 3zw - (0.25zw \times 2) = 2.5zw$$

という方程式が立つ (zw は親文字全角の幅，進入量込)。 n_1 から n_5 まで計 5 本の方程式が立つが、これらは Gauß の消去法で解くことができ、 g_i, r_i 達の幅が求まる。

- (d) また、ルビ処理を統括している whatsit W の値も

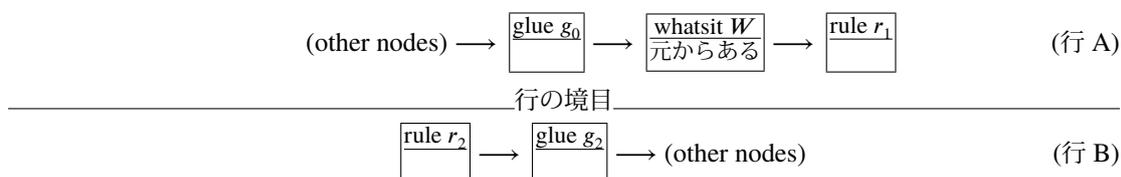


に置き換えておく。

4. Lua TeX の行分割処理を普通に行う。

5. 行分割の結果に従って, g_i, r_i 達を適切に置換する.

例えば行分割の結果



のようになったとしよう.

(a) 処理は段落の上の行から順番に行われる. 行 A の処理がまわってきたとしよう.

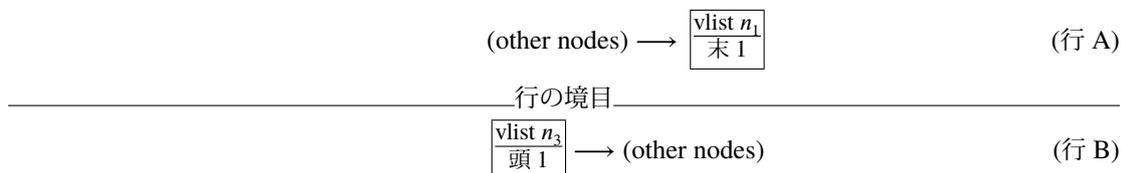
(b) 行 A の先頭から順番に眺めていく. すると「whatsit W 由来」のノード, g_0, W, r_1 が見
つかり, 行 A はここで終わっている.

まず, 行 A の hbox の中身から whatsit W を消去 (リストから取り除くだけで, W のメモ
リを解放するわけではない) する. $g_0, (W), r_1$ というノードの並びは, 「行末 1 グル
ープ」 n_1 に対応しているので, g_0, r_1 を行 A から除去・メモリ解放し, 代わりに n_1 を行 A
の中身に追加する.

(c) 次に行 B の処理にうつる. 行 A でルビの処理は完了していない (2 パーツのルビなのに
まだ 1 パーツ目しか使っていないからである) ので, 「whatsit W 由来」のノードがいく
つか残っているはずである.

案の定, r_2, g_2 というノード列が見つかった. これは「行頭 1 グループ」 n_3 に対応してい
るので, r_2, g_2 を行 B から除去・メモリ解放し, 代わりに n_3 を行 B の中身に挿入する.

(d) これで 2 パーツとも使い切ったことになるので, 隔離しておいた W を, (使われなかった
 n_2, n_4, n_5 などと共に) メモリ解放する. 結果として次のようになった:



5 「日本語組版処理の要件」20120403 の例

■3.3.1 節

- 3.49 君子^{くんし}は和^わして同^{どう}ぜず
- 3.50 人^{ひと}に誨^{おし}えて倦^うまず
- 3.51 鬼門^{きもん}の方角^{ほうかく}を凝視^{ぎようし}する
- 3.52 鬼門^{きもん}の方角^{ほうかく}を凝視^{ぎようし}する
- 3.53 茅場町^{かやばちよう} 茅場町^{かやばちよう}
- 3.54 紫陽花^{あじさい} 埧塙^{るつぼ} 田舎^{いなか}
- 3.55 模型^{モデル} 顧客^{クライアント} 境界面^{インターフェース} 避難所^{アジール}
- 3.56 編集者^{editor} editor

■3.3.3 節

- 3.58 に幟^{のぼり}を に幟^{のぼり}を 韋編三絶^{いへんさんぜつ}
- 3.59 に幟^{のぼり}を
- 3.60 韋編三絶^{いへんさんぜつ} 韋編三絶^{いへんさんぜつ}

■3.3.4 節

- 3.61 図（両側ルビ）はまだ未サポートにより省略

■3.3.5 節 モノルビ

- 3.62 の約^{やく}に
- 3.63 版面^ちの地^ちに 版面^ちの地^ちに
- 3.64 × 版面^ちの地^ちに（まだ縦組み未サポート）
- 3.65 の砦^{とりで}に の砦^{とりで}に
- 3.66 上^{しゆん}の句^{しゆん}に 後句^{しゆん}に
- 3.66 下^{しゆん}の句^{しゆん}又 後句^{しゆん}又

■3.3.6 節 グループルビ

- 3.67 は冊子^{コーデックス}体と
- 3.68 模型^{モデル} 利用許諾^{ライセンス}
- 3.69 模型^{モデル} 利用許諾^{ライセンス}
- 3.70 なげき^じの聖母^エ像^タ ←自動調整
- 3.71 顧客^{クライアント} 境界面^{インターフェース}
- 3.72 顧客^{クライアント} 境界面^{インターフェース}

■3.3.7 節 熟語ルビ

- 3.73 左^{きゆう}杞憂^{いふ} 畏怖^{いふ}
- 3.73 右^{きゆう}杞憂^{いふ} 畏怖^{いふ}

3.74 ^{りゆうぎ}の流儀を ^{むじょう}の無常を ^{じょうじゆ}の成就を

3.74 ^{もんしやう}の紋章を ^{しやうちやう}の象徴を

3.75 ^{りゆうぎ}の流儀を ^{むじょう}の無常を ^{じょうじゆ}の成就を

3.75 ^{もんしやう}の紋章を ^{しやうちやう}の象徴を

3.76 × ^{りゆうぎ}の流儀を ^{むじょう}の無常を
ああああああ

要調整 3.77 ^{りゆうぎ}流儀がある。

←ルビ中で改行するには？

……の無
^{じょうじゆ}常を

■3.3.8 節 ルビはみ出し

3.78 ^{ひと}人は死して名を^な残す

要調整 3.79 漢字の部首には^{へん}偏・^{かんむり}冠・^{きやく}脚・^{つくり}旁がある

要調整 3.79 漢字の部首には^{へん}偏、^{かんむり}冠、^{きやく}脚、^{つくり}旁がある

3.79 ^{うわさ}この噂の好きな人は^{ふところ}懐ぐあいもよく、^{ひのき}檜を

3.80 漢字の部首には「^{へん}偏」「^{かんむり}冠」「^{きやく}脚」「^{つくり}旁」がある

3.80 ^{うわさ}この噂好きな人は^{ふところ}懐 具合もよく、^{ひのき}檜材を

3.81 × ^{あかつき}に暁の^{おもむき}趣を

3.82 ^{あかつき}に暁の^{おもむき}趣を

3.83 ^{うわさ}この噂の好きな人は^{ふところ}懐ぐあいもよく、^{ひのき}檜を

3.83 ^{うわさ}この噂好きな人は^{ふところ}懐 具合もよく、^{ひのき}檜材を

3.84 ^{うわさ}この噂の好きな人は^{ふところ}懐 ぐあいもよく、^{ひのき}檜を

3.84 ^{うわさ}この噂好きな人は^{ふところ}懐 具合もよく、^{ひのき}檜材を

あああああああああ

^{こみち}徑をあああああああああ

あああああああああ

要調整 3.85 ^{こみち}徑ああ

なぜ行末形にならない

あああああああああ

をあああああああああ

3.86 ^{こみち}徑を

あああああああああ

^{アクセサリ}飾り 等あああああああ

あああああああああ ^{アクセサリ}共 飾り

3.87 あ。

■**圈点の例**（常用漢字表前書きより） この表は、法令、公用文書、新聞、雑誌、放送など、一般の社会生活において現代の国語を書き表す場合の漢字使用の目安を示すものである。

以下は要調整

■F.1-2 節

F.01 治癒ちゆ 模索もさく 遷移せんい 混沌こんどん

F.01 中治癒ちゆ 模索もさく 遷移せんい 混沌こんどん

F.02 橋頭堡きやうとうぼ

F.03 凝視ぎやうし 調理師ちやうりし 思春期ししゆんき 管状花かんじやうか 蒸気船じやうきせん

F.03 凝視ぎやうし 調理師ちやうりし 思春期ししゆんき 管状花かんじやうか 蒸気船じやうきせん

F.04 未熟みじゆく 法華経ほけきやう 顕微鏡けんびきやう 課徴金かちやうきん 古戦場こせんじやう

F.04 未熟みじゆく 法華経ほけきやう 顕微鏡けんびきやう 課徴金かちやうきん 古戦場こせんじやう

F.05 の峻別しゆんべつは

F.06 以降は未チェック