

ルビの簡便な配置ルール (案)

小林 敏

0 この文書の目的

この文書では、CSS, SVG 及び XSL-FO などの技術で実現が求められる日本語組版のルビ処理について、実装する際の参考となる簡便な処理方法を示した。簡便なルビ処理方法を実現するために、ここでは、JLReq とは異なり、何を選べばよいか、何が重要であるかを考慮し、配置方法を一つに絞ることにした（どのような事項を考慮したかは次項で解説する）。あわせて、JLReq で解説していない両側ルビの配置方法を追記した。

JLReq では、過去に行われていた処理例を紹介する意味もあり、一つのこと複数の処理方法や、かなり複雑な処理方法を示している。なかでも、ルビ処理では、さまざまなケースが出現し、また、ある要求事項を組版で実現しようとするすると矛盾が出てしまう例もある（注の“複雑な処理となる例”参照）。こうした事項まで考慮して自動処理を行うためには、かなり複雑な方法となる（例：JLRq の付属書“F 熟語ルビの配置方法”，この方法でもいくつかの問題は残る）。そこで、ある程度の品質を確保しながら、機械的な処理に対応した単純化した処理方法を検討する意味はある。

以下に示した方法は、理想ではないが、誤読されないということを考えて、例外のあまりでない、また、機械的に処理できる簡便な配置処理方法の一案である。

なお、ここでの用語は、JLReq（日本語組版処理の要件, Requirements for Japanese TextLayout）による。

1 簡便な配置ルールで考慮した事項

ルビ処理のむつかしさ

最初にルビ処理のむつかしさについてまとめておく。

ルビの組版処理は、次のような事項を考慮して、その配置位置を決める必要がある。

- (1) 親文字とルビの対応をどう処理すればよいのか
- (2) 親文字列の全長に比べ、ルビ文字列の全長が短い場合、どう処理すればよいのか
- (3) 親文字列の全長に比べ、ルビ文字列の全長が長い場合、どう処理すればよいのか、親文字列からルビ文字列がはみ出してよいの

注釈としてのルビ（行間注） 行間にルビと同じような配置位置に注釈を配置する方法がある（行間注）。この配置処理は、ここでは適用範囲としない。ルビは、字または単語の読み、別語での言い直しなどであり、親文字は字又は単語（複合語を含む）、ルビも単語（複合語を含む）となる。親文字とルビとの対応は強く、ルビ文字列および親文字列は一体として扱い、熟語ルビを除いて2行にわたる分割は禁止されている。また、親文字列からはみ出したルビ文字列が親文字列の前後の文字に掛からないのが望ましい（ある条件で、限られた範囲では認める場合がある）。これに対し、行間注は、注記であり、注記の対象、注記の内容、注記に使用する文字種も様々であり、注記の対象（ルビという親文字）の指示方法も異なる。注記の対象からの注記の文字列のはみ出しは、ある程度関係はとるとしても、問題とならず、分量が多くなれば、2行にわたる分割も認められる。このように内容・形式・処理方法に多くの違いがあるので、ここでは扱わない。

複雑な処理となる例 できるだけ字間を空けないという条件を満たすために、親文字からはみ出したルビを漢字には掛けないが、仮名には掛ける、とする処理方針がある。しかし、前後が同じ仮名又は漢字となる場合はよいとしても、親文字の前が仮名で、後ろが漢字といったときに、ルビ文字の字数によっては見た目のバランスを壊す場合も出てくる（モノルビの例を図1に示す、上側が漢字には掛けないが、仮名に掛けるとした方針で処理した例）。グループルビでも、同様である（図2参照）。こうした問題を解決するためには、ルールをさらに追加しないといけない（活字組版では個別箇所のケースに応じて配置位置を工夫していたので、その箇所ごとに適当に処理できた）。

また、ルビ文字が片仮名の場合、単語としてのまとまりがあり、前後の仮名にも掛けないという処理方法もあった（実際にもルビの字数によっては、仮名にも掛けないとした方が見た目のバランスはよい（図3参照、下側が仮名に掛けないとした方針で処理した例）。これを実現するためにも、さらなるルールの追加が必要となる。

か

(4) 親文字列からルビ文字列がはみ出した場合、前後に配置する文字や約物に掛かってよいのか、また、このことは、親文字とルビとの配置位置に影響するのか

(5) 親文字列からルビ文字列がはみ出した場合において、行の先頭又は末尾に配置するときは、親文字列の先頭又は末尾を行の先頭又は末尾にそろえるのか、それともルビ文字列の先頭又は末尾を行の先頭又は末尾にそろえるのか

(6) 親文字が複数の場合、2行にわたる分割をしてよいのか

活字組版では、こうした場合、原則的な考え方に従って処理し、問題があれば校正段階で修正の赤字が入り、それに従って直していた。いってみれば、個別箇所でも臨機応変の工夫をして配置位置を決めていた。コンピュータ組版では、ある程度は、配置方針を決め、一定のルールに従って配置処理をしていたが、個別箇所でも、親文字とルビ文字の対応を変える、あるいは配置処理方針を変えるなどの工夫も必要とした。

Web での配置処理を考えた場合、活字組版のように個別箇所でも配置位置を工夫する処理は避けるのが望ましい。となると、上記の問題をすべて解決できる配置処理方針を決め、処理系に実装していく必要がある。

配置ルールで考慮した事項

簡便な配置ルールを考えるにあたっては、次のような事項を考慮し、また前提とした。

- (1) ルビは、親文字の読み方又は意味を示すものである。そこで、誤読されないことを第一とした。具体的には、親文字列からはみ出したルビ文字は、前後に配置する漢字だけでなく、仮名にも掛けない方式とした。
- (2) 縦組と横組とで配置処理法を変えることなく、共通の処理ができる方法とした。具体的には、モノルビでは、親文字列とルビ文字列の中心をそろえる方式のみとした。
- (3) 2レベルの処理方式とした。第1レベルでは、親文字及びルビ文字（両者を合わせて、以下、親文字群という）のみの情報から、親文字とルビ文字の配置関係を決める。第2レベルでは、親文字群の前後に配置する文字を考慮して、行の中での親文字群の配置位置を確定する。逆に言えば、第1レベルで決めた親文字とルビ文字の配置関係は、親文字群の前後に配置する文字により修正することはしない。また、親文字群が行頭又は行末に配置されることにより、親文字とルビ文字の配置関係を直し、親文字の先頭又は末尾を行頭又は行末にそろえる配置方法は採用しない。つまり、第1レベルで決定した事項は、第2レベルでは修正しない方式とした。
- (4) ルビの配置方法として、JLReq や JIS X 4051（日本語文書の組版方法）では、複数の方法が示されているケースがあるが、ここでは上記の方針で1つの方法に限定した。また、ここで示した処

こうした複雑さは、“できるだけ字間を空けないという条件”は満たさないが、前後の仮名にも、はみ出したルビは掛けないという単純なルールにすれば解決できる。

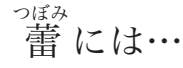
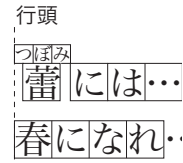


図1 親文字からはみ出しのあるモノルビの例 (後ろが漢字)

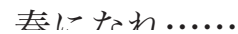
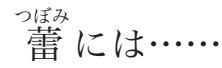
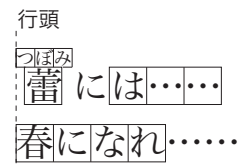


図2 親文字からはみ出しのあるグループルビの例 (後ろが漢字)

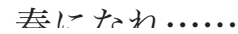
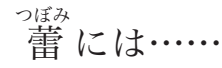
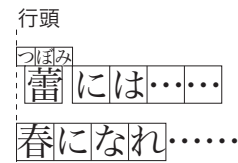


図3 ルビ文字が片仮名のグループルビの例 (前後が仮名)

はみ出したルビと前後の文字の関係 JIS X 4051 では、親文字群（親文字及びそれに付随するルビ文字）と、その前後に配置する文字との関係について、ルビ文字を最大でルビ文字の文字サイズまで、親文字群の前後に配置する仮名にかけてよいという規定とともに、“処理系定義として、ルビ文字を前又は後ろの文字にかけずに配置してもよい”との規定も書かれている。

2レベルの処理方式 熟語ルビでは、2行にわたる分割を認めている。分割された場合は、分割前の親文字とルビの配置位置は変化する。しかし、例えば3字で構成された熟語ルビでは、親文字は3字、1字+2字、2字+1字の3つの組合せがあるが、それぞれの場合において、親文字とルビの配置位置は、それだけで決定され、その配置位置は変化しない。

理法は、原則として JIS X 4051 で規定している方法によった。ただし、処理系定義として採用できる処理方法（オプション）を採用したことがある。例えば、はみ出したルビを仮名にも掛けないという方法は、処理系定義として採用できる方式である。

- (5) ルビ文字の文字サイズを大きく（逆に小さく）したいという要求がある。そこで、ルビの文字サイズは、親文字の文字サイズの 1/2 を初期値（デフォルト値）として採用し、図版では、ルビの文字サイズは親文字の文字サイズの 1/2 で示したが、空きなどのサイズはルビ文字の文字サイズを基準とするのではなく、親文字の文字サイズを基準として規定することにより、ルビの文字サイズが親文字の文字サイズの 1/2 以外であっても採用できる配置方法とした。

ルビの種類

ルビの種類は、親文字とルビとの対応関係から、次の 3 つとする (JLReq, “3.3.1 ルビの使用” 参照)。

- (1) モノルビ
- (2) 熟語ルビ
- (3) グループルビ

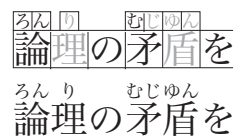
どのルビとするかは、親文字とルビとの対応関係による。モノルビは 1 文字の親文字とルビ文字が対応し、熟語ルビは、複数の親文字のそれぞれの 1 文字とルビ文字が対応し、かつ複数の親文字とルビとを一体として扱い、グループルビでは、複数の親文字の全体に対しルビが対応する (図 4 参照)。

モノルビは親文字が 1 字であるので、その 1 字の親文字だけにルビを対応させている場合は、モノルビであり、明確である。しかし、親文字が複数の熟語ルビとグループルビでは、その扱いが明確でない。複数の親文字全体に対してだけ、ルビを対応している場合は、グループルビとして扱い、複数の親文字全体を指示し、そのうえで親文字を構成する 1 字 1 字の親文字にルビが対応している場合は熟語ルビとして扱う。したがって、対象の漢字が熟語であっても、複数の漢字の個々の漢字だけにルビを対応させ、熟語全体を指示していない場合は、熟語であってもモノルビとして扱う。また、複数の親文字に対するルビの対応はあるが、個々の親文字への対応がない場合は、グループルビとして扱う。

モノルビの例



熟語ルビの例



グループルビの例

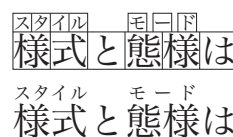


図 4 ルビの種類

2 ルビの簡便な配置ルール

ルビの文字サイズと配置位置

ルビの文字サイズと親文字に対する行送り方向の配置位置は、次による。

- (1) ルビの文字サイズは、親文字の文字サイズの 1/2 を初期値（デフォルト値）とする。
- (2) 縦組のルビは、親文字の右側とし、親文字の文字の外枠とルビ文字の外枠を接して配置する (図 5 参照)。

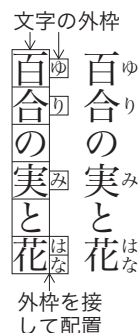


図 5 縦組のルビの例

(3) 横組のルビは、親文字の上側とし、親文字の文字の外枠とルビ文字の外枠を接して配置する（図6参照）。

以下では、モノルビ、熟語ルビ、グループルビの配置処理を解説するが、熟語ルビは、処理がやや複雑なので、モノルビ、グループルビ、熟語ルビの順序で解説する。

モノルビの配置処理

親文字1字にルビを対応させるモノルビの配置処理は、次による。

なお、前述の“配置ルールで考慮した事項”の(3)の2レベルの処理方式でいえば、以下の説明の(1)、(2)及び(3)項は、第1レベルの親文字及びルビ文字の配置処理であり、(4)及び(5)項は、第2レベルの親文字群（親文字及びそれに付随するルビ文字）を行中に配置する処理である。

- (1) ルビの字数が2字以上の場合は、ルビ文字列の字間はベタ組とする。なお、ルビ文字が連数字中の文字（cl-24）、単位記号中の文字（cl-25）、欧文用間隔（cl-26）、欧文用文字（cl-27）のように固有の字幅を持つ文字の場合には、それぞれの文字の固有の字幅に応じて配置する（図7参照）。
- (2) ルビ文字列と親文字の字詰め方向の中心をそろえて配置する（図8参照）。
- (3) モノルビの場合、親文字とそれに付くルビ文字の文字列は、一体として扱い、2行に分割してはならない。
- (4) 親文字よりルビ文字列の全長が長い場合、親文字からはみ出したルビ文字を親文字の前又は後ろに配置する文字に掛けてはならない（図8参照）。

ただし、次に掲げる句点類（cl-06）など、その後ろ又は前に空きがある約物などにはルビを掛ける（図9参照）。（ここで、句点類（cl-06）などの処理に差をつけたのは、句点類（cl-06）などは、文章の区切りとして重要な役割を果たしており、これらの後ろや前後の空きをできるだけ一定していることが望ましく、特にこれらの空気が大きくなると、区切りの意味を変える恐れがあること、また、冒頭の注の“複雑な処理となる例”で述べたような問題もでないことによる。）

- 親文字の前に配置する終わり括弧類（cl-02）、句点類（cl-06）、読点類（cl-07）、和字間隔（cl-14）又は中点類（cl-05）の後ろのアキ（このアキは、中点類（cl-05）以外、通常は親文字の二分（中点類（cl-05）は四分）、ただし、行の調整処理で二分アキや四分アキが詰められている場合は、調整で詰められた空き量までとする、例えば、四分アキとなっていれば、四分まで）
- 親文字の後ろに配置する始め括弧類（cl-01）、和字間隔（cl-14）又は中点類（cl-05）の前のアキ（始め括弧類（cl-01）は通常は親文字の二分、中点類（cl-05）は四分）、ただし、行の調整処理で二分アキや四分アキが詰められている場合は、調整で詰められた空き量までとする、例えば、四分アキとなっていれば、四分まで）

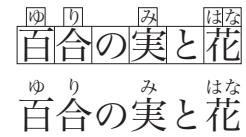


図6 横組のルビの例

行送り方向のルビの配置方法 行送り方向のルビの配置方法は、親文字の文字の外枠とルビ文字の文字の外枠を接して配置するのが原則であるが、視覚障害者にとって、親文字とルビの行送り方向の間隔がある程度は空いているのが望ましい場合もあり、この間隔は変更できることが望ましい。

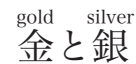


図7 欧字のモノルビの例

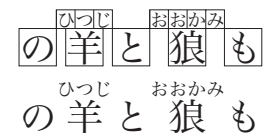


図8 はみ出しのあるモノルビの例1

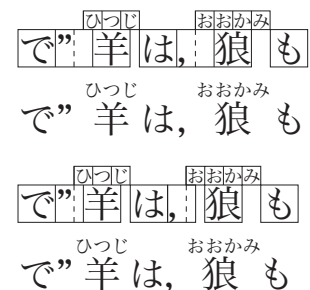


図9 はみ出しのあるモノルビの例2
(下側が約物の後ろが四分詰められている例)

(5) 親文字列よりルビ文字列の全長が長い場合、行頭ではルビ文字列の先頭を行頭にそろえ(図 10 参照)、行末ではルビ文字列の末尾を行末にそろえる(図 11 参照)。行頭又は行末において、親文字の前後にアキが生じていても、ルビ文字が行頭又は行末に接していれば、それでよいとする処理方法であり、JIS X 4051 でも採用され、実際にも行われている方法である。

グループルビの配置処理

ここでは連数字中の文字 (cl-24)、単位記号中の文字 (cl-25)、欧文用文字 (cl-27) のように固有の字幅を持った文字(以下、この項では“欧文用文字等”という)と、平仮名 (cl-15)、片仮名 (cl-16)、漢字等 (cl-19) など(以下のこの項では“仮名等”という)の 2 つに分けて、その組合せで配置方法を説明する。欧文用文字等は、複数の文字をまとめて読むので、字間を空ける処理は、できるだけ避けたいという事情があるからである。いずれの場合も最初に、それぞれをベタ組にした場合のルビ文字列の全長と、親文字列の全長を比較し、配置方法を定める。この場合のルビ文字列の全長と親文字列の全長が同じときは、それぞれをベタ組とし、ルビ文字列及び親文字列の字詰め方向の中心をそろえて配置する(図 12 参照)。それ以外のグループルビの配置処理は、次による。

なお、前述の 2 レベルの処理方式でいえば、以下の説明の (1)、(2)、(3) 及び (4) 項は、第 1 レベル、(5) 項は、第 2 レベルの配置処理である。

(1) ルビ文字及び親文字が“仮名等”の場合は、次による。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より短いときは、ルビ文字列の字間及びその前後を空け、それぞれの文字列の全長を同じにし、各文字列の字詰め方向の中心をそろえて配置する。空ける量は、ルビ文字列の字間の空き量の大きさ 2 に対し、親文字列の先頭からルビ文字列の先頭までの空き量及び親文字列の末尾からルビ文字列の末尾までの空き量を 1 の比率で空ける(図 13 参照)。ただし、ルビ文字列の先頭及びルビ文字列の末尾の最大の空き量は、親文字の文字サイズの 1/2 とし、ルビ文字列の字間の空き量を均等に増やす(図 14 参照)。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より長いときは、親文字列の字間及びその前後を空け、それぞれの文字列の全長を同じにし、各文字列の字詰め方向の中心をそろえて配置する。空ける量は、親文字列の字間の空き量の大きさ 2 に対し、ルビ文字列の先頭から親文字列の先頭までの空き量及びルビ文字列の末尾から親文字列の末尾までの空き量を 1 の比率で空ける(図 15 参照)。

(2) ルビ文字が仮名等、かつ、親文字が欧文用文字等の場合は、次による(図 16 参照)。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より短いときは、ルビ文字列の字間及びその前後を空け、

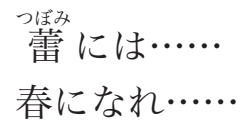
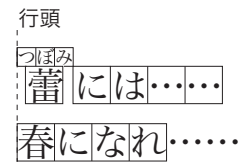


図 10 行頭のモノルビの例

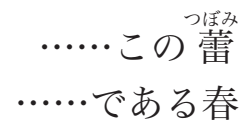
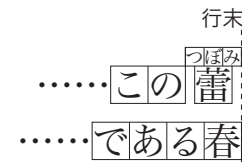


図 11 行末のモノルビの例

グループルビの字間を空ける処理 グループルビの場合、ルビ文字の全体と親文字の全体とが対応していることが望ましい。文字列の長さが異なる場合、親文字又はルビ文字の字間を空けて、それぞれの全長をそろえる、それぞれをベタ組にして文字列の中心をそろえるなど、いくつかの処理方法がある。親文字もルビ文字も漢字や仮名の場合において、ここで採用した方法、活字組版の時代でも、見た目のバランスが良いといわれていた、前後のアキ 1 に対し、字間を 2 の比率で空ける方法である。これに対し、欧文用文字等では、全体の対応よりは、欧文用文字等のまとまりを重視し、字間を空ける処理を採用しないこととした。

グループルビと欧文用間隔 グループルビの親文字列又はルビ文字列に欧文用間隔 (cl-26) を含む場合もあるが、ここでは適用範囲外とした。文字列の長さをそろえる調整に使用する方法と使用しない方法がある。JIS X 4051 では文字列の長さの調整に使用しない方法を規定している。

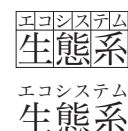


図 12 グループルビの例 1

それぞれの文字列の全長を同じにし、各文字列の字詰め方向の中心をそろえて配置する。空ける量は、ルビ文字列の字間の空き量の大きさ2に対し、親文字列の先頭からルビ文字列の先頭までの空き量及び親文字列の末尾からルビ文字列の末尾までの空き量を1の比率で空ける。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より長いときは、それぞれをベタ組とし、ルビ文字列及び親文字列の字詰め方向の中心をそろえて配置する。この場合、ルビ文字は親文字からはみ出すことになる。

(3) ルビ文字が欧文用文字等、かつ、親文字が仮名等の場合は、次による(図16参照)。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より短いときは、それぞれをベタ組とし、ルビ文字列及び親文字列の字詰め方向の中心をそろえて配置する。

—それぞれをベタ組にした場合のルビ文字列の全長が親文字列の全長より長いときは、親文字列の字間及びその前後を空け、それぞれの文字列の全長を同じにし、各文字列の字詰め方向の中心をそろえて配置する。空ける量は、親文字列の字間の空き量の大きさ2に対し、ルビ文字列の先頭から親文字列の先頭までの空き量及びルビ文字列の末尾から親文字列の末尾までの空き量を1の比率で空ける。

(4) グループルビの場合、親文字列とそれに付くルビ文字の文字列は、一体として扱い、2行に分割してはならない。

(5) 親文字列よりルビ文字列の全長が長い場合、親文字からはみ出したルビ文字を前又は後ろに配置する文字に掛けてよいかどうかはモノルビの処理で説明した方法による(図17参照)。また、親文字列よりルビ文字列の全長が長い場合の行頭又は行末での配置処理も、モノルビの処理で説明した方法による。

熟語ルビの配置処理

熟語ルビの配置処理は、次による。なお、前述の2レベルの処理方式でいえば、以下の説明の(1)、(2)及び(3)項は、第1レベル、(4)項は、第2レベルの配置処理である。

(1) 熟語ルビでは、各親文字とルビ文字とが対応している。この各親文字に対応したそれぞれのルビ文字列をベタ組にした場合において、各親文字に対応したそれぞれのルビ文字列のすべてにおいて、その全長が親文字の文字サイズ以下のときは、次による。

—各親文字に対応したルビ文字が1字の場合は、親文字とルビ文字の字詰め方向の中心をそろえて配置する(図19参照)。

—各親文字に対応したルビ文字が2字以上の場合は、ルビ文字列の字間をベタ組とし、親文字とルビ文字列の字詰め方向の中心をそろえて配置する(図19参照)。

(2) 各親文字に対応したそれぞれのルビ文字列をベタ組にした場合において、各親文字に対応したそれぞれのルビ文字列の全長が1つでも親文字の文字サイズを越えるときは、グループルビと同じ

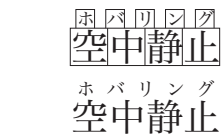
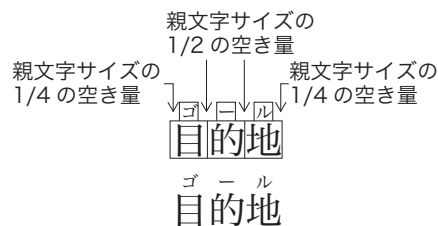


図13 グループルビの例2

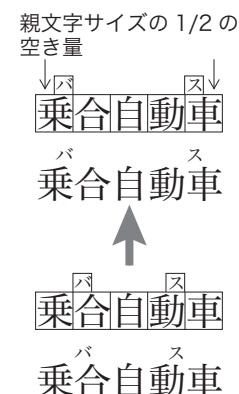


図14 グループルビの例3

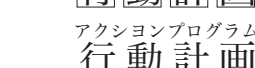
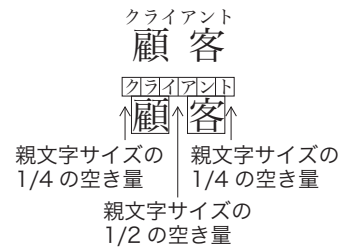


図15 グループルビの例4

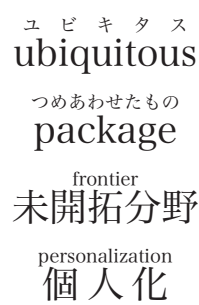


図16 欧字を含むルビの例

配置処理とする（図 20，図 21 参照）。

(3) 熟語ルビは，各親文字間で，各親文字とルビ文字の組合せを維持したうえで，2 行にわたる文字間での分割ができるものとする。この場合，行末又は行頭で親文字が 1 字となったときは，モ

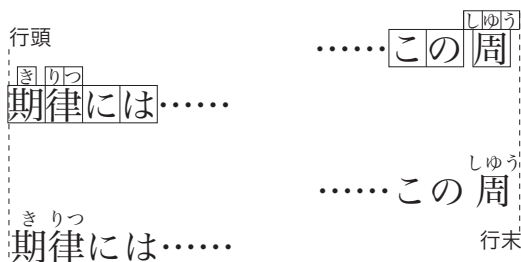


図 22 熟語ルビの分割例

ノルビと同じ配置処理とし，親文字が 2 字以上となったときは，ここで説明した熟語ルビの配置処理とする（図 22 参照）。

(4) ルビ文字列の全長が親文字列の全長より長い場合，親文字からはみ出したルビ文字を親文字の前又は後ろに配置する文字に掛けてよいかどうかは，モノルビの配置処理と同じとする。また，親文字列よりルビ文字列の全長が長い場合の行頭又は行末での配置処理も，モノルビの処理で説明した方法による。

3 両側ルビの配置処理

ルビの種類で組合せで配置を決める

両側にルビを配置する場合，その処理は複雑になるが，簡単な方法として，モノルビ，グループルビ及び熟語ルビの組合せで配置方法を決めることができる。なお，親文字からはみ出したルビ文字の前後の文字との関係，行頭・行末の配置処理は，片側にルビが付く場合と同じである。

両側ルビの組合せ

ルビの組合せとしては，次がある。

- (1) モノルビとモノルビ
- (2) グループルビとグループルビ
- (3) モノルビとグループルビ
- (4) モノルビと熟語ルビ
- (5) 一方が熟語ルビで片方が熟語ルビ又はグループルビ

ルビの種類で組合せと配置方法

JIS X 4051 では，前項の (1)，(2) 及び (3) の組合せの配置方法のみを規定している。ただし，(3) は，連続するモノルビを一つのグループルビとして扱う処理なので，結果として，その配置方法は，(2) である。

ところで，(4) のモノルビと熟語ルビとの組合せは，熟語ルビの熟語を構成する個々の漢字とそれに対応するルビ文字との組合せを一つのモノルビとして扱えば，(1) の方法が採用でき，(5) の一方が熟語

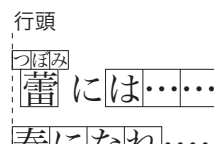


図 17 はみ出しのあるグループルビの例

グループルビの分割 グループルビは一体として扱い分割禁止であるが，活字組版において複合語などでは分割していた。行の調整処理で極端な調整となる例もあったからである。このことを考慮するとグループルビでも，親文字とルビの組合せを考慮した分割ができることが望ましいといえよ

行中

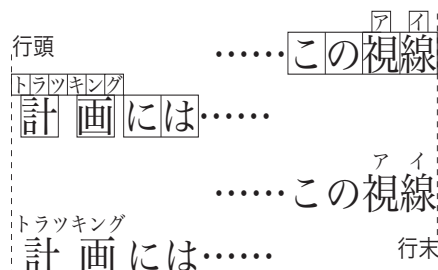
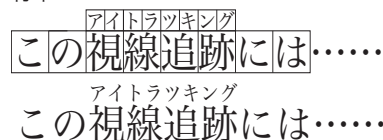


図 18 グループルビの分割例

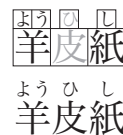


図 19 熟語ルビの例 1

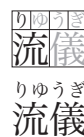


図 20 熟語ルビの例 2

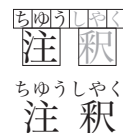
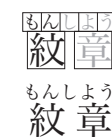
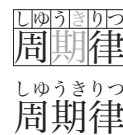


図 21 熟語ルビの例 3

ルビで片方が熟語ルビ又はグループルビの組合せは、熟語ルビをグループルビとして扱えば、(2)の方法が採用できる。

このように、(3)及び(5)は(2)の配置方法と同じ、(4)は(1)の配置方法と同じと考えてよい。そこで、(3)、(4)及び(5)の配置処理は、(1)又は(2)の配置方法を参照してもらうこととし、ここでは、(1)及び(2)の配置方法を説明する。

なお、ルビ文字をどちら側に配置するかは、指定による。

モノルビとモノルビの配置処理

モノルビとモノルビの組合せでは、ルビ文字列の字間はベタ組とし、それぞれのルビ文字列と親文字列とは、字詰め方向の中心をそろえて配置する(図23参照)。その他は、前述したモノルビの配置方法と同じである。

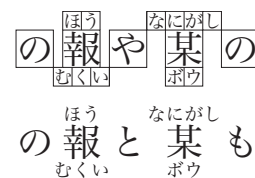


図23 モノルビとモノルビの例

グループルビとグループルビの配置処理

両側のルビ文字列がともに親文字より短い場合は、前述した“グループルビの配置方法”により配置する。ルビ文字が同項でいう仮名等の場合は、それぞれのルビ文字列の字間とその前後を空けて配置する(図24参照)。

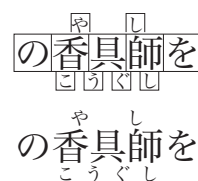


図24 グループルビとグループルビの例1

親文字より長いルビ文字列を含む場合は、長い方のルビ文字列の長さに従い、前述した“グループルビの配置方法”により配置する。親文字が同項でいう仮名等の場合は、親文字列の字間とその前後を空ける。次に、その(仮名等の場合は字間を空けた)親文字列の長さ(前後の空きは含めない)に応じて残りの短い方のグループルビの配置方法を定める。

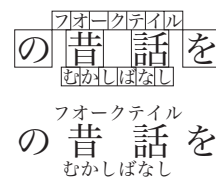


図25 グループルビとグループルビの例2

短い方のグループルビのルビ文字列の長さが(字間を空けた)親文字列長以上の場合は、ルビ文字列はベタ組とし、ルビ文字列と親文字列の字詰め方向の中心をそろえて配置する(図25参照)。

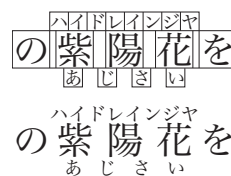


図26 グループルビとグループルビの例3

短い方のグループルビのルビ文字列の長さが(字間を空けた)親文字列長未満の場合は、字間を空けた親文字の長さにそろえて、前述の“グループルビの配置方法”で配置する。ルビ文字が同項でいう仮名等の場合は、ルビ文字列の字間とその前後を空ける(図26参照)。

いずれの場合でも、その他は、前述したグループルビの配置方法と同じである。

両側ルビと行間

両側にルビを付けた行が重なると、行間の設定によっては、隣り同士の行のルビが重なるケースが出る。これは避けないといけない。以下のような方法が考えられる。

- (1) あらかじめ隣り同士のルビ文字が重ならないように、文書全体の行間を設定しておく。
- (2) 重なりが発生した該当する行間だけを広げて、ルビが重ならないようにする。この場合、重なった前の行のルビと、後ろのルビが重ならないだけでなく、例えば、その間は本文文字サイズの四分は空けるとする方法も行われていた。

(3) 該当する行間ではなく、両側にルビの付いた行そのものを複数の行の領域に配置する。両側にルビの付いた行を、例えば、2行分のスペース（2行取り）とする。

(4) 重なりが発生した行間だけでなく、該当する段落全体の行間を広げて、ルビが重ならないようにする。

なお、活字組版では、ルビが多く付く、あるいはルビと共に注の合印などが多く入る場合は（1）の方法、ルビが少ない場合は（2）又は（3）の方法がとられていた。しかし、Webの自動処理を考えると（3）の処理でよいであろう。この方法で該当する行に整数行の領域（例えば2行取り）を割り当てれば、行位置の乱れは、その部分だけですみ、段組とした場合などでは、隣りの段との行位置がそろう。

Appendix ルビ処理の簡略化の内容

親文字とルビの位置関係

1) モノルビ

モノルビでは、親文字とのルビの位置関係で主なものとして以下のような方法がある。ここでは1を採用した。なお、ルビの文字列はベタ組である。

- (1) 親文字の中心とルビ文字列の中心をそろえる。
- (2) 親文字の先頭とルビ文字列の先頭をそろえる
- (3) 親文字の先頭とルビ文字列の先頭をそろえる方法を原則とする。この方式で親文字からルビ文字列がはみ出した場合の処理は、後述の2とし、後ろへのはみ出しを優先する。このため、前後の文字種によりルビと親文字との位置関係は変わることがある。

なお、行頭と行末の処理については、すべての方法で後述の2を選ぶ方式もある。この方式を選んだ場合は、ルビと親文字との位置関係は変わることがある。

2) グループルビ

グループルビでは、ベタ組にした親文字列とルビ文字列の全長が異なる場合、主なものとして以下のような方法がある。ここでは1を採用した（欧字の場合は除く）。

- (1) 短い方の文字列の先頭、末尾及び字間を空ける。先頭及び末尾と字間との比率は、原則として1/2にする。
- (2) 短い方の文字列の字間だけを空け、両方の長さをそろえる。
- (3) 親文字列とルビ文字列ともにベタ組とし、それぞれの先頭、末尾又は中心をそろえる。

なお、グループルビで親文字列からルビ文字列がはみ出した場合、前後の文字種や行頭・行末の位置により、ルビと親文字との位置関係を変えることは、原則としてしないが、中には、位置関係を変える方法も一部では行われている。

3) 熟語ルビ

熟語ルビでは、主なものとして以下のような方法がある。ここでは

1を採用した。2又は3の処理法は簡略であるが、熟語ルビの性質から、あるいは、ルビと親文字の対応方法を変えれば、この方法は採用できることから、採用しなかった。

- (1) 熟語ルビを構成する各親文字に対応するルビの文字列の長さにより、各親文字に対応するルビをモノルビで処理するか、又は熟語ルビ全体をグループルビとして処理する。
- (2) 熟語ルビを構成する各親文字に対応するルビの組み合わせにより、それぞれをモノルビとして処理する。
- (3) 熟語ルビ全体をグループルビとして処理する。
- (4) 熟語内では、各親文字と対応しないルビについて、その前後の漢字に親文字サイズで1/2まで掛かってよいとする処理を行う。この方式では、親文字列からのルビのはみ出しは、後述の2とし、後ろへのはみ出しを優先する。行頭・行末の処理も後述の2とする例が多い。このため、前後の文字種や行頭・行末の位置によりルビと親文字との位置関係は変わることがある。

前後の文字に掛ける処理

親文字からルビ文字列がはみ出した場合、前後の文字に掛けるか掛けないかでは、主なものとして以下のような方法がある。ここでは1を採用した。

- (1) 親文字の前後の文字には、アキのある約物を除き、掛けない。
- (2) 親文字の前後の漢字には掛けないが、仮名や一部の約物には親文字サイズの1/2を掛ける。
- (3) 親文字の前後の文字には、文字種を限らないで親文字サイズの1/4を掛ける。

なお、この処理は、モノルビ、グループルビ、熟語ルビで共通であるが、原則的な処理として2を選択した場合、ルビ文字が片仮名のグループルビに限り1とする方式もある。

行頭と行末の処理

親文字からルビ文字列がはみ出した場合、行頭及び行末の配置方法の主なものとして以下のような方法がある。ここでは1を採用した。2の場合には、モノルビ、グループルビ、熟語ルビにおいて、前述の親文字とルビの位置関係で説明したものとは変わるケースがある。

- (1) 行頭ではルビ文字列の先頭、行末でもルビ文字列の末尾をそろえる。
- (2) 行頭では親文字の先頭、行末でも親文字の末尾をそろえる。

〈参考〉 両側ルビのやや複雑な処理方法

モノルビとグループルビの組合せの場合、次のような方法も考えられる。

まずモノルビとした各親文字とルビの配置を決め（以下、そのまわりを“モノルビ親文字群のブロック”と呼ぶ）、次にそのモノルビ親文字群のブロックを字間を空けないでベタ組で並べ、その全長に対し、グループルビを対応させる処理を行う。グループルビ文字列が、モノルビ親文字群のブロックを並べた全長より短い場合は、グループルビ文字列の字間と先頭・末尾を空ける。長い場合は、モノルビ親文字群のブロック間と全体の先頭・末尾を空ける、という方法である。この方法は、モノルビの親文字とルビとの対応で、ルビ文字列が親文字の字幅以下の場合は、問題がないが、親文字の字幅を超える場合は、グループルビとしてみるとバランスがよい配置とはいえない場合が出てくる。

そこで、すべてのモノルビにおいて親文字の字幅以下の場合と、字幅を超えるケースを含む場合に分け、前者は各モノルビを親文字に配置し、親文字をベタ組で並べ、その上で、グループルビを親文字列に配置する処理を行う。後者はモノルビ全体をグループにし、グループルビとグループルビの組合せという配置方法とする。

この方法は、熟語ルビと熟語ルビの組合せでも適用できる。具体的には、熟語ルビの各親文字とルビ文字列の長さを比べ、次の3つの場合に分ける。

- (1) すべてのルビ文字列の長さが親文字の字幅以下の場合
- (2) 片方で一部のルビ文字列で親文字の字幅を超えるケースを含む場合
- (3) 両側で一部のルビ文字列で親文字の字幅を超えるケースを含む場合

そのうえで、(1) はモノルビとモノルビの組合せのルール、(2) はモノルビとグループルビの組合せで、すべてのモノルビ文字列の長さが親文字の字幅以下の場合のルール、(3) はグループルビとグループルビの組合せのルールを適用するという方法である。

さらに、この方法は、熟語ルビとグループルビの組合せでも考えることができる。熟語ルビの各親文字とルビ文字の対応において、すべてのルビ文字列の長さが親文字の字幅以下の場合と、一部のルビ文字列で親文字の字幅を超えるケースを含む場合に分け、前者はモノルビとグループルビの組合せで、すべてのモノルビ文字列の長さが親文字の字幅以下の場合のルール、後者はグループルビとグループルビの組合せのルールを適用するという方法である。

しかし、こうした方法は、処理が複雑になること、また、JIS X 4051 で採用されていないことから、ここでは、できるだけシンプルな処理方法を考え、こうした方法を採用しなかった。