

アンドールシステムサポート株式会社

プロダクトソリューションチーム

佐々木 陽助 (SASAKI, Yohsuke)

2021年12月13日

Rev 0

# Local Positioning System

## GPSを補間する屋内で利用可能な位置情報の提供方法

### 1. 初めに

今日、GPS（Global Positioning System）は、全世界的な位置情報を提供し、様々な交通システムのインフラにとって、欠かせない存在となっており、その有益性は揺らぐ事が無いものと成っている。また、ノートPCやスマートフォンやタブレットなどの情報端末の上で利用される地図情報と連携した様々なアプリケーションは、道案内のための情報を提供するだけでなく、情報端末の位置に合わせて提供できる様々なサービスを提示することで、広告や仲介手数料を介した利益を地図サービスの提供者にもたらしている。GPSは有益なサービスですが、特にGPSの電波が届かない屋内では、運用できないという弱点を持っている。本ドキュメントでは、GPSの電波が届かない屋内で、デンソーウェーブが開発したQRコードのモデル<sup>2</sup>を利用し、壁や柱、天井、床、案内板の中にそのシンボルを印刷しておくことで、位置情報を提供する低価格な構築できるシステムとしてLPS（Local Positioning System）および、マークからの相対座標を基に、道案内や障害物の情報、操作方法などを提供する、案内マップシステムGMS(Guide Map System)と名付けたオープンで誰もが自由に利用可能な規格を提案するものである。

※「QRコード」はデンソーウェーブの登録商標です。

## 2. GPSを利用する上での課題と解決策

GPSによる位置情報の取得は、GPSを構成する複数の衛星群から提供される電波を位置では、最小4つの衛星から、受信できることを前提とした測位システムとなっている為、必然的に衛星の電波を受信する事ができない屋内やビルの谷間では、利用することができないか、精度の大幅な低下が発生する事となる。また、参考までに携帯電話会社がGPSを利用した携帯電話の位置情報提供サービスの保証精度は、受信感度が良い状況で50mとなっている。

### 2.1 既存の電波発信機を利用したGPS補間技術と課題と解決策

ビルの谷間での運用には、「みちびき」と呼ばれる、準頂点軌道衛星により、ビルの谷間でも、端末が拾う事ができる電波の数を増やすという方式が、日本国内では、提供されている。この仕組みを利用する事ができれば、日本国内限定でセンチメートル精度の位置情報の取得が可能となる。このサービスを利用するためには、情報端末側にGPSに加えて、「みちびき」が提供する電波の受信機能を追加する事が必須となる。

また、GPSや「みちびき」の電波が届かない、屋内で、GPSの電波を利用する為には、IMES (Indoor MESSaging System)の設置が必要となる。

また、GPSの代わりに、屋内でも利用可能な位置情報の提供方法として、Bluetooth等の電波の強度を利用した、Beacon方式がある。Bluetoothを利用したBeacon方式は、イメージセンサーを利用する事が憚られる場所で、位置情報を提供する事が可能であり、送る事が可能な情報量も大きい為、今後は、機器の普及による量産効果によるコストの低下と、エネルギーハーベスティング(環境発電)技術の性能向上により幅広く普及する事を期待するところである。

また、その他の屋内で位置情報を提供する方法として、LED照明を利用した、可視光通信を利用する方法も考えられるが、既存の建物に、新たに適用

する場合には、照明器材の改修や交換が必要となる為、この方式は、費用面が普及の課題となると考えられる。

既存の設備をそのまま利用する方法としては、WiFiスポットが多く存在する場所の場合には、複数のWiFiスポットの電波強度から位置を推測する方法がも提案されている。

GPSの補間技術は、GPS同様に、屋内に於ける社会的インフラの一つの役割を持っているが、上記の全ての方式は、電波発信機の利用を前提としている為、仮に災害が発生し、送電網が麻痺した場合の対策としては、停電に備えて予めバッテリーを電波発信機側に搭載する事であるが、バッテリーの容量には、限りがある為、位置情報を提供する電波発信機側のバッテリーが尽きた時点で位置情報を提供する機能が失われてしまう事となる。

残念な事に、日本では、地震、台風等の震災により、長時間送電網が失われる事態が発生する確率が高い。その為、送電網が麻痺している場合でも、位置情報の提供を受ける機器側のバッテリーが残っている間は、その機能を利用する事ができるバックアップ手段を提供できる事が望ましい。

上記の方式に対し、QRコードを利用した位置マーカは、電子機器ではなく、単なる印刷物であるためマーカ自身は電力が不要であり、マーカを照らす外部からの光源さえあれば、位置情報の提供機能を提供する事ができる。その為、位置情報を求める側の装置にライトが搭載されていれば、電力網が麻痺した場合であっても、その機能は維持される事となる。また、QRコード自身に蓄光塗料を活用すれば、外部からの光源が失われた場合にも短時間であればその機能を発揮する事ができると考えられる。

### 3. 既存のQRコードを利用した情報提供サービスの課題

QRコードは、デンソーウェーブが開発した、汎用的な優れた2次元バーコードである。その優れた特徴から、様々な目的で、活用されている。

#### 3.1 肉眼での判別の課題

QRコードに保存された情報は、バイナリーデータのため、リーダーの機能を持った端末を通さなければ、どのような情報が保存されているのかを、QRコード単体を見ても人間には判断する事ができない。この課題を解消するには、そのQRコードの側にどのような目的で、提供されているのかを利用者が、識別できるシンボルを側に配置する方法が考えらる。

#### 3.2 提供される情報量と印刷面積に関する課題と解決策

QRコードは型番（Version）1から40まであり、型番が大きい程、シンボルも大きくなり、一つのコードに入れられる情報量が増え、型番が小さい程、シンボルが小さくなる。この事は、同一の面積内に型番を変えたQRコードを印刷した場合、型番が小さいもの程、入れられる情報量は減るが、遠くから読み取ることができる。逆に、型番が大きいものは、入れられる情報量は増えるが、遠くからは読み取る事ができないというトレードオフが発生することを意味する。

GPSを補間する使い方をできるようにするためには、印刷されたシンボルに対して離れた距離から読み取れる事が必要となる。遠くからも読み取り易くするためには、型番の小さなQRコードを利用し、できる限り大きく印刷する事が必要となる。小さなバージョンを使うには、盛り込む情報の数を絞り込む事が必要となる。

また逆に、QRコードは、バージョンを大きくすることで、扱える情報の量も大きくなる。遠くから、読み込む事は困難になるが、より大きなバージョンのQRコードを利用する事により、位置情報に加えて案内のための情報を付け加えることもできると考えられる。

現在携帯電話やタブレットに搭載されているイメージセンサーの性能は年々向上しているため、遠くからでも確実に検出可能なQRコードのバージョンは技術の発展により大きいものに対応できるようになっていく事が期待する事ができるが、2021年現在の時点で個人が持ち歩く携帯電話やタブレット機器の性能を考慮した場合、限られた、広さの場所で、遠くからも利用でき、道案内などの情報も提供できる様にするには、独立したQRコードを複数用意し、必須となる情報とそれ以外の情報に分け、必須となる情報は、バージョンの小さなQRコードを使用しできるだけ大きな面積に印刷し、その傍に、詳細な情報をバージョンの大きなシンボルを印刷して配置する方法が考えられる。

#### バージョンの異なるQRコードを並べた例



上に示した例のように、QRコードを遠距離用(この例では左)、近距離用(この例では右)に分けたものを両方用意する事で、遠距離からの認識を可能にし、必要に応じて、詳細な情報を提供する事も可能となる。

QRコードの規格では、1つのQRコードのデータを、より小さなバージョンの複数のQRコードに分割し、最大16個まで連結する事で、縦横比が大きく異なる面に対し効率的に、印刷できる機能を持っているが、この機能は、利用しない事とし、複数のQRコードのうちどれか1つのQRコードしか、認識できなかった場合であっても、十分役立つ情報を入手でき仕様とする。

### 3.3 位置情報の取得に関する課題と解決策

GPSが利用できない場所では、同時にネットワークとの通信もできない電波が届かない場所である可能性もある。このような場所で、QRコードのマーカを利用して、位置情報を取得する場合には、予めネットワークとの通信接続が途切れる前に、今後検出する可能性の高い周囲の全てのQRコードによるマーカを識別し、位置情報と結び付ける為のデータベースを、位置情報の取得を行う携帯情報端末等の位置情報を取得したい機器側にダウンロードしておく方法も考えられるが、QRコードに保存されている情報に、緯度、経度、海面からの高さ、階数名の情報が含まれていれば、データベースに依存しない運用も可能となる。

### 3.3 位置情報の補正に関する課題

GPSは、受信した電波から直接端末が位置を取得できるが、QRコードに、位置情報が結び付いている。従って、QRコードから取得した位置情報に、端末とQRコードの距離情報を算出し、補正を行う必要がある。

LiDARを搭載していない端末もあるので、この様な端末でも距離を割り出せるように、QRコード自体の横幅を情報として埋め込む方法が考えられる。

QRコードの実際の横幅が分かれば、イメージセンサーから得られた画像とレンズフォーカスの情報から高い精度でQRコードと端末からの距離をLiDARなどの別のセンサーがなくても求める事ができる様になると考えられる。

### 3.4 Geotagging情報に関する課題

geo:<緯度>,<経度>,<標高>

Geo情報は、屋外で、位置情報を指定することを想定した形式となっているため、屋内で位置を特定するためには、不向きな内容となっている、標高が分っても階は、別途情報が無いと求めることができない。

屋内で利用するためには、標高よりも階数の情報が必要とされるため、提供する情報に、階数の情報を入れるべきである。

また、屋外では、磁気センサーを利用した方位の測定が容易であるが、屋内では、鉄筋などの影響を受けるため磁気センサーによる方位の推定が困難となる場合も考えられる。従って、方位についての情報もQRコードのデータに入れるべきである。

#### 4. LPSの実現方法について

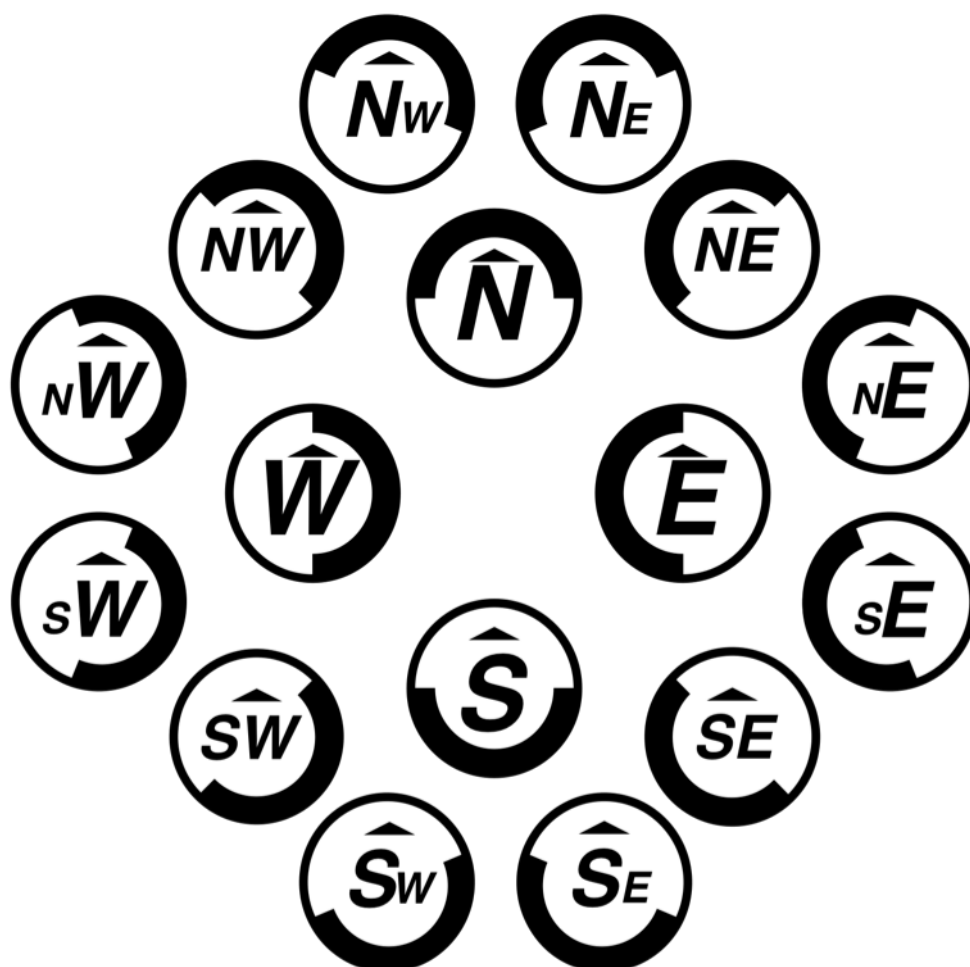
LPSは、情報携帯端末を持ち歩かない人にもメリットを提供する仕組みを提供するものとする。

方位を示すシンボルとQRコードをセットで表記する。

これによりQRコードを読み取る機能を持った情報端末を持ち歩いていない人でも、方位の情報を視覚的に提供することができるようになり、QRコードを読み取る機能を持つ端末を持ち歩く人には、LPSの情報が入っているQRコードが記載されている事に気づく事ができるようになる。

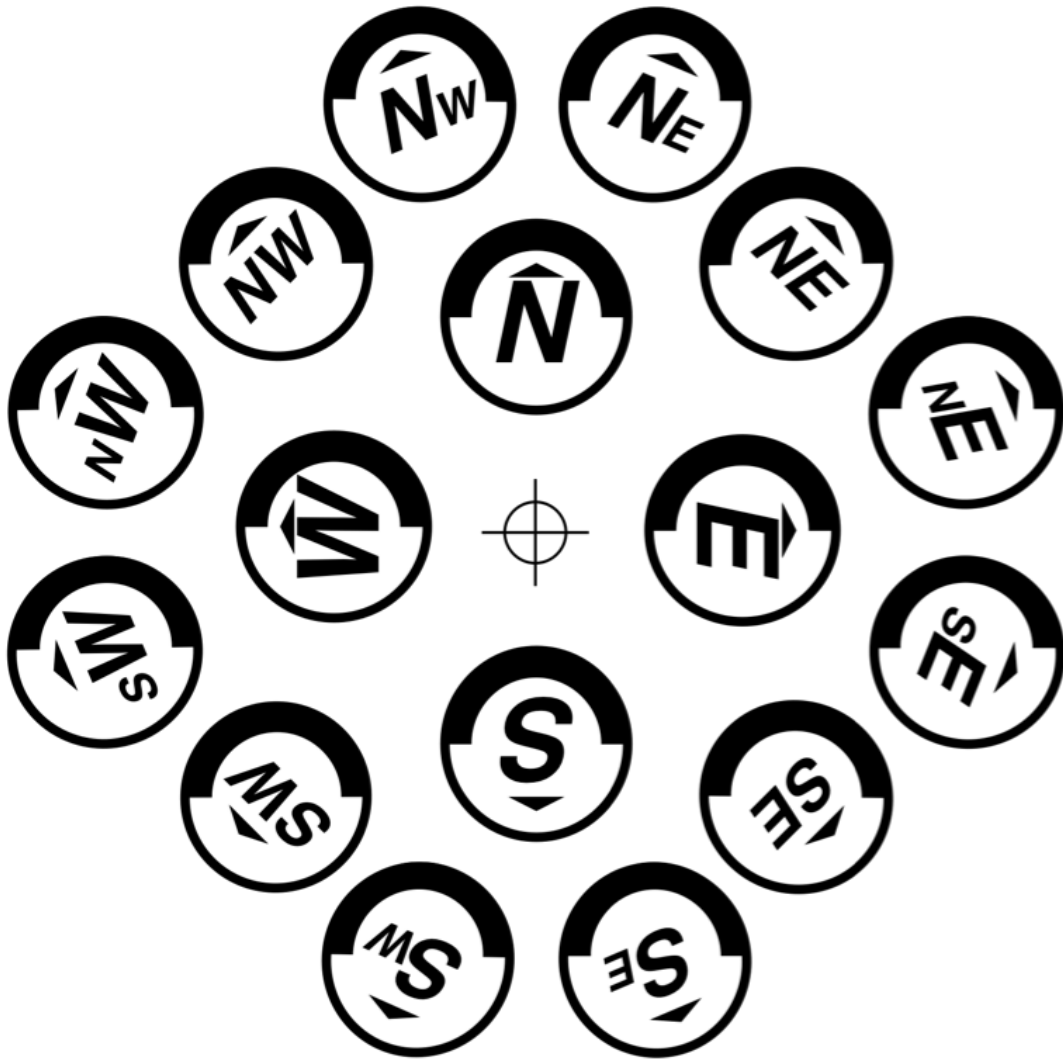
下記に、方位を示すためのシンボルの例を示す。

方位を示すシンボルの例





### 中心点から周囲を見た場合の例



シンボルの色に関しては、検出を容易にするために、白黒の2色を利用することを推奨とするが、モノトーンカラーであれば、周囲のデザインに合わせて、調和の取れた色を選ぶ事もできる。ただし、明度が逆転してはならない。下地が黒地に近い輝度が低い色の場合は、外枠を付ける事が必要となる。

### 北向き用シンボルの白地の場合と黒字の場合の例



- ※LPSの記載場所の周囲に、円を基調としたより重要なシンボルがある場合には、背景が黒地の例のように周囲を四角の黒ベタで囲いより、円を基調としたより重要なシンボルを見落とし難くする配慮が必要となる。
- ※QRコードは、距離が離れると、フォーカスが合わない場合は、検出が困難となるが、LPSのシンボルは、遠くからでも潰れ難いので、イメージセンサーから得られた情報を元に、ニューラルネットワークなどの画像フィルターやパターンマッチングのアルゴリズムを利用する事で、離れた位置から方位の情報を取得するマーカの役割も狙ったデザインとした。
- ※LPSのシンボルの中に斜体の文字を配置する事によ、イメージセンサーが捉えた際に、虚像が写り込んでいる事を容易に検出できる事も考慮したデザインとした。
- ※QRコード本体は、ISO規格、JIS規格で定められたものを使用すべきであるが、LPSやGMSの中では、QRコードモデル2を利用する事以外は定めないものとするが、広く不特定多数の人に利用される公共スペースでLPSやGMSを設置する場合は、市販の普及している携帯情報端末のイメージセンサーにより無理なく読み取る事ができる事を、確認する義務を負うものとする(特定のメーカーの特定の機種でしか読めないものは、規格に反している)。逆に、特定企業の構内などで外部の人間が立ち入らない場所で利用されるものやセキュリティを要求される場所で用いられるものについては、市販の普及している携帯情報端末から読み取る事の確認は義務としない。また、この様な場所で、LPSやGMSを利用する必要がある場合には、通常のQRコードではなく、デンソーウェーブ社のSQRCの適用を検討すべきである。
- ※LPSのシンボルの利用は、努力義務であり、必須では無い。印刷するスペースが不足している場合や、デザイン上の都合により、LPSのシンボルと一緒に配置する事が不適切な場合は省略しても良い。これは、シンボルマークを入れる事ではなく、QRコードを利用した位置情報提供システムを利用できる機会を増やすこと事態が多くの人に役立つ事に繋がるからである。

## 5. LPSの適用範囲

LPSは単体で使用するものではなく地図と合わせして利用される事を想定した仕組みとなっている。

主に公共空間において、特定の地点に対し、特にGPSの電波の受信を期待できない、屋内、地下街において、GPSを補完する位置情報を直接QRコード内に直接付与する方法を提供する仕組みである。

LPSはGPSが有効に機能しない、屋内屋、ビルの谷間、トンネル地下道、地下街やジオフロンティアで、位置情報のみならず方位の情報を提供する事ができる。

この仕組みは、情報携帯端末を持ち歩く人のみならず、イメージセンサーを搭載しているものであれば、車の自動運転、ドローンや配送ロボットなど、目的地に対し自律移動する様々な対象に、位置情報を補正する為に役立つものとなると考えられる。

LPSは特定の地点に対する絶対位置を基盤とするコード体系の為、対象の位置が移動するもの、例えば、鉄道や、バス、飛行機や船の船内等に対して、LPSを適用することはできない。

これに対し、GMSは、絶対位置を持たない相対位置情報だけで構成されている為、LPSが利用できない鉄道や、バス、飛行機や船の船内等にも適用する事ができる。

LPSやGMSは位置の繋がりを記述する事ができるので、デジタル案内板としての機能を持たせる事ができる。

LPS、GMSは、QRコード上に、全ての情報を格納し、携帯情報端末等のエッジ上で自己完結できる事を目指した処理体系を想定したものとなっている

が、ネットワーク上に存在するデータベースに常に通信ネットワークを通してアクセスできる環境下であれば、バーコードによる商品管理システムの考え方を拡張し、あらゆるモノに、個別のIDを割り振るという考え方については、トロンフォーラム傘下のユピキタスIDセンターが提供するucodeと呼ぶ規格を利用する方がより柔軟性に優れている。

QRコード自身に詳細なデータを保存する本手法では、QRコード自身に保存されているデータ自身を書き換えるには、直接、印刷内容を書き換える必要がある(位置情報と更新内容をセットにしたデータベースを用意する事で、更新をかける手法も考えられるが、LPSやGMSでは、QRコード自身を書き換える事が望ましい)。

また、本ドキュメントとで示した実装例は、QRコード自身に、位置情報を直接付与しているが、トロンフォーラム傘下のユピキタスIDセンターが提供するucodeを利用し、サーバー側に、上記の情報を保存すれば、間接参照の形で、LPSが提供する情報を入手する事もできる。

これは、単純な、位置情報や地図サービスの提供のみならず、ARによるデジタルサイネージサービスにも活用できると考えられる。

位置情報と地図サービス、都市OSの仕組みを関係させる事で、坂村 健氏がTRONプロジェクトのなかで提唱したイネーブルウェアの実現に役立つ基盤となると考えられる。

地図サービスに、バリアに関する情報を追加し、信号機、駅のフォームなどで、向きと位置情報を情報端末が収集、都市OSを介して、誘導を行う。公共のスペースで、目の不自由な方にもバリアフリーな社会を実現する。日本語が読めない海外の方にも、適切な情報を提供する事ができるようになる。また、省力化のために、無人で移動する装置(搬送ロボット、警備ロボット、ドローン)に対し、位置情報を提供するための手段としても利用できる。

特にGPSの電波を利用できない、地下街、地下鉄の駅、トンネル、ビルの谷間における、GPSと方位磁石の補完をする事ができるようになると考えられる。

## 6. LPSのQRコード上の基本書式

LPSデータは以下に示す書式を持つ。

LPSの基本書式:

**LPS:**<プロパティ 1><プロパティ 2> ... <プロパティ n>.

「**LPS**」という識別子で始まり、コロン(:)で、プロパティと分けられ、ピリオド(.)で終わるものとする。

プロパティの書式 1:

<プロパティ名>:<データ>.

プロパティの書式 2:

:<データ>.

<プロパティ名>:

プロパティ名はアルファベットの大文字から始まりアラビア数字を入れる事もできる。

プロパティ名とデータはコロン(:)により分けられる。

<データ>:

データは、コロン(:)の後から始まり、ピリオド(.)の手前で終わるものとする。また、データの中に、別のプロパティや、複数のパラメータを並べて入れる事ができるものとする。データの書式は、プロパティ事に別途定めるものとするが、1つのピリオド(.)が終わりを示すデータは、直近に現れた1つのコロン(:)に対応するという規則は、どのプロパティであっても常に適用される(\*)。

プロパティの書式2は、前に登場したプロパティと、同じプロパティが連続する場合に、利用できる書式である。階層毎に同一のプロパティが連続して現れる場合には、プロパティ名を省く事ができる。

※データの中に入れられた、パラメータの中の文字列の中で、コロン(:)、ピリオド(.)は、それぞれエスケープ文字を利用して「/:」、「/。」とえて表記する事ができるものとするが、エスケープ文字で書かれたものは、文字列の一部と解釈される為、識別子とプロパティの区切り、プロパティ名とデータの区切りや終わりを示す区切り文字としては扱われることはなものとす。

※プロパティの書式2は、プロパティ名を省略する事で、文字数を減らす事を目的とした書式である。プロパティが持つデータの中に、子のプロパティが入る入れ子構造になっている場合に、対応する省略されたプロパティは、同じ階層のプロパティとなる。

例1:

**α:β:<データ1>.β:<データ2>..α:ω:<データ3>.ω:<データ4>..**

上記と同じ内容を下記の様に表現する事ができる。

**α:β:<データ1>.:<データ2>...ω:<データ3>.:<データ4>..**

例2:

**α:β:<データ1>.β:<データ2>..α:β:<データ3>.β:<データ4>..**

上記と同じ内容を下記の様に表現する事ができる。

**α:β:<データ1>.:<データ2>...:<データ3>.:<データ4>..**

プロパティ名は階層毎に管理されている。

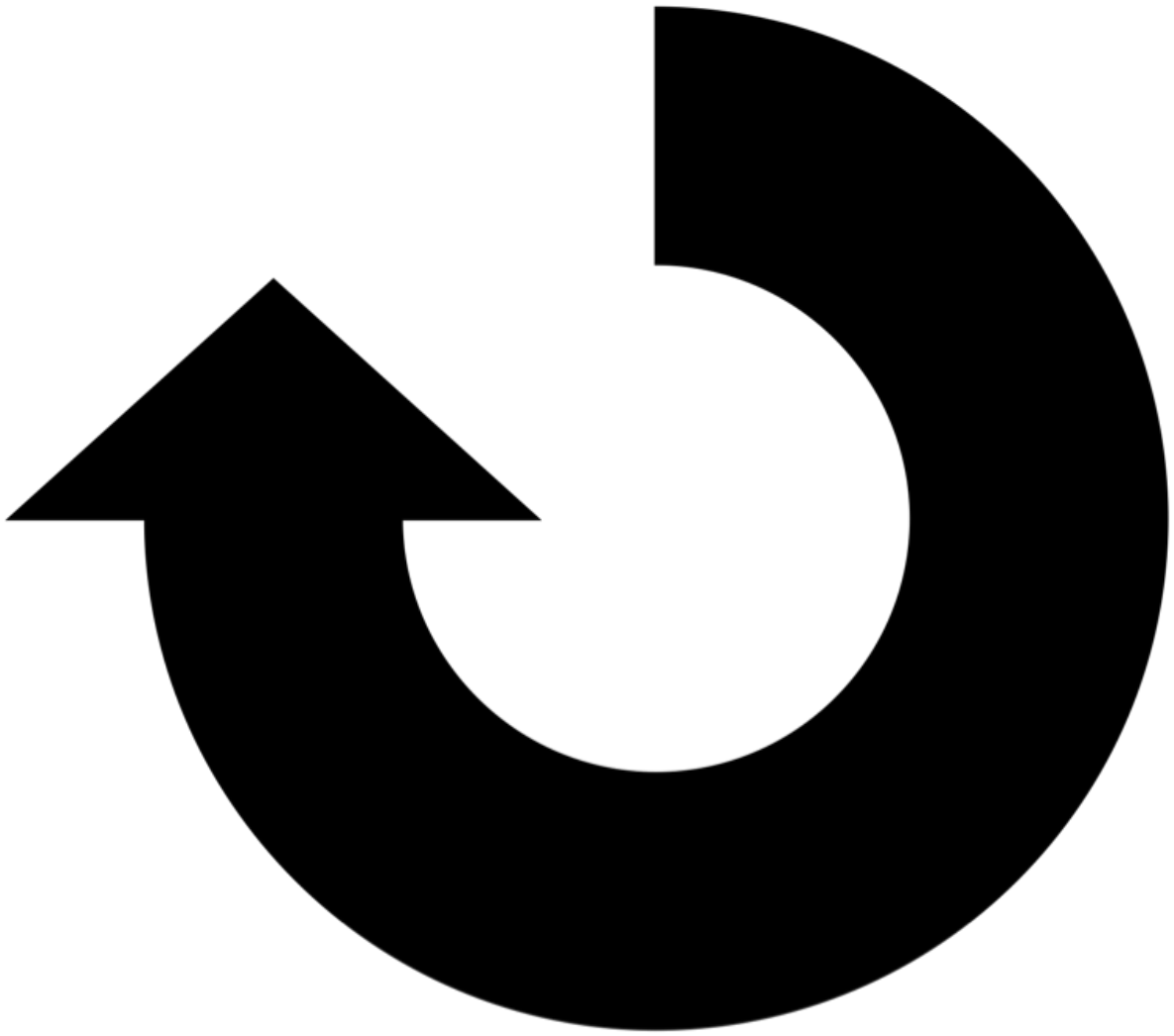
※実際にはαβωのギリシャ文字ではなく、英数字が使われる。プロパティ名にギリシャ文字を使用する事はない。

## 6.1 LPSが持つプロパティ名

LPSのプロパティ名と情報の対応を下記の表に示す。

プロパティ名	プロパティの名称	主な情報
A	<領域のプロパティ>	領域の有効半径
B	<バリア情報のプロパティ>	バリア情報
C	<サブ領域円形のプロパティ>	サブ領域円形
D	<地球上の方角のプロパティ>	地球上の方角、標高
E	<緊急設備のプロパティ>	非常設備の情報
F	<階数名のプロパティ>	階数の名称と特徴
G	<ゲートのプロパティ>	ゲートの情報
H	<サブ領域凹形のプロパティ>	サブ領域凹形
J	<サブ領域多接続形のプロパティ>	サブ領域多接続形
L	<固有識別番号のプロパティ>	固有の識別番号
M	<マークのプロパティ>	マークの取付け角度と横幅、高さ
N	<外部マークの情報のプロパティ>	連絡先のQRコードがある方向と距離
P	<地球上の位置のプロパティ>	地球上の位置
Q	<サブ領域長方形のプロパティ>	サブ領域長方形
R	<繰り返し指定のプロパティ>	サブ領域の繰り返し指定
S	<サブ領域多頂点形のプロパティ>	サブ領域多頂点形
T	<サブ領域樹木形のプロパティ>	サブ領域樹木形
U	<ユーザーインターフェースのプロパティ>	ユーザーインターフェースの情報
W	<英語以外の言語の指定>	英語以外の言語の情報
X	<角度と長さの定数宣言のプロパティ>	角度と長さの定数宣言
Y	<マークの作成年月日のプロパティ>	作成年月および時分秒
Z	<文字列定数の宣言のプロパティ>	文字列定数の宣言

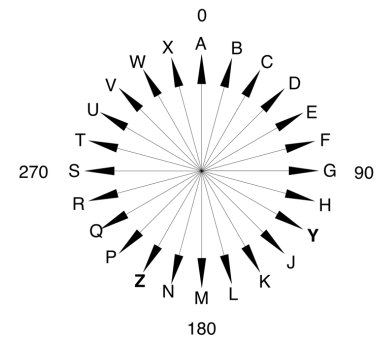




## 6.2 角度の指定

<角度のパラメータ>は、角度のスタック定数と、下記の置換表と計算式によりアルファベットで符号化したもので指定する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X



アルファベット1文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

例: A=0度、B=15度、D=45度、E=60度、G=90度、Y=120度、M=180度

0度から15度刻みで345度まで表現できる。

アルファベット2文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

+360/24/24\*<2文字目の対応表で示される数値>

例: AB=0.625度、GQ=97.5+2.5=100度

0度から0.625度刻みで359.375度まで表現できる。

1文字のAに対し2文字のAAはより高い有効精度の場合に使用される。

アルファベット  $n$  文字で表記する場合:

$$y = \sum_{k=1}^n \frac{x_k}{24^k}$$

$y$  = 角度

$x_k$  = アルファベットで表記された対応する数値

$k$  = アルファベットの左から数えた文字数

$n$  = アルファベットの全体の文字数

角度とは  $2\pi$  を分割する値に対する比率であり、度数法では 360 等分した際の割合で示すものである。LPS では 360 等分ではなく、 $24^n$  等分するという考えに基づいている。その為、 $n = 1$  の場合には 24 等分と精度が低くなるが、 $n = 2$  の場合には 576 等分となり、度数方で整数の範囲で表記する場合よりも精度が良くなる。

従って実用上、アルファベット 1 文字から 2 文字で十分な精度が得られる事が期待できるが、必要に応じて 3 文字以上で、より高い精度を用いる事ができる。

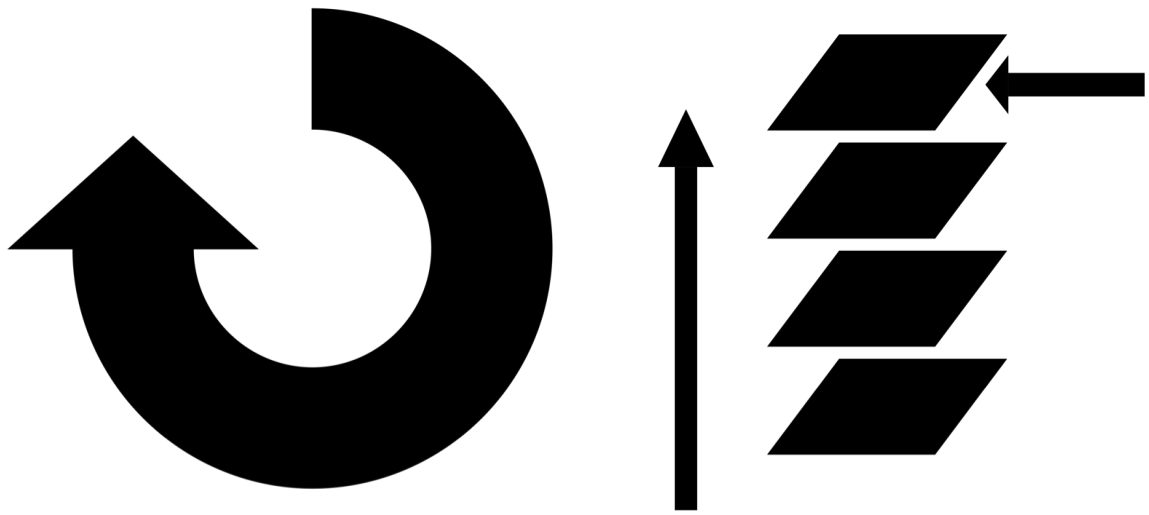
※符号化する際にアルファベットの大文字の I と O はアラビア数字の 1 と 0 との判読が困難となるため、それぞれ大文字の Y と Z に置き換える。

※アルファベットから数値に復号する際は、アルファベットの大文字の I は数値 8、アルファベットの大文字の O は、数値 14 として扱うものとする。

※アラビア数字ではなく、アルファベットでコード化する理由は、より少ない文字数で多くの情報を入れる事を可能にし、精度と情報量のトレードオフを可能にする為である。

角度アルファベット符号化表

数値	文字	1文字目の角度	2文字目の角度	3文字目の角度
0	A	0	0	0
1	B	15	0.625	0.0260416666666667
2	C	30	1.25	0.0520833333333333
3	D	45	1.875	0.078125
4	E	60	2.5	0.104166666666667
5	F	75	3.125	0.130208333333333
6	G	90	3.75	0.15625
7	H	105	4.375	0.182291666666667
8	Y	120	5	0.208333333333333
9	J	135	5.625	0.234375
10	K	150	6.25	0.260416666666667
11	L	165	6.875	0.286458333333333
12	M	180	7.5	0.3125
13	N	195	8.125	0.338541666666667
14	Z	210	8.75	0.364583333333333
15	P	225	9.375	0.390625
16	Q	240	10	0.416666666666667
17	R	255	10.625	0.442708333333333
18	S	270	11.25	0.46875
19	T	285	11.875	0.494791666666667
20	U	300	12.5	0.520833333333333
21	V	315	13.125	0.546875
22	W	330	13.75	0.572916666666667
23	X	345	14.375	0.598958333333333



## <角度のスタック定数>

<角度のパラメータ>は、現れた順に、スタックに順に積まれ行く、スタック変数は、スタックに積まれたパラメータの値を定数として利用する為の仕組みである。

<角度のパラメータ>を保存するスタックは、2種類用意されている。

次の2種類が用意されている。

<シングルデータの角度スタック>

<マルチデータの角度スタック>

<シングルデータ角度スタック>には、2文字以上のアルファベットで指定された角度を保存する。この角度スタックに保存される値は、最後の1つだけとなる。

<マルチデータの角度スタック>には、3文字以上のアルファベットで指定された角度を保存する。この角度スタックに保存される値の数には制限を持たない。

3文字以上の<角度のパラメータ>が利用された場合には、<シングルデータの角度スタック>と<マルチデータの角度スタック>の両方に、値が保存されるものとする。

<角度のスタック定数>では、下記に示す2つの書式を利用できる。

#### 書式 1

**%**

最後に<シングルデータの角度スタック>に入った値を参照するものとする。

過去に2文字以上の角度のパラメータが利用されている場合にだけ、この<スタック定数>を利用する事ができる。

#### 書式 2

**%<アルファベット>**

<マルチデータの角度スタック>に入った値を参照するものとする。

**<アルファベット>:**

<マルチデータの角度のスタック定数>では、24進数でアルファベットを利用してスタック上で参照したいデータの位置を指定するものとする。アルファベットと24進数との対応は、下記の表に従うものとする。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

アルファベットAは、数値0に相当し、<マルチデータの角度スタック>に、最後に入れられた値を参照することができる。

<角度のスタック>に入っている量を超えた値を指定してはならない。

<角度のスタック>の値を参照してもスタックは、低くならない。参照されたスタックの値は、更にスタックの最後に積まれるものとする(文字数が2文字だった場合は、<角度のシングルデータのスタック>のみに、3文字以上の場合は、両方のスタックに入れられる)。

<角度のスタック定数>で利用される、<角度のパラメータ>用のスタックには、<角度のパラメータ>専用のスタックである。他のパラメータが入る事は無いものとする。

この仕組みは、角度以外のパラメータをフィルタリングにより除去する事で、より少ない文字数で、より多くの<角度のパラメータ>の値を参照できる様にする為である。

便宜上スタック定数と読んでいるが、スタックは、常に積まれる一方向のみの動作となり、事態としては、辞書圧縮を実現する為の仕組みの一種と言える。従って、スタック定数は、直接値を指定する場合よりも文字数が短くなる場合だけ利用される。

<角度のパラメータ>の始まりと終わりの文字の特徴:

<角度のパラメータ>の始まりの文字:

アルファベットまたは、%

<角度のパラメータ>の終わりの文字:

アルファベットまたは、%



<角度のスタック>に、<角度のパラメータ>が保存される迄の流れ:

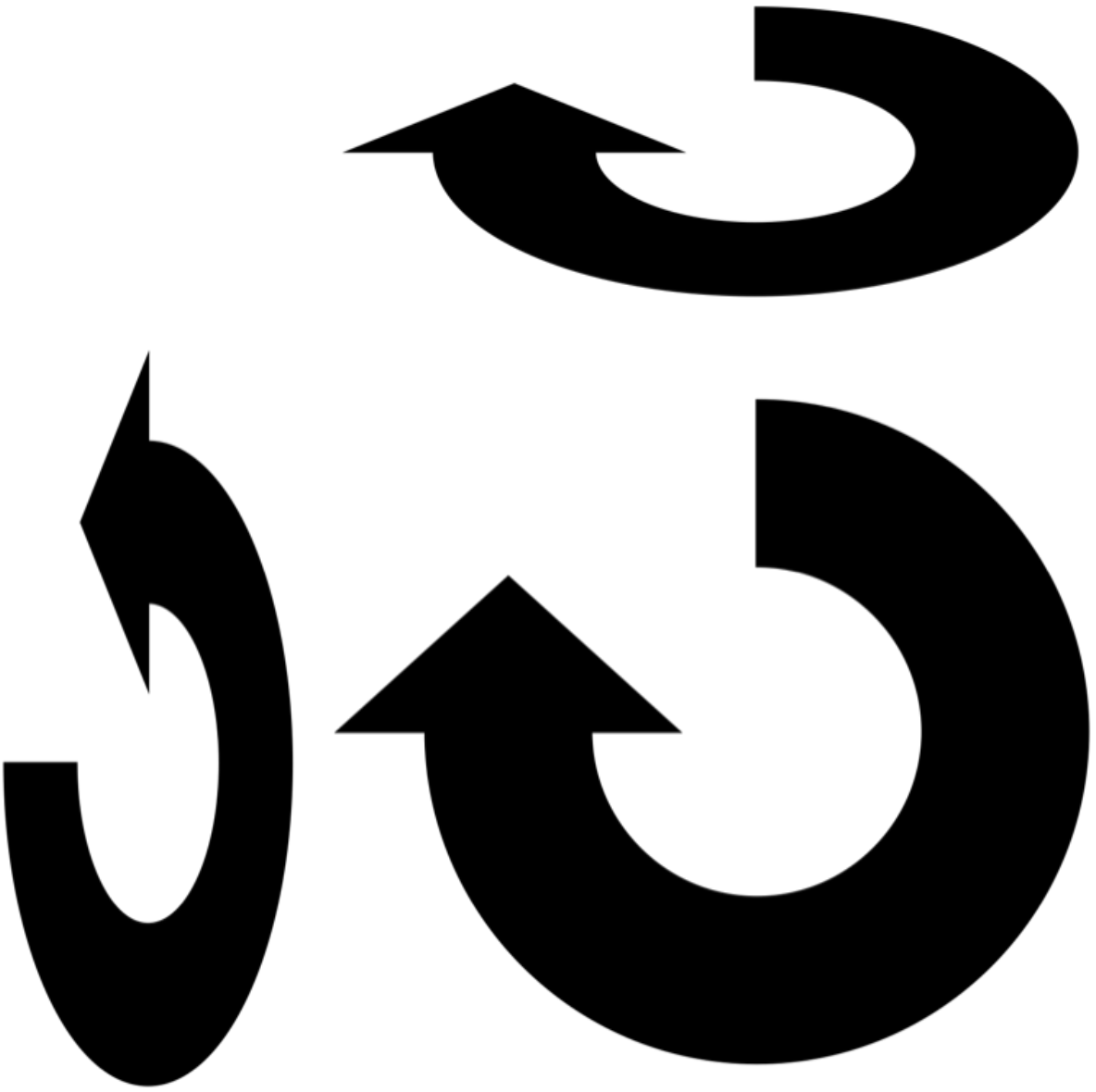
1つの<長さのパラメータ>の中で、<角度のパラメータ>の値を複数回呼び出す事ができる。

<長さのパラメータ>の中で、呼び出された場合は、呼び出された順番でリスト構造に一旦保存され、<長さのパラメータ>の終わった時点で重複を取り除いた上でそれぞれの<角度のスタック>で保存される。

この動作を詳しい手順は次の通りである。

- (1) リストに最後から最初に向かって値が重複するデータが存在するかどうかを確認する。
- (2) 重複しているデータが存在する場合には、一番最後の順番にあるデータだけが残され、重複は削除される。
- (3) <長さのパラメータ>の終わった時点で、リストに入れられた順番を守って、文字数に従って<シングルデータの角度スタック>と<マルチデータの角度スタック>に保存される。

<長さのパラメータ>の中以外では、リスト構造を使わずに直接<角度のスタック>に直接保存される。



### 6.2.1 <3軸回転角度のパラメータ>

<3軸回転角度のパラメータ>は、形状の相対角度を、<yaw角度>、<pitch角度>、<roll角度>の3つの角度を指定するパラメータである。

#### 書式1

<yaw角度>\*<pitch角度>\*<roll角度>

#### 書式2

\*<pitch角度>\*<roll角度>

<yaw角度>=0

#### 書式3

<yaw角度>\*\*<roll角度>

<pitch角度>=0

#### 書式4

<yaw角度>\*<pitch角度>

<roll角度>=0

書式5

<yaw角度>

<pitch角度>=<roll角度>=0

書式6

\*<pitch角度>

<yaw角度>=<roll角度>=0

書式7

\*\*<roll角度>

<yaw角度>=<pitch角度>=0

書式8

全て省略

<yaw角度>=<pitch角度>=<roll角度>=0

## <3軸回転角度の指定>のパラメータ

### <yaw角度>

<yaw角度>は<角度のパラメータ>である。

地面に平行な回転角度を指定するためのパラメータである。

図形が、仮に、x-z平面上で<pitch角度>、<roll角度>ともに、0度の場合に、<yaw角度>は、y軸に対する回転角度となる。

この場合の角度は、上から地面を見下ろした際の角度となり、上から見て、角度が増加すると図形は、地面に対して時計回りに回転する事となる。

### <pitch角度>

<pitch角度>は<角度のパラメータ>である。

地面に対する傾きを指定する為のパラメータである。

進行方向に対し上下方向の傾きを指定する役割を持つ。

図形が、仮に、<yaw角度>、<roll角度>ともに、0度の場合は、<pitch角度>は、x軸に対する回転角度となる。

この場合の角度は、上から地面を見下ろした際に、0度で、x-z平面と平行となる。LPS及びGMSでは、ユーザーインターフェースなるものを除き地面に対し、垂直に立っている状態を<pitch角度>=0度としているため、90度迄は、原点から、z軸方向に傾く事となる。

## <roll角度>

<roll角度>は<角度のパラメータ>である。

地面に対する傾きを指定する為のパラメータである。

進行方向に対し左右方向の傾きを指定する役割を持つ。

図形が、仮に、<yaw角度>、<pitch角度>ともに、0度の場合  
は、<roll角度>は、z軸に対する回転角度となる。

この場合の角度は、上から地面を見下ろした際に、0度で、x-z  
平面と平行となる。LPS及びGMSでは、ユーザーインター  
フェースなるものを除き地面に対し、垂直に立っている状態を  
<roll角度>=0度としているため、90度迄は、原点から、x軸方向  
に傾く事となる。



### 6.3 長さの指定

<長さのパラメータ>は、必ずアラビア数字列から始まるものとする。単位はメートル法で処理されるが、下記に示す表に示すアルファベットの組み合わせにより係数や演算を指定する事ができる。<長さのパラメータ>の先頭に於いて、アラビア数字に続いてパーセントマーク(%)が続く場合には、<長さのスタック定数>の規定が適用される。

#### 単純係数を示すアルファベット

B	C	D	F	G	H	J	K	L	M	N	Q	R	T	U	V	W	Y	Z
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

#### 係数演算子表

係数記号	値	注
A	A系用紙の短辺	※
B	$10^{-3}$	
C	$10^{-2}$	
D	$10^{-1}$	
E	$10^n$	※
F	0.3048	
G	$10^9$	
H	$10^2$	
J	1/33	
K	$10^3$	
L	1609.344	
M	1	

係数記号	値	注
N	1852	
P	+	※
Q	1/8	
R	1/3	
S	$1/(\cos(n/2))$	※
T	$10^6$	
U	$10^4$	
V	$10^5$	
W	1/6	
X	÷	※
Y	0.9144	
Z	0.0254	

※米印が注の欄に入っているものは、特殊係数か演算子を示す。米印が入っていないものは、単純係数を示す。



- ※係数は、少ない文字数で良く日常的に使われる値を表現する為に都合の良いものを選択した。
- ※係数に10が無い理由は、アラビア数字で0を追加した場合と比べて文字数が変化しない為である。
- ※アルファベットの大文字のIとOは、アラビア数字の1と0との判別が、困難な場合があるため、係数を表す文字としては利用しないものとする。

### 6.3.1 単純係数

単純係数はアラビア数字の列から始まり、その次に、単純係数を示すアルファベットが続くものとする。書式1で示すように、続くアルファベットを省略することもできる。

#### 書式1

<アラビア数字列1>

<アラビア数字列1>が整数の値として解釈される。

#### 書式2

<アラビア数字列1><単純係数 1>

<アラビア数字列1>が整数の値として解釈され<単純係数 1>の値と掛け合わされる。

#### 書式3

<アラビア数字列1><単純係数 1><アラビア数字列2>

<アラビア数字列1>が整数の値、<アラビア数字列2>が続く小数点の値として解釈され<単純係数 1>の値と掛け合わされる。

#### 書式4

<アラビア数字列1><単純係数 1><アラビア数字列2><単純係数 2> ... <単純係数 n> ...

<アラビア数字1>が整数の値、<アラビア数字列2>が続く小数点の値として解釈され<単純係数 1>からn迄の値と掛け合わされる。書式4ではnは2以上の値となる。

#### 書式5

<アラビア数字列1><単純係数 1><単純係数 2> ... <単純係数 n> ...

<アラビア数字列1>が整数の値として解釈され<単純係数 1>からn迄の値と掛け合わされる。

書式5ではnは2以上の値となる。

書式4、書式5の後に、アラビア数字の列がある場合は、その後ろに新たに、書式1から書式5が存在すると解釈され、前の書式4または書式5で得られた数値と、続く後ろの書式で得られた値は、掛け合わされる事となる。

## 書式6

<単純係数 1><単純係数 2> ... <単純係数 n> ...

書式6は、アラビア数字で終了する特殊係数の後にだけ現れる。

特殊係数で求められた値と単純係数1からnまでの値とが掛け合わ

される。書式6の後には、書式1から書式5を続けて各事ができ

る。この場合は、書式6までに求められた値と後続の書式で得た

値が掛け合わされる事となる。

### 係数を利用した表記の例

表現したい値	実際の表記	他の表記1	他の表記2	他の表記3
12mm	12B	1C2	0D12	0M012
2cm	2C	20B	0D2	0M02
482.6mm	19Z	482B6	48C26	0M4826
965.2mm	38Z	38ZM2	96C52	965B2
88cm	88C	880B	8D8	0M88
90cm	9D	90C	0M9	900B
2m	2	2M	200C	2000B
12m	12	12M	120C	120B
20m	20	20M	200C	200B
12.5m	12M5	125D	1250C	12500D
91.44m	3HF	91M44	914D4	9144C
200m	2H	200	200M	20000B
1609.344m	1L	1609M34 4	160934C4	1609344 B
2km	2K	2000M	2000	2000000 B
2.3km	2K3	2K3M	2300M	2300
10km	1U	10K	10000	10000M
16km	1U6	16K	16000	16000M
16093.44m	10L	16093M4 4	16093M44 M	1609344 C
2000km	2T	2KK	2000K	2000000

※単純係数は、最小限の文字数で様々な値を表記する為に用意されたものである。従って、次の表で示す様に、同じ値を示す為に、複数の表記の方法が存在するが、できる限り目的を達成可能な範囲で、最小限の文字数になる組み合わせを選ぶべきである。

※19ZM2と表記した場合は、 $19 \times 0.0254 \times 1 \times 2$ と解釈され、965.2mmの値となる。19ZM2の代わりに38Zと表記した方が最小限の文字数に収まる。

### 6.3.2 特殊係数

特殊係数は、単純係数の後または、特殊係数、演算子、スタック定数の後から始まるものとする。演算子以外の、単純係数、特殊係数、スタック定数の値が前にある場合は、特殊定数の値と掛け合わされる。

特殊定数の後に、演算子以外の、単純係数、特殊係数、スタック定数の値が後ろにある場合も、特殊係数の値と掛け合わされる。

#### 書式1

##### **A**<アラビア数字>

アラビア数字は、ISO 216のA列の番号を示す。この係数は、指定された用紙サイズの短辺の長さを示す。

この後に単純係数が続く場合には、単純係数の書式6が適用される。

#### 例外規定:

マークの取り付け角度及び横幅のプロパティが持つパラメータの1つであるマークの横幅の中では、QRコードのクワイエットゾーンのを含めて最大の大きさを印刷した際の、切り出しマークの端から端までの横幅を示す。

QRコードのクワイエットゾーンは、QRコードを構成する1つのセル(モジュール)のサイズ掛ける4となり、選択された型番(バージョン)に応じて、セル数が変化する為、結果的に同じA4用紙に印刷する場合でも、バージョンが小さいほど、余白であるクワイエットゾーンが大きくなり、バー

ジョンが大きい程、余白であるクワイエットゾーンが小さくなる。

## 書式2

### **E**<アラビア数字>

$10^n$ の値を示す。後ろに続くアラビア数字は、 $n$ の値を示す。

この後に単純係数が続く場合には、単純係数の書式6が適用される。

\*  $10^{-1}$ ,  $10^{-2}$ ,  $10^{-3}$ の値は、それぞれD、C、Bで表現できるため、 $n$ は正の値のみを扱うものとする。



### 書式3

#### **S<角度>**

特殊係数Sで指定する角度は、<角度のパラメータ>である。アルファベットの文字列で符号化した値を利用する。

アルファベット1文字の場合は、0度から15度刻みで345度まで、アルファベット2文字で表記する場合は、0度から0.625度刻みで359.375度まで表現できる。

角度の値の部分にスタック定数を利用する際には、角度のパラメータのスタック定数が適用される。

<角度のパラメータ>として利用された値は、角度のスタックの最後に新たに積まれる事となる。

係数として戻る値は下記の式によるものとする。

$$y=1/\cos(n/2)$$

y: 係数として求められる値

n: アルファベットで符号化された角度

例外規定:

n = 180の場合は、y = -1とする。

cosの値は、90度の際に0を返す。上記の式では、角度を2で割っているため、n=180度の場合に、値が無限大に発散してしまうためこの規定を設ける事とする。

後続に対する影響:

直後に演算子を置く事はできない。

単純係数が後ろに続く場合は、単純係数の書式1から書式5の何れかが適用されるものとする。

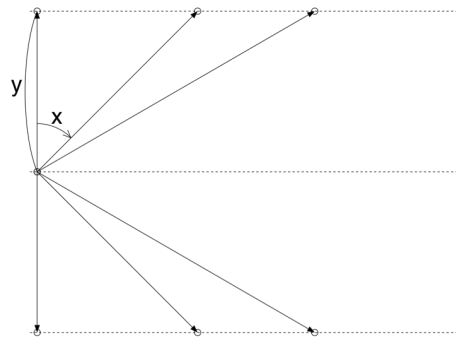
※cosは180度から360度のまでの範囲で返す値は、0度から180度の値を折り返して得られる値の範囲と一致する(180度から360度までの値は、0度から180度で得られる値を鏡に写したグラフとなる)。その為、180度を越えた値を扱う事は無駄となる為、0度から360度の値を2で割る事により、少ないデータ数でより精度を稼ぐ事ができる仕様とした。以降のページにこの特殊係数により得られる値の例を示す。

アルファベット1文字で得られる係数

アルファベット	n	x	係数
A	0	0	1.00000000
B	15	7.5	1.00862896
C	30	15	1.03527618
D	45	22.5	1.08239220
E	60	30	1.15470054
F	75	37.5	1.26047241
G	90	45	1.41421356
H	105	52.5	1.64267963
Y	120	60	2.00000000
J	135	67.5	2.61312593
K	150	75	3.86370331
L	165	82.5	7.66129758
M	180	90	-1.00000000
N	195	97.5	-7.66129758
Z	210	105	-3.86370331
P	225	112.5	-2.61312593
Q	240	120	-2.00000000
R	255	127.5	-1.64267963
S	270	135	-1.41421356
T	285	142.5	-1.26047241
U	300	150	-1.15470054
V	315	157.5	-1.08239220
W	330	165	-1.03527618
X	345	172.5	-1.00862896

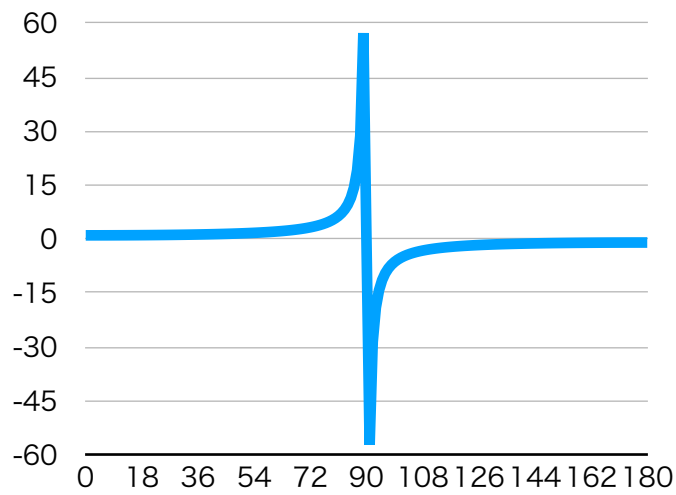
例:

SA=1  
SG=2^(1/2)  
SY=2  
SM=-1  
SQ=-2  
SS=-2^(1/2)



$x=n/2;$   
 $\text{if}(x==90)\{y=-1;\}\text{else}\{y=1/\cos(x);\}$

— yの値



横軸=x, 縦軸=y

$y = \frac{1}{\cos(x)}$ , もし  $x = 90$  度の場合は  $y = -1$

#### 書式4

**S**<アラビア数字列>

アラビア数字列が10進数の小数点の値として解釈され-1が掛け合わされる。

この後に単純係数が続く場合には、単純係数の書式6が適用される。

#### 例

1S0=-1.0m

1S23=-1.23m

3S14=-3.14m

1S0K=-1.0km

1S23K=-1.23km

#### 書式5

**S**

-1を示す特殊係数として扱われる。

特殊係数書式4の後続を省略した形式である。

後続が無い場合のみ使用する事ができる。

#### 例

1S=-1m

3S=-3m

1KS=-1km

3KS=-3km

### 4.3.3 演算子

割り算演算子はアルファベットのXで表すものとする。

足し算演算子はアルファベットのPで表すものとする。

演算子の優先順位と結合規則:

- (1) 係数同士の掛け算
- (2) 割り算
- (3) 足し算

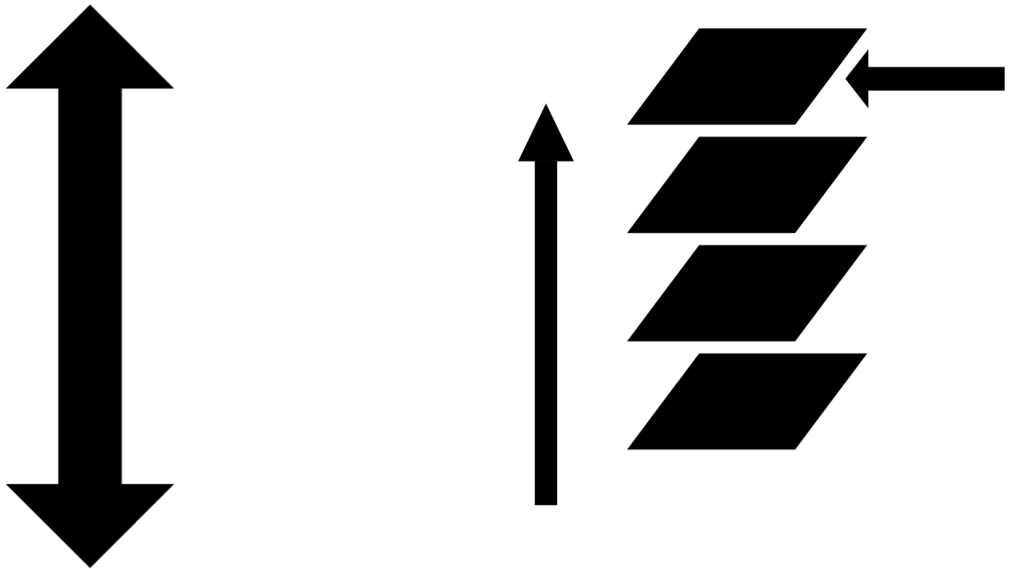
上記の優先順位で左から右に結合されるものとして解釈されるものとする。

後続に対する影響:

演算子の直後に演算子を置く事はできないものとする。

また演算子で終了してはならない。必ずその他の単  
純係数や演算子を除いた特殊係数やスタック定数が入るものとする。

- \* 係数は並べるだけで掛け合わされる為、掛け算専用の演算子は用意していない。
- \* 引き算用の演算子は用意していないが、負の値を足す事で代用できる。
- \* 演算子をアルファベットで表す事にした理由は、QRコードの英数字モードによる符号化で、利用可能な文字の種類が限られている都合上、プラス、マイナス、アスタリスク、スラッシュやピリオドを、別の目的で流用できるようにする為である。



#### 6.3.3.4 <長さのスタック定数>

<長さのパラメータ>は、現れた順に、リスト構造に一度保存され、重複を取り除かれた上で、長さのスタックに積まれ行く。スタック変数は、スタックに積まれたパラメータの値を定数として利用する為の仕組みである。

<長さのスタック定数>では、長さ以外のパラメータは、フィルタリングされるスタックには入らない仕組みとなっている。

パラメータを保存するスタックは、2種類用意されている。

<長さのスタック>は、次の2種類が用意されている。

<3文字以上のデータ長スタック>

<4文字以上のデータ長スタック>

<3文字以上のデータ長スタック>には、3文字以上のアルファベットで指定された長さを保存する。参照可能な個数は最後の値から、最大で7個までとなる。

<4文字以上のデータ長スタック>には、4文字以上のアルファベットで指定された長さを保存する。この長さスタックに保存される値の個数には制限を持たない。

4文字以上の<長さのパラメータ>が利用された場合には、<3文字以上のデータ長スタック>と<4文字以上のデータ長スタック>の両方に、値が保存されるものとする。

どちらのスタックも、初期値として、<1M20>という値で満たされている(この値は1.20mと解釈される)。



<長さのスタック>に、<長さのパラメータ>が保存される迄の流れ:

1つの<長さのパラメータ>の中で、<長さのパラメータ>の値を複数回呼び出す事ができる。

<長さのパラメータ>の中で、呼び出された場合は、呼び出された順番でリスト構造に一旦保存され、<長さのパラメータ>の終わった時点で重複を取り除いた上でそれぞれの<長さのスタック>で保存される。

この動作を詳しい手順は次の通りである。

- (1) リストに最後から最初に向かって値が重複するデータが存在するかどうかを確認する。
- (2) 重複しているデータが存在する場合には、一番最後の順番にあるデータだけが残され、重複は削除される。
- (3) <長さのパラメータ>の終わった時点で、リストに入れられた順番を守って、文字数に従って<3文字以上のデータ長スタック>と<4文字以上のデータ長スタック>に保存される。

<長さのスタック定数>の特徴:

<長さのスタック定数>には、下記の4つの形式が含まれている。

<長さのパラメータ先頭用短縮形式>

<長さのパラメータ先頭用通常形式>

<長さのパラメータ先頭用特殊形式>

<長さのパラメータ中間用形式>

この内、下記の3つの形式は、<長さのパラメータ>の先頭で、一度だけ利用できる。

<長さのパラメータ先頭用短縮形式>

<長さのパラメータ先頭用通常形式>

<長さのパラメータ先頭用特殊形式>

これに対し、<長さのパラメータ中間用形式>は、<長さのパラメータ>の中で、何度でも使用できるが、<長さのパラメータ>は、必ずアラビア数字で始まるという規則があるため、先頭に置く事ができない。

<長さのスタック定数>後続の書式に対する影響:

直後にアルファベットの文字で指定する<演算子>を置くことはできない。アラビア数字で始まる<単純係数>が後ろに続く場合は、<単純係数>の書式1から書式5の何れかが適用されるものとする。

<長さのパラメータ先頭用短縮形式>の書式の特徴:

<長さのパラメータ>の先頭で、一度だけ利用できる。

アラビア数字の1文字とパーセント(%)の組み合わせで10通りの書式を持っている。

全て、<3文字以上のデータ長スタック>を利用する。

書式1~7までで、6つ前、前から7番目迄(基数=0)の値を参照する事ができる。書式8~10は、最後の値に(-1, 0.5, 2の何れかの)定数を掛けた値を返す。

<長さのパラメータ先頭用通常形式>の書式の特徴:

<長さのパラメータ>の先頭で、一度だけ利用できる。

アラビア数字の1文字(0~3)とパーセント(%)の組み合わせで4通りの書式を持っている。短縮形式とは異なり、<4文字以上のデータ長スタック>に入った値を参照するために、24進数を使用する<アルファベット>の文字が後ろに続く書式となっている。

<4文字以上のデータ長スタック>のスタック定数を、24進数でアルファベットを利用してスタック上で参照したいデータの位置を指定するものとする。アルファベットと24進数との対応は、下記の表に従うものとする。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

アルファベットAは、数値0に相当し、<4文字以上のデータ長スタック>に、最後に入れられた値を参照することができる。

<長さのパラメータ先頭用特殊形式>の書式の特徴:

<長さのパラメータ>の先頭で、一度だけ利用できる。

アラビア数字の1文字(4~9)とパーセント(%)の組み合わせで6通りの書式を持っている。他の形式とは異なり後続の<角度のパラメータ>に対応する三角関数の値を返す特殊なものとなっている。

<長さのパラメータ中間用形式>の書式の特徴:

<長さのパラメータ>の途中で、何度でも利用できる。

書式1では、最後に<3文字以上のデータ長スタック>に入った値を参照するものとする。

書式2では、<4文字以上のデータ長スタック>のスタック定数を、24進数でアルファベットを利用してスタック上で参照したいデータの位置を指定する。

<長さのパラメータ先頭用短縮形式>の書式:

書式1

**0%**

最後に<3文字以上のデータ長スタック>に入った値を参照する。

書式2

**1%**

最後に<3文字以上のデータ長スタック>に入った値から1つ前の値を参照する。

書式3

**2%**

最後に<3文字以上のデータ長スタック>に入った値から2つ前の値を参照する。

書式4

**3%**

最後に<3文字以上のデータ長スタック>に入った値から3つ前の値を参照する。

書式5

**4%**

最後に<3文字以上のデータ長スタック>に入った値から4つ前の値を参照する。

書式6

**5%**

最後に<3文字以上のデータ長スタック>に入った値から5つ前の値を参照する。

書式7

**6%**

最後に<3文字以上のデータ長スタック>に入った値から6つ前の値を参照する。

書式8

**7%**

最後に<3文字以上のデータ長スタック>に入った値の符号を反転した値が得られる。

書式9

**8%**

最後に<3文字以上のデータ長スタック>に入った値を0.5倍した値が得られる。

書式10

**9%**

最後に<3文字以上のデータ長スタック>に入った値を2倍した値が得られる。

<長さのパラメータ先頭用通常形式>の書式:

書式1

**0%**<アルファベット>

<4文字以上のデータ長スタック>に入った値を参照するものとする。4文字以上の長さの定数を参照し、この値の符号を逆にした値が得られるものとする。

書式2

**1%**<アルファベット>

<4文字以上のデータ長スタック>に入った値を参照するものとする。

書式3

**2%**<アルファベット>

<4文字以上のデータ長スタック>に入った値を参照し2倍した値が得られるものとする。

書式3

**3%**<アルファベット>

<4文字以上のデータ長スタック>に入った値を参照し0.5倍した値が得られるものとする。

<長さのパラメータ先頭用特殊形式>の書式:

書式1

**4%**<角度のパラメータ>

**sin**(<角度のパラメータ>)の式で得られた値を返す。

書式2

**5%**<角度のパラメータ>

**abs(sin**(<角度のパラメータ>))の式で得られた値を返す。

書式3

**6%**<角度のパラメータ>

**cos**(<角度のパラメータ>)の式で得られた値を返す。

書式4

**7%**<角度のパラメータ>

**tan**(<角度のパラメータ>)の式で得られた値を返す。

書式6

**8%**<角度のパラメータ>

<角度のパラメータ>が90度と270度の値を示す場合は、**-1**の値を返し、それ以外場合は、下記の式で求めた値を返す。

**1/cos**(<角度のパラメータ>)

※ 特殊係数Sの、<角度のパラメータ>の値をとば計算式が異なり、そのままの角度が適用される。



## 書式7

**9%**<角度のパラメータ>

<角度のパラメータ>が90度と270度の値を示す場合は、**1**の値を返し、それ以外場合は、下記の式で求めた値を返す。

**abs(1/cos(<角度のパラメータ>))**

<長さのパラメータ中間用形式>の書式:

書式1

**%**

最後に<3文字以上のデータ長スタック>に入った値を参照するものとする。

書式2

**%<アルファベット>**

<4文字以上のデータ長スタック>に入った値を参照するものとする。4文字以上の長さの定数を参照する事ができる。

アルファベット:

<4文字以上のデータ長スタック>のスタック定数では、24進数でアルファベットを利用してスタック上で参照したいデータの位置を指定するものとする。アルファベットと24進数との対応は、下記の表に従うものとする。

<b>数値</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>
<b>文字</b>	A	B	C	D	E	F	G	H	Y	J	K	L
<b>数値</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>16</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>	<b>21</b>	<b>22</b>	<b>23</b>
<b>文字</b>	M	N	Z	P	Q	R	S	T	U	V	W	X

アルファベットAは、数値0に相当し、<4文字以上のデータ長スタック>に、最後に入れられた値を参照することができる。

### 6.3.5 <長さのパラメータ>の特徴

始まりの文字:

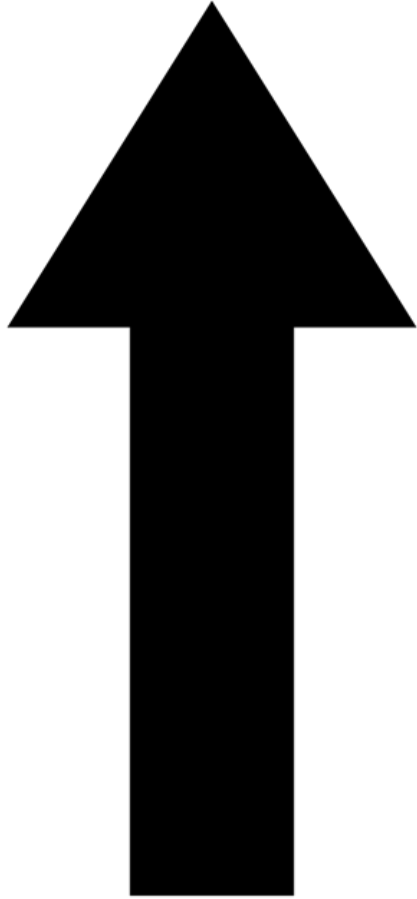
制約条件: 必ずアラビア数字で始まる。

終わりの文字:

アルファベット、数字、パーセントマーク(%)の何れかで終わる。

※<角度のパラメータ>に続いて、<長さのパラメータ>が指定される場合は、角度はアルファベットで符号化されており、<長さのパラメータ>が、必ずアラビア数字で始まるという規則がある為、区切り文字を間に挟む必要は無い。

※<長さのパラメータ>の後に、<長さのパラメータ>や別のパラメータが続く場合には間に、アルファベット、アラビア数字、パーセント(%)以外の文字を区切り文字として入れる事が必要となる。



#### 6.4 高さ、距離の指定

高さ、距離を指定するパラメータは、長さの指定で定義した、<長さのパラメータ>そのものである。

通常は、長さの値は、0から正の値範囲の値しか利用しないが、高さは負の値も利用できる(\*)。

従って、高さや、距離のは、<長さのパラメータ>とし扱われるものとする。

例

$$0 = 0\text{m}$$

$$1\text{M}0 = 1.0\text{m}$$

$$1\text{S}0 = -1.0\text{m}$$

$$1\text{M}23 = 1.23\text{m}$$

$$1\text{S}23 = -1.23\text{m}$$

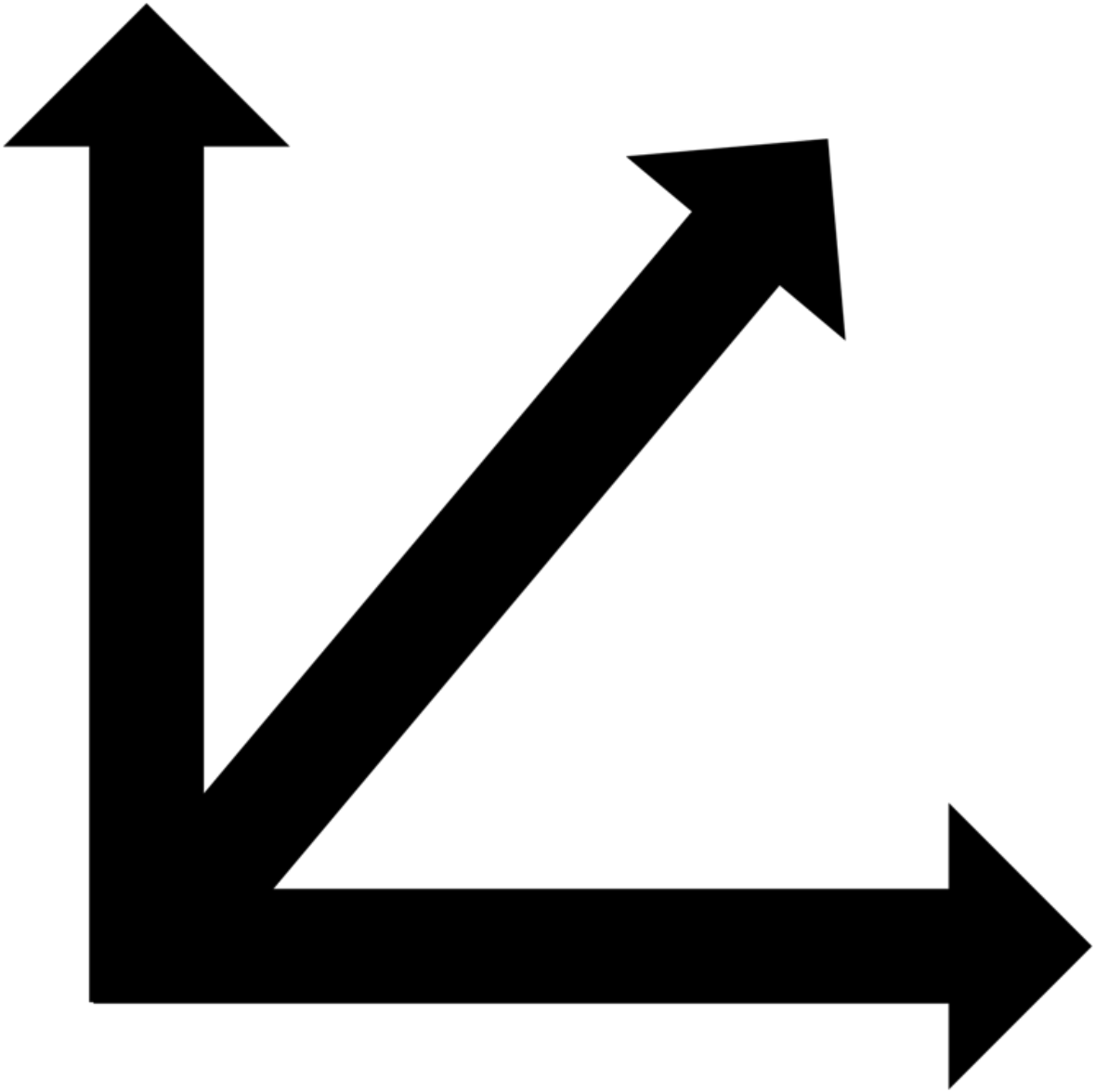
$$1\text{K} = 1\text{km}$$

$$1\text{K}\text{S} = -1\text{km}$$

$$1\text{K}23 = 1.23\text{km}$$

$$1\text{S}23\text{K} = -1.23\text{km}$$

\*長さの指定の中で定義された長さの値は、もともと負の値も定義できる様に、設計されている。



## LPSとGMSの<座標系>の定義:

QRコードに、対し同じ情報を出来る限り少ない文字数で、表現できれば、同じ面積に印刷する型番(バージョン)が小さくなるため、QRコードを構成する各ユニットを大きくする事が可能となり、より遠くから認識できるようになる。そのために、座標系もより短く表現できる工夫を盛り込む事とする。道案内やガイドにまず必要とされるものは、どちらに進めば良いかを示す方向の情報である。特にすぐ側にある場合は、距離の情報は不要の場合もある。そこで、まずは、方向、そして、必要に応じ、距離を加える事ができる様に<座標系>を設計した。

LPSおよびGMSにおける、表現の基本的な考え方は、自分の現在位置を基準点として、周りに何が存在するのかという考え方である。

3次元で、座標を表すには、 $x, y, z$ の3軸に対し原点からの距離で指定する<直交座標系>に対し、球面に対し、球の半径と横方向の角度、縦方向の仰角で表現する<極座標系>(球面座標系)、円柱に対し、円柱の半径と縦方向の高さで表現する<円筒座標系>(円柱座標系)が考えられる。

LPSおよびGMSでは、方向だけを示す事により情報の削減を行いたいという用途があり得る為、直交座標系による表現は、避けるべきである。これに対し、<円柱座標系>は、対象の高さのデータが揃っていれば、値を求め易い。

球面座標系で有れば、方向さえ示す事が出来れば良いという場面では、距離情報を省いても、情報として成立するというメリットが存在する。

情報量の削減の点から考えた場合は、<極座標系>にメリットがあるため、この座標系だけを定義する方法も考えられるが、<極座標系>は、高さ方向の値

が重要な場合は、誤差が大きくなる性質を持つ為、高さ方向の誤差を少なく表現可能な、<円筒座標系>も利用することができるものとする。

また、平面上の座標の指定には、<極座標系>から仰角の情報を省くか、<円筒座標系>から高さ方向の情報を省く事で得られる<円座標系>を利用する事とする。

特に、LPSでは、<円座標>で十分な情報を表現できる場合が多い為、基本は、円座標系を用いるものとするが、必要に応じて、<極座標系>や、<円筒座標系>、<直交座標系>と<平面直交座標系>も利用できるものとする。

表現したい対象に要求される精度、目的、必要となるデータの量とのバランスを見て、この中から、選ぶものとする。

<座標系>は、下記の6つの座標系を定義する。

<直行座標系>

<平面直交座標系>

<直線座標系>

<極座標系>

<円筒座標系>

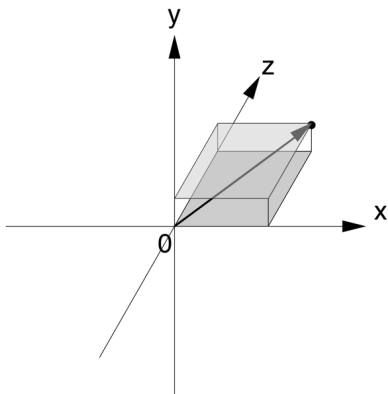
<円座標系>

上記の<座標系>は、LPSおよびGMSでは、混雑ができるように設計されている。<座標系>のパラメータでは、上記の<座標系>から、状況に応じて、適切なものを選ぶものとする。

各<座標系>に対する詳細を以降のページに示す。



<直交座標系>:



x軸: 横方向

0: 原点である事を示す

正の値: 原点に対し右側にある事を示す。

負の値: 原点に対し左側にある事を示す。

y軸: 高さ方向

0: 原点である事を示す

正の値: 原点に対し上にある事を示す。

負の値: 原点に対し下にある事を示す。

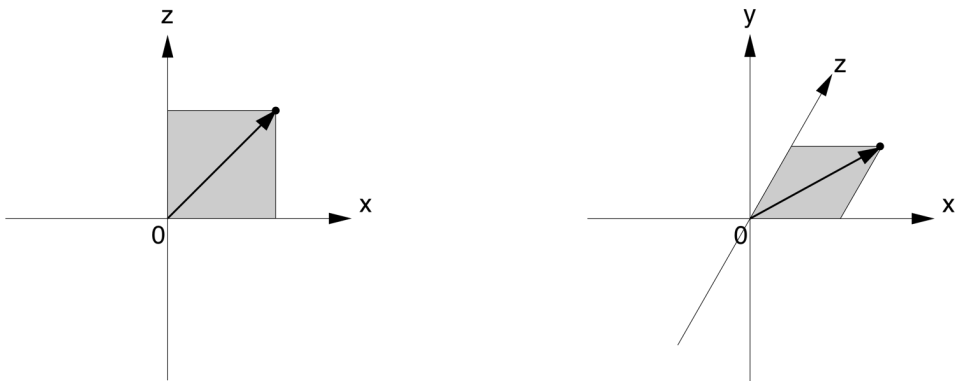
z軸: 奥行き

0: 原点である事を示す

正の値: 原点に対し前方にある事を示す。

負の値: 原点に対し後方にある事を示す。

<平面直交座標系>:



<平面直交座標系>は、<直交座標系>から、高さ方向を示すy軸の情報を省いた特別な形式である。

x軸: 横方向

0: 原点である事を示す

正の値: 原点に対し右側にある事を示す。

負の値: 原点に対し左側にある事を示す。

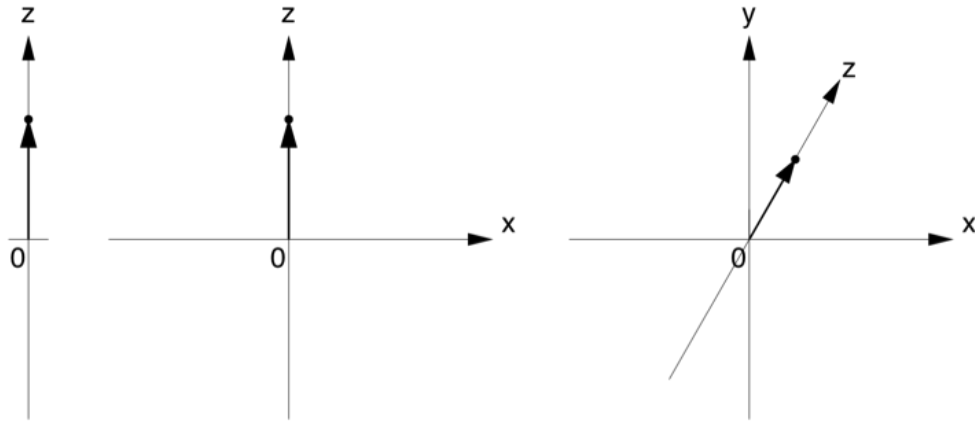
z軸: 奥行き

0: 原点である事を示す

正の値: 原点に対し前方にある事を示す。

負の値: 原点に対し後方にある事を示す。

<直線座標系>:



<直線座標系>は、<直交座標系>から、高さ方向を示すy軸の情報と横方向のx軸を省いた特別な形式である。

平面直交座標系から、横方向のx軸を省いた特別な形式とも言える。

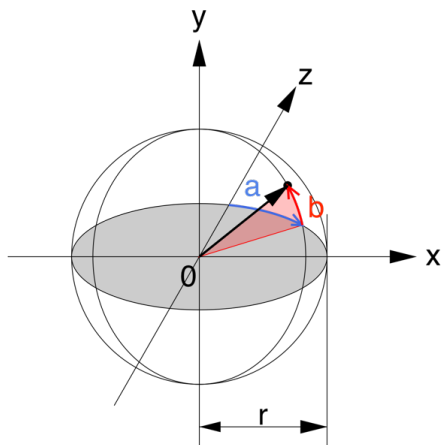
z軸: 奥行き

0: 原点である事を示す

正の値: 原点に対し前方にある事を示す。

負の値: 原点に対し後方にある事を示す。

<極座標系> (球面座標系):



a: 方向

y軸から見た時計回りの方向で表される角度

0: 正面にある事を示す。

90: 正面に対し右側にある事を示す。

180: 正面に対し後ろ側にある事を示す。

270: 正面に対し、左側にある事を示す。

角度の数值は、アルファベット1文字から2文字にコード化して表す事とする。

b: 仰角

0: 原点である事を示す。

90: 真上を向いている事を示す。

270: 真下を向いている事を示す。

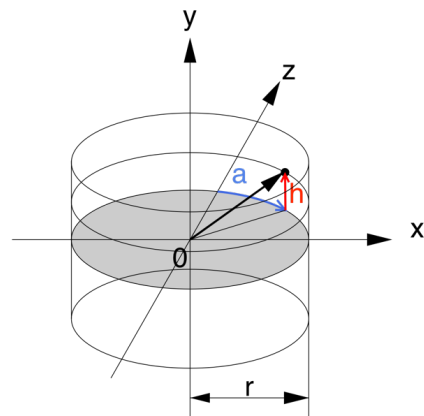
角度の数值は、アルファベット1文字から2文字にコード化して表す事とする。

r: 距離

この項目は必要に応じて省く事ができる。

0: 原点である事を示す。

<円筒座標系> (円柱座標系):



a: 方向

y軸から見た時計回りの方向で表される角度

0: 正面にある事を示す。

90: 正面に対し右側にある事を示す。

180: 正面に対し後ろ側にある事を示す。

270: 正面に対し、左側にある事を示す。

角度の数値は、アルファベット1文字から2文字にコード化して表す事とする。

r: 距離

0: 原点である事を示す。

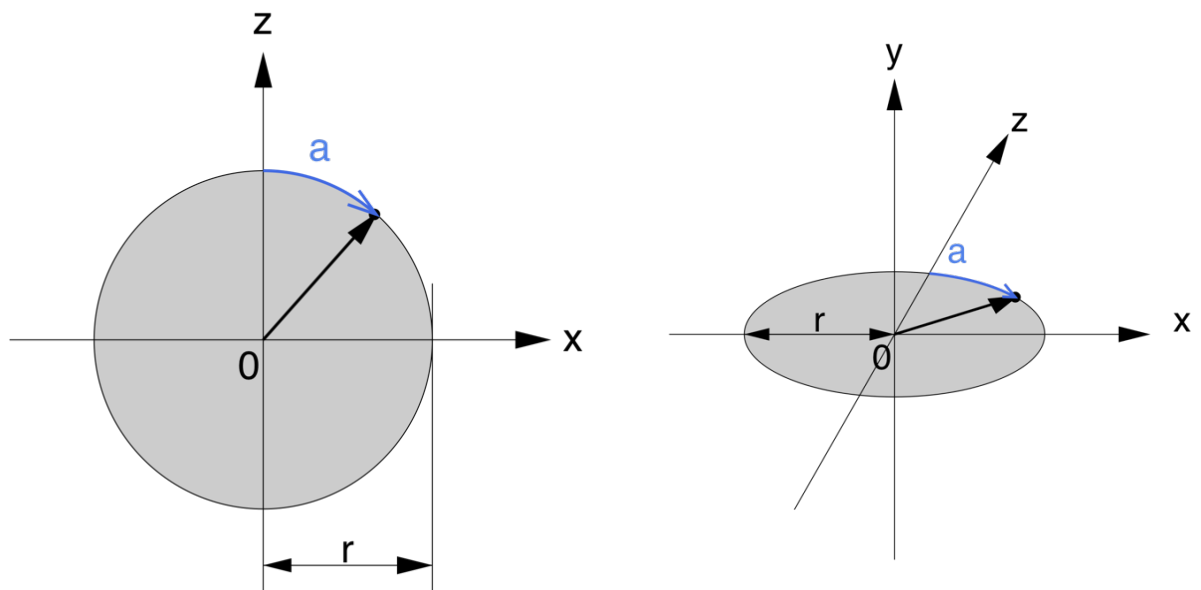
b: 高さ方向

0: 原点である事を示す

正の値: 原点に対し上にある事を示す。

負の値: 原点に対し下にある事を示す。

<円座標系>:



平面を表す際には、上から見下ろした際の方角、基点からの直線距離で、表現するものとする。これは極座標系では、仰角を省いた形式、円筒座標系では、高さの項目を省いた場合の特別な形式としても考える事ができる。

a: y軸から見た時計回りの方向で表される角度

0: 正面にある事を示す。

90: 正面に対し右側にある事を示す。

180: 正面に対し後ろ側にある事を示す。

270: 正面に対し、左側にある事を示す。

角度の数值は、アルファベット1文字から2文字にコード化して表す事とする。

r: 距離

0: 原点である事を示す。

LPSおよびGMSの<座標系>の書式:

<座標系>は、下記の書式を利用するものとする。

<直行座標系>:

<奥行き方向の位置>\*<横方向の位置>\*<高さ方向の位置>

<平面直交座標系>:

<奥行き方向の位置>\*<横方向の位置>

<直線座標系>:

<奥行き方向の位置>

<極座標系>:

<方向>\*<仰角><距離>

<円筒座標系>:

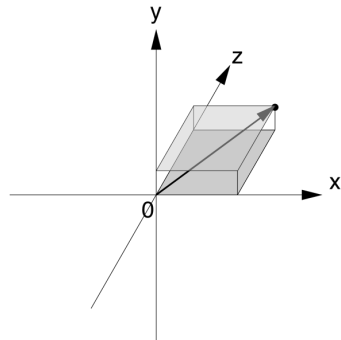
<方向><距離>\*<高さ>

<円座標系>:

<方向><距離>

各<座標系>の書式の詳細を以降のページに示す。

<直行座標系>:



<直行座標系>の書式:

<奥行き方向の位置>\*<横方向の位置>\*<高さ方向の位置>

<直行座標系>のパラメータ:

<横方向の位置>:

長さのパラメータである。

x軸方向の距離である。

<奥行き方向の位置>:

長さのパラメータである。

z軸の距離である。

<高さ方向の位置>:

長さのパラメータである。

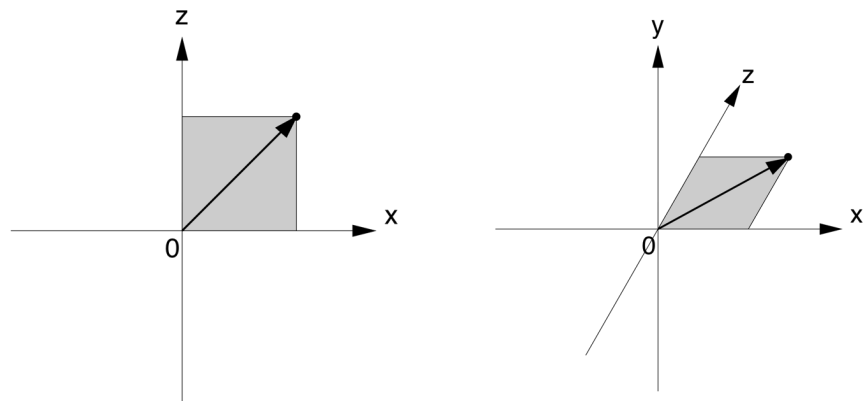
y軸の距離である。

x-z平面上のy軸に対する方向を求める必要がある場合:

LPSでは、基本的に、x-z平面上に投射されるて処理される場合は、横方向の位置と奥行き方向の位置のベクトルから求めるものとする。ベクトルが0の場合は、前の方向が維持される。



<平面直行座標系>:



<平面直交座標系>は、<直交座標系>から高さを省いた特別な形式である。

<平面直行座標系>の書式:

<奥行き方向の位置>\*<横方向の位置>

<平面直行座標系>のパラメータ:

<横方向の位置>:

長さのパラメータである。

x軸方向の距離である。

<奥行き方向の位置>:

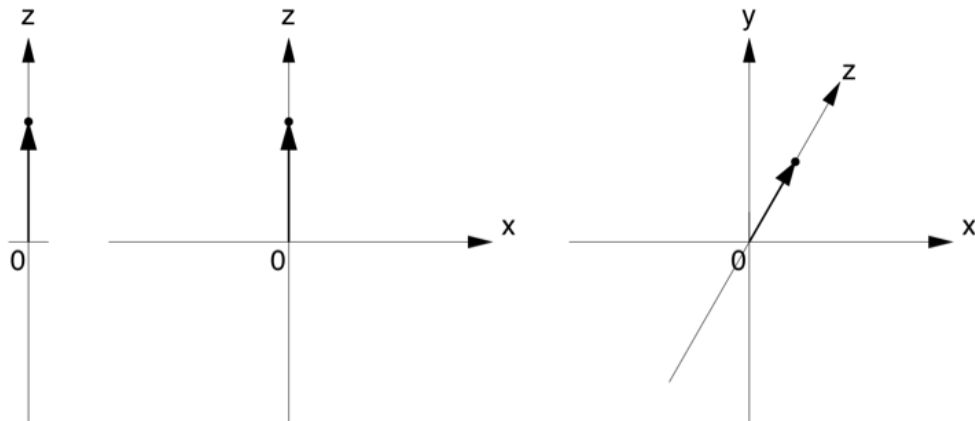
長さのパラメータである。

z軸の距離である。

x-z平面上のy軸に対する方向を求める必要がある場合:

LPSでは、基本的に、x-z平面上に投影されるて処理される場合は、横方向の位置と奥行き方向の位置のベクトルから求めるものとする。ベクトルが0の場合は、前の方向が維持される。

<直線座標系>:



<直線座標系>は、<平面直交座標系>から横幅を省いた特別な形式である。

<直線座標系>の書式:

<奥行き方向の位置>

<直線座標系>のパラメータ:

<奥行き方向の位置>:

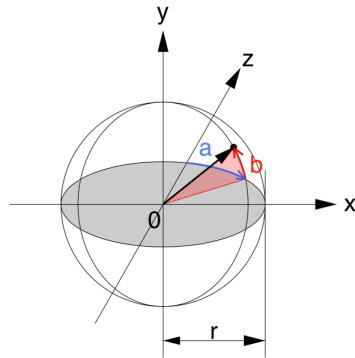
長さのパラメータである。

z軸の距離である。

x-z平面上のy軸に対する方向を求める必要がある場合:

LPSでは、基本的に、x-z平面上に投射されるて処理される場合は、<奥行き方向の位置>の値が正または0であるならば、前の方向が維持される。負の値の場合は、前と180度反転した方向となる。

<極座標系>:



<極座標系>の書式:

<方向>\*<仰角><距離>

<極座標系>のパラメータ:

<方向>:

角度のパラメータである。

y軸回りの角度を示す。

<仰角>:

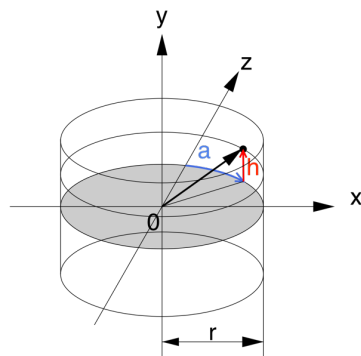
角度のパラメータである。

x-z平面に対する角度を示す。

<距離>:

長さのパラメータである。

<円筒座標系>:



<円筒座標系>の書式:

<方向><距離>\*<高さ>

<円筒座標系>のパラメータ:

<方向>:

角度のパラメータである。

y軸回りの角度を示す。

<距離>:

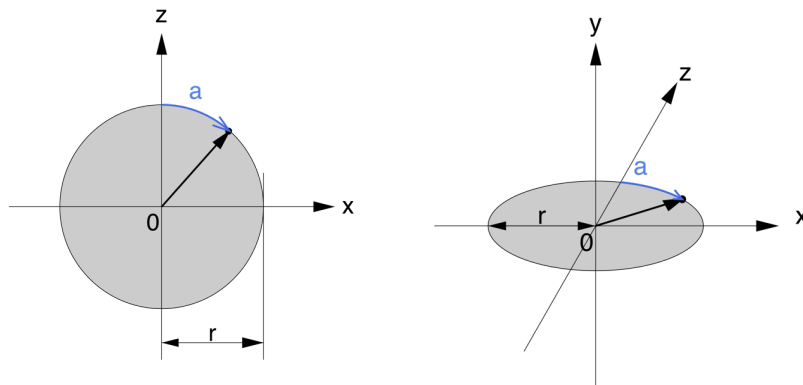
長さのパラメータである。

<高さ>:

長さのパラメータである。

y軸の距離である。

<円座標系>:



<円座標系>の書式:

<方向><距離>

<円座標系>のパラメータ:

<方向>:

角度のパラメータである。

y軸回り(左ネジ)の角度を示す。

<距離>:

長さのパラメータである。

x-z平面上のy軸の<角度のパラメータ>は以下のように求める場合:

<直行座標系>と<平面直交座標系>の場合:

x-z平面上のy軸の<角度のパラメータ> =

0の位置から<横方向の位置>と<奥行き方向>のx-z平面上に投射した場合の<角度のパラメータ>を求める。

ベクトルが0となる場合は、前のy軸の<角度のパラメータ>が適用される。

<直線座標系>の場合:

x-z平面上のy軸の<角度のパラメータ> =

<奥行き方向の位置>の値が正または0の場合:

前のy軸の<角度のパラメータ>が適用される。

<奥行き方向の位置>の値が負の場合:

前のy軸の<角度のパラメータ>を180度反転した角度が適用される。

<極座標系>と<円筒座標系>と<円座標系>の場合:

x-z平面上のy軸の<角度のパラメータ> = <方向>となる。

x-z平面上の<長さのパラメータ>は以下の様に求める:

<直行座標系>と<平面直交座標系>の場合:

$$\begin{aligned} & \text{x-z平面上の<長さのパラメータ>} \\ & = \mathbf{sqrt}(\text{<奥行き方向の位置>}^2 + \text{<横方向の位置>}^2) \end{aligned}$$

<直線座標系>の場合:

$$\text{x-z平面上の<長さのパラメータ>} = \mathbf{abs}(\text{<奥行き方向の位置>})$$

<極座標系>の場合:

$$\text{x-z平面上の<長さのパラメータ>} = \text{<距離>} * \mathbf{abs}(\mathbf{cos}(\text{<仰角>}))$$

<円筒座標系>と<円座標系>の場合:

$$\text{x-z平面上の<長さのパラメータ>} = \text{<距離>}$$

LPSおよびGMSの座標系の書式のパラメータの種類から見たまとめ:

<直行座標系>:

<奥行き方向の位置>\*<横方向の位置>\*<高さ方向の位置>

<平面直交座標系>:

<奥行き方向の位置>\*<横方向の位置>

<直線座標系>:

<奥行き方向の位置>

<極座標系>:

<方向>\*<仰角><距離>

<円筒座標系>:

<方向><距離>\*<高さ>

<円座標系>:

<方向><距離>

角度と長さの順番でパラメータが並ぶ場合には、<角度のパラメータ>は終端がアルファベットまたは、角度のスタック定数を示すパーセントマークであるのに対し、<長さのパラメータ>は、先頭が、必ずアラビア数字から始まるという規則になっているため、<角度のパラメータ>と<長さのパラメータ>の間には区切り文字は不である。

この仕組みから、LPS規格の中で、例を示す場合には、主に<円座標系>を用いている。



これは、<円座標系>が、パラメータ間の区切り文字を本規格では必要としないので、最も汎用的で少ないデータサイズで必要な精度をこの<座標系>で得られると考えられるためである。繰り返しになるが、要求される精度を満たし、必要とするデータ量を最小限にできる<座標系>を状況に合わせて適切に選択するべきである。

<直線座標系>は、正負の値で、前の角度を前後の方向しか選ぶことしかできないが、一番少ないデータ量で表現できる。従って前の座標からの相対で、前後の座標を指定したい場合には最も相応しい<座標系>である。

角度のパラメータに利用されている文字の種類:

アルファベットとパーセント(%)マークのみ。

先頭に使われている文字:

アルファベットとパーセント(%)マークのみ。

終端に使われている文字:

アルファベットとパーセント(%)マークのみ。

長さのパラメータに利用されている文字の種類:

アラビア数字とアルファベットに、パーセント(%)マークのみ。

先頭に使われている文字:

アラビア数字のみ。

終端に使われている文字:

アルファベットとパーセント(%)マークのみ。

角度のパラメータと長さのパラメータでは、先頭に利用されている文字の種類が異なる為、どちらのパラメータか判断する事ができる。

座標のパラメータに利用されている文字の種類:

アラビア数字とアルファベットに、パーセント(%)、アスタリスク(\*)  
マークのみ。

従って上記以外の文字が区切り文字として利用できる。

A ↔ a

## 7. 文字列の指定

<文字列のパラメータ>ではQRコードの英数字モードを標準としているため、アラビア数字、アルファベットの大文字と、空白については自由に利用することができるが、それ以外の文字を利用する場合には、エスケープ文字等を利用しなければならない場合がある。

### LPS及びGMS共通のエスケープ文字

LPS及びGMSでは、エスケープ文字を文法上の制約と、英数字モードで扱う事ができる文字の種類が限られているという符号化上の制約という2つの理由で利用する事となる。

#### 7.1 文法上の制約から必要なエスケープ文字

どのプロパティの中でも常に適用されるエスケープ文字の記述方法:

データの中で、文字列の一部として「/」、「:」、「.」、「%」、「\*」を利用する場合には、LPSやGMSの文法の制約上のエスケープ文字として記述する必要がある。エスケープ文字の表記法を下記の表に示す。

プロパティ共通のエスケープ文字の表記法

表記したい文字	英語名	実際の記述
/	Slash	//
:	Colon	/:
.	Period	/.
%	Per cento	/%
*	Asterisk	/*

※記号文字の読み間違いを防ぐため、英語名も表に併記した。

一部のプロパティの中だけで適用されるエスケープ文字の記述方法:

特定のプロパティに含まれるデータの中で、区切り文字や、種別を示す特殊な文字の意味合で「+」、「-」、「\$」が利用されている場合には、文字列の一部としてこの文字を利用する場合には、LPSやGMSの文法の制約上のエスケープ文字として記述する必要がある。エスケープ文字の表記法を下記の表に示す。

特定のプロパティの中のエスケープ文字の表記法

表記したい文字	英語名	実際の記述
+	Plus	/F
-	Hyphen-minus	/H
\$	Period	/\$

## 7.2 英数字モードの符号化の範囲を超える文字を扱うためのエスケープ文字

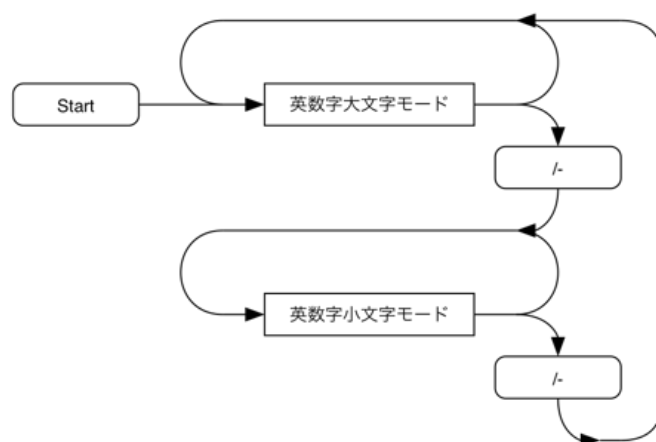
英数字モードの符号化を利用し、他のモードに切り替える事なく、英数字範囲外のUS-ASCIIに含まれる制御文字以外の文字とローマ字の長音文字を扱う方法を定義する。この規定は、あくまでデータの量を減らす為の手段であり、英数字以外のモードを利用した符号化の使用を禁止するものではない。LPSやGMSでは、英数字モードから始める事とするが必要に応じて、モードを切り替える事が許される。

### 7.2.1 英文字の小文字を扱うためのCaps lock制御エスケープ文字

英数字モードの符号化では、大文字だけしか扱う事ができない為、小文字を扱う為に、英文字の大文字、小文字の切り替えを行うCaps Lock制御の為のエスケープ文字である<caps切換>を定義する。

モードの指定	英語名	実際の記述	効果
<caps切換>	<caps-change>	/-	英数字の小文字/大文字の切り替え

このエスケープ文字は、後続の英字に対し、大文字として解釈されるのか、小文字として解釈するのかを切り替える役割を持つものとする。切り替えの有効範囲は限定的なものとし、各プロパティやパラメータの内部に、留まるものとする(別のパラメータ、プロパティには影響を及ぼさない)。



例: 「T/-OKYO /-JAPAN」と記述した場合は、「Tokyo JAPAN」と解釈される。

文字列のスタック定数として保存された文字列は、<caps切換>の影響を受けない。<文字列のスタック定数>に入った文字列は、<文字列のスタック>に保存された際の<caps切換>の指定が反映される為、現在の<caps切換>の指定の影響を受けない。また、現在の<caps切換>の指定に対しても影響を及ぼさないものとする。

「Tokyo JAPAN」の文字列を、<文字列のスタック定数>として、呼び出した場合は、<caps切換>でどちらのモードになっているかに関わらず常に、「Tokyo JAPAN」となる。

QRコードの漢字モードや、バイナリモードでUTF-8、シフトJISの中に含まれる場合、これらのモードに含まれているアルファベットの文字列は、<caps切換>の影響を受けない。また、<caps切換>に影響を与えない。

- ※ Caps Lock制御の為のエスケープ文字は、小文字の英数字を持たない英数字モード符号化の為にだけ用意された規則であり、他の小文字の英数字が利用可能な文字の符号化体系には適用されない。
- ※ 英文中の小文字は、連続して現れる性質がある為、キーボードのCaps lock キーの振る舞いを真似た仕様とした。

「/--」を先頭に付ける事で、1文字だけcapsを切り替える事ができる。

モードの指定	英語名	実際の記述	効果
<1文字caps切換>	<one-character-caps-change>	/--	1文字だけ英数字の小文字/大文字の切り替え



## 7.2.2 ローマ字の長音記号の符号化

英数字モードの符号化や漢字モードの符号化では、ローマ字の長音記号を直接扱う事ができない為、長音記号を扱うためのエスケープ文字を以下のように定義する。

「 $\bar{A}$ 」は、「/A」と記述する。

「 $\bar{E}$ 」は、「/E」と記述する。

「 $\bar{I}$ 」は、「/I」と記述する。

「 $\bar{O}$ 」は、「/O」と記述する。

「 $\bar{U}$ 」は、「/U」と記述する。

長音記号は、Caps lock制御エスケープ文字の影響を受ける。

従って、「U/-PPER CASE/- /A/E/I/O/U L/-OWER CASE /A/E/I/O/U」

と記述した場合は、「Upper case  $\bar{A}\bar{E}\bar{I}\bar{O}\bar{U}$  Lower case  $\bar{a}\bar{e}\bar{i}\bar{o}\bar{u}$ 」と解釈される。そ

して、「T/-/OKY/O /-JAPAN」と記述した場合は、「Tōkyō JAPAN」と解釈される。

また、英数字モード以外の小文字の英文字を利用できる他の漢字モードなどの文字コードの中で、長音文字を直接記述できない場合は、下記のように記述する事とする。

「/a」と記述した場合は、「 $\bar{a}$ 」と解釈され、

「/e」と記述した場合は、「 $\bar{e}$ 」と解釈され、

「/i」と記述した場合は、「 $\bar{i}$ 」と解釈され、

「/ō」と記述した場合は、「 $\bar{o}$ 」と解釈され、

「/u」と記述した場合は、「 $\bar{u}$ 」と解釈される。

「T/oky/o」と記述した場合は、「Tōkyō」と解釈される。

※GMSでは、道案内などの為、ローマ字表記が必要になる場合があるため、LPSでも共通で利用できるように、長音表記の方法を定義する事とした。

### 7.2.3 英数字モードで扱えない記号の符号化

英数字モードの符号化ではその都合上、そのままでは、US-ASCIIの記号文字の範囲を網羅して扱う事ができない為、下記の表で示すようにエスケープ文字を利用して表記するものとする。

表記したい文字	英語名	実際の記述	表記したい文字	英語名	実際の記述
!	Exclamation mark	/X	?	Question mark	/Q
“	Double Quotation mark	/W	@	Commercial at	/T
#	Number sign	/N	[	Left square bracket	/4
&	Ampersand	/M	\	Backslash	/B
'	Apostrophe	/P	]	Right square bracket	/5
(	Left parenthesis	/2	^	Caret	/K
)	Right parenthesis	/3	_	Underline (low dash)	/D
,	Comma	/C	`	Grave accent	/R
;	Semicolon	/S	{	Left curly bracket	/6
<	Less-than sign	/L		Vertical bar	/V
=	Equal sign	/Y	}	Right curly bracket	/7
>	Greater-than sign	/G	~	Tilde	/Z

※ 記号文字の読み間違えを防ぐため、英語名も表に併記した。

※ 記号文字の読み間違えを防ぐため、英語名も表に併記した。

※ 英数字モード以外の文字コードで符号化する場合は、上記の表で示した規則は適用されないものとする。

※ LPSやGMSで共通に使用されるデータの書式で使用される文字は、QRコードの英数字符号化(Alphanumeric)の範囲で利用できる文字に限定する事により、少ないデータサイズ(2文字で11ビット)に収まるように設計されている。

※英数字符号化(Alphanumeric)で表現できる文字は、アラビア数字に、アルファベットの大文字と、利用できる記号は下記の9つとなり、制御文字は含まれている。

- (1) 空白「 」
- (2) ドル「\$」
- (3) パーセント「%」
- (4) アスタリスク「\*」
- (5) プラス「+」
- (6) マイナス「-」
- (7) ピリオド「.」
- (8) スラッシュ「/」
- (9) コロン「:」

※その他の文字を利用する為には、他の文字が利用できるモードに切り替える事が必要となるが、QRコードのモード切り替えを含んだ、エンコード方式は、同じモードの範囲の文字が連続した場合に、一番効率的に機能する設計となっており、頻繁なモード切り替えた場合には、モードインジケータと、文字数インジケータが挿入された分だけ、ビット数が増え、合計ビット数が増加する。英数字モードでは、1文字当たり5.5ビットであるため、たった1文字、英数字モードで表現できない文字を表現する為にモード切り替えを行うには、オーバーヘッドが大き過ぎて割に合わないため、エスケープ文字による代替え手段を提供する事とした。

QRコードの文字コード切り替えのオーバーヘッド

QRコードのバージョン	文字数インジケータ		モードインジケータ		合計
	バイナリ	英数字	バイナリ	英数字	
1~9	8	9	4	4	25
10~26	16	11	4	4	35
27~40	16	13	4	4	37

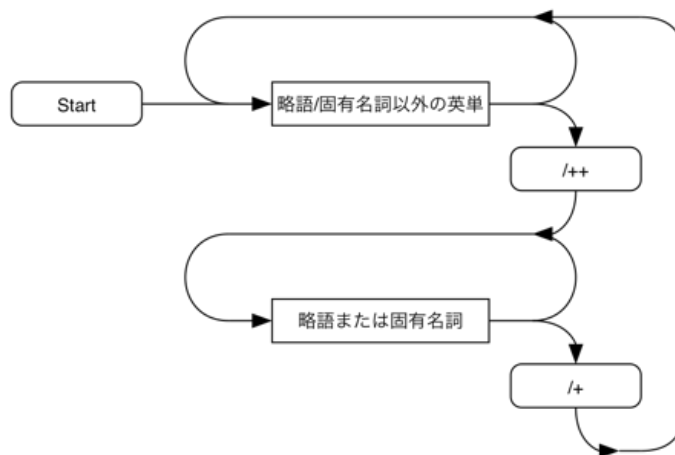
※勿論、同じ文字の種類が連続する場合には、モード切り替えのオーバーヘッドが許容範囲に収まるようになる筈である。従って、LPSやGMSのデータを作成する際には、できる限り、小さなQRコードのバージョンを選ぶ事ができるように、モードの切り替えと、エスケープ文字を利用した表記法の両方を活用するべきである。

固有名詞との切り替え:

モードの指定	英語名	実際の記述	効果
<連続した略語/固有名詞の指定>	<continues-proper-noun>	/++	連続した外来語や略語、固有名詞の指定の指定

「/++」から後の文字列は、英単語から始まる事を前提としており、<略語固有名詞切替>の為の、エスケープ文字を受けると、以降は、略語または固有名詞として解釈される。

切り替えの有効範囲は限定的なものとし、各プロパティやパラメータの内部に、留まるものとする(別のパラメータ、プロパティには影響を及ぼさない)。このエスケープシーケンスは、文字列のプロパティの中身を翻訳エンジンに通す際のヒント情報を提供する事を目的としている。英語では、文章の一番先頭を除き大文字から始まる単語は、固有名詞と解釈するルールが存在するが、装飾的に固有名詞以外も含めて、全て大文字のみで表記される場合があり得る。このエスケープ文字を定義する事により、小文字か、大文字かではなく、訳すと本来の意味から変わってしまう単語を、明示する事ができる様になる。



建物の名前や地名などは、その発音が意味を持つ。

「White House」を「白い家」と訳した場合、「アメリカ合衆国大統領官邸」の意味合いは無くなってしまう事になる。固有名詞として指定する事でこの様な問題が生じる事を防ぐ事ができる。

文字列として、表示した場合は、標準フォントから、ボール体、または、イタリック、体の指定のOn/Offに該当する機能である。

<文字列のスタック定数>に入った文字列は、<文字列のスタック>に保存された際の<固有名詞切替>の指定がそのまま反映される為、現在の、<固有名詞切替>の指定の影響は、受けない。また、<文字列のスタック定数>の参照は、現在の<固有名詞切替>の指定に影響を及ぼさ無いものとする。

- ※ 「実際には、White House」は有名なので、一般的な翻訳エンジンは、「ホワイトハウス」の様に訳し、「白い家」とは訳す事はまず無いと考えられるが、「red house」と表記し、携帯情報端末の言語指定が日本語の場合は、日本語で「赤い家」と訳されるであろう。もし、携帯情報端末が運用される場所が、日本語以外の言語圏の場合に、「赤い家」について教えてくださいと固有名詞を日本語訳された言葉で尋ねても伝わる事は無いと考えられる。固有名詞が特定できるのであれば、翻訳エンジンは、「red house」を例えば「レッドハウス(赤い家)」の様な方で、表す事が容易となると考えられる。

モードの指定	英語名	実際の記述	効果
<固有名詞>	<proper-noun>	/+	外来語や略語、固有名詞の指定の指定

「/+」から後の文字列は、英単語から始まる事を前提としており、<略語固有名詞切替>の為の、エスケープ文字を受けると、以降は、略語または、固有名詞として解釈される。

「/++」は、「/+」に到達するまで効力を保つが、「/+」は、アルファベットもしくは「-」以外の文字(空白も含む)が入ると、その後の文字からは効力を失われる事となる。切り替えの有効範囲は限定的なものとし、各プロパティやパラメータの内部に、留まるものとする(別のパラメータ、プロパティには影響を及ぼさない)

例えば、「ID」などの略語の場合は、この後に続く空白から先は、略語や固有名詞が使われる確率が低いので、この様な仕様とした。

固有名詞が連続2つ以下であれば、「/+」を利用し、連続3つ以上であれば、「/++」を利用した方が、短い符号長なる。

漢字について:

LPSでは、英語での表記を基本とし、漢字は使用しないものとする。

GMSでは、英語に加えて、2番目に、現地のローカル言語を、指定する事もできる。日本語で漢字を利用する場合には、QRコードの漢字モードを利用するものとする。QRコードの漢字モードは、オーバーヘッドの発生は存在するが、漢字1文字を13ビットでエンコードすることができるので漢字を表現する為に、最も効率が良いエンコーディングとなっている。

#### QRコードの文字コード切り替えのオーバーヘッド漢字の場合

QRコードのバージョン	文字数インジケータ		モードインジケータ		合計
	漢字	英数字	漢字	英数字	
1~9	8	9	4	4	25
10~26	10	11	4	4	29
27~40	12	13	4	4	33

また、バイナリモードを利用して、UTF-8でエンコードする事により、CJKスペースに置かれた漢字を利用する事もできるが、漢字モードよりも効率が劣る為、漢字を利用するのであれば、漢字モードの利用を推奨する。但し、漢字モードで収録されていない漢字をどうしても使用する必要がある場合には、UTF-8を使用するべきである。



UTF-8による表記について:

バイナリモードに切り替えられた場合には、UTF-8形式の文字列が利用されている事を意味する。

UTF-8を利用する為に、バイナリモードに切り替えの為に必要となるオーバーヘッドは大きいので使用には、注意を要する。

オーバーヘッドのビット数は、QRコードのビット数に応じて変化する。

バイナリモードでは、1バイト(1オクテット)につき8ビットのデータが必要となる。1バイトの範囲で表現できる文字は、US-ASCIIの範囲と一致し、ラテン文字の拡張の範囲を表現するには、2バイトが必要となる。

QRコードの文字コード切り替えのオーバーヘッドUTF-8

QRコードのバージョン	文字数インジケータ		モードインジケータ		合計
	バイナリ	英数字	バイナリ	英数字	
1~9	8	9	4	4	25
10~26	16	11	4	4	35
27~40	16	13	4	4	37

<UTFエンコード>について:

バイナリモードに切り替えてUTF-8でエンコードする場合のオーバーヘッドが大きい為、英数字モードの中で、Unicodeの文字コードを指定する為に、<UTFエンコード>を定義する事とする。

モードの指定	英語名	実際の記述	効果
<UTFエンコード>	<UTF-encode>	/8	<UTFのエンコード>によるエンコード開始

「/8」を<UTFエンコード>の開始を意味する。この後の文字は、テーブルに従ってUnicodeの文字コードを指す為に、変換表に従ってエンコードされる。<UTFエンコード>では、Unicodeの符号位置を指定する為の値を求める仕組みであり、「UTF-8」に従って直接エンコードするものではない。

QRコードの英数字モードでは、<QRコード英数字コード表>に従い、アラビア数字のゼロ「0」からコロン「:」までの文字に、0から44までの番号割り振られている。ここではこの番号を<#C>として

テーブルのインデックスに使用する。

<UTFエンコード>の最初文字はまず<Table-A>を利用して評価される。

Base変数の初期値は0x40となっており、特定の条件で更新される。

以降にテーブルの詳細を示す。

<QRコード英数字コード表> (1/2)

文字	<#C>	備考
0	0	アラビア数字
1	1	アラビア数字
2	2	アラビア数字
3	3	アラビア数字
4	4	アラビア数字
5	5	アラビア数字
6	6	アラビア数字
7	7	アラビア数字
8	8	アラビア数字
9	9	アラビア数字
A	10	アルファベット
B	11	アルファベット
C	12	アルファベット
D	13	アルファベット
E	14	アルファベット
F	15	アルファベット
G	16	アルファベット
H	17	アルファベット
I	18	アルファベット
J	19	アルファベット
K	20	アルファベット
L	21	アルファベット
M	22	アルファベット

<QRコード英数字コード表> (2/2)

文字	<#C>	備考
N	23	アルファベット
O	24	アルファベット
P	25	アルファベット
Q	26	アルファベット
R	27	アルファベット
S	28	アルファベット
T	29	アルファベット
U	30	アルファベット
V	31	アルファベット
W	32	アルファベット
X	33	アルファベット
Y	34	アルファベット
Z	35	アルファベット
Space	36	空白
\$	37	ドル
%	38	パーセント
*	39	アスタリスク
+	40	プラス
-	41	マイナス
.	42	ピリオド
/	43	スラッシュ
:	44	コロソ

UTF用&lt;Table-A&gt; (1/2)

<#C>	備考	Table-B参照回数
0	下位5ビットは0x00	0
1	下位5ビットは0x01	0
2	下位5ビットは0x02	0
3	下位5ビットは0x03	0
4	下位5ビットは0x04	0
5	下位5ビットは0x05	0
6	下位5ビットは0x06	0
7	下位5ビットは0x07	0
8	下位5ビットは0x08	0
9	下位5ビットは0x09	0
10	下位5ビットは0x0A	0
11	下位5ビットは0x0B	0
12	下位5ビットは0x0C	0
13	下位5ビットは0x0D	0
14	下位5ビットは0x0E	0
15	下位5ビットは0x0F	0
16	下位5ビットは0x10	0
17	下位5ビットは0x11	0
18	下位5ビットは0x12	0
19	下位5ビットは0x13	0
20	下位5ビットは0x14	0
21	下位5ビットは0x15	0
22	下位5ビットは0x16	0

UTF用&lt;Table-A&gt; (2/2)

<#C>	備考	Table-B参照回数
23	下位5ビットは0x17	0
24	下位5ビットは0x18	0
25	下位5ビットは0x19	0
26	下位5ビットは0x1A	0
27	下位5ビットは0x1B	0
28	下位5ビットは0x1C	0
29	下位5ビットは0x1D	0
30	下位5ビットは0x1E	0
31	下位5ビットは0x1F	0
32	Base=0x20	0
33	Base=0x40	0
34	Base=0x60	0
35	Table-B	2
36	Table-B	3
37	Table-B	4
38	Table-B	5
39	Table-B	6
40	Table-Cを参照	
41	最後から1つ前のBase更新履歴を参照	
42	<プロパティの終了>を意味する。	
43	<エスケープ>文字を意味する。	
44	使用しない	

UTF用<Table-A>に対応する処理:

Base変数の初期値は0x40となっており、特定の条件で更新される。

<#C>が0~31の場合には、Base変数の下位5ビットの値をマスクし、<Table-A>で示す値がOffsetとしてそのまま適用される。

$$U+((Base \& (\sim 0x1FLU)) + Offset);$$

Baseの値は変更しない。

<#C>が32の場合には、Base=0x20;

<#C>が33の場合には、Base=0x40;

<#C>が34の場合には、Base=0x60;

<#C>が35~39の場合には、<Table-B>を参照する。

Table-B参照回数分だけ、<Table-B>を参照し、Unicodeの符号位置を指定し、Baseの値を更新する。

Table-B<#C>が41を指す回数は含まないものとする。

<#C>が40の場合には、<Table-C>を参照する。

<Table-C>は、Baseの更新履歴を利用してUnicodeの符号位置を指定する。

Baseの値も更新される。

<#C>が41の場合には、Baseの更新履歴をもとに、Baseの値を最後から1つ前に戻す。

<#C>が42の場合には、UTFエンコードを終了と<プロパティ>の終了を意味する。

<#C>が43の場合には、<エスケープ文字>を意味する。

この後に8が続く場合には、UTFエンコードの終わりだけを意味し、それ以外の文字が続く場合には、UTFエンコードの終わりに加えてエスケープ文字の指定となる。

UTF用&lt;Table-B&gt; (1/2)

<#C>	ビット数	ビットパターン
0	5	00000
1	5	00001
2	5	00010
3	5	00011
4	5	00100
5	5	00101
6	5	00110
7	5	00111
8	5	01000
9	5	01001
10	5	01010
11	5	01011
12	5	01100
13	5	01101
14	5	01110
15	5	01111
16	5	10000
17	5	10001
18	5	10010
19	5	10011
20	5	10100
21	5	10101
22	5	10110

UTF用&lt;Table-B&gt; (2/2)

<#C>	ビット数	ビットパターン
23	5	10111
24	5	11000
25	5	11001
26	5	11010
27	5	11011
28	5	11100
29	5	11101
30	5	11110
31	5	11111
32	6	000000
33	6	100000
34	6	111111
35	7	0000000
36	7	1000000
37	7	1111111
38	8	00000000
39	8	10000000
40	8	11111111
41	Table-Bの呼び出し回数の固定(開始/終了)	
42	<プロパティの終了>を意味する。	
43	<エスケープ>文字を意味する。	
44	使用しない	

UTF用<Table-B>に対応する処理:

<Table-A>で指定された<Table-B>の参照回数分だけ連続で、<Table-B>のビットパターンを指定する事で、Unicode上の符号位置を指定し、Baseの値を更新する。

<C#>=41は、<Table-B>の参照回数には含まれていない。

Table-Bで参照された値は、新しい値が前の値に対し、(LSB側)右詰めで並ぶ事となる。

<C#>=41が示された場合には、次に<C#>=41が出現するまで、<Table-A>には戻らずに、<Table-B>の参照回数分の固定の呼び出し回数で、Unicode上の符号位置を指定し続ける。同じ呼び出し回数が連続する場合には、<Table-A>の呼び出し回数が減る分だけ、保存が必要な符号長を短くする事ができる。

漢字に代表される種類が多い文字の場合には、オフセットで指定できる範囲を大きく超えた符号が必要とされる。エンコードを行う時点で、固定長で繰り返した方が符号が短くできるかどうかを予め判定できるので、この様な仕様とした。

参照回数は、最大6回までで、標準が5ビットとなるため、最大30ビットまでの符号位置を直接指定できる事となる。2021年現在Unicodeの符号は最大21ビットまでの為、十二分に、仕様の拡張にも耐えられる仕組みとなっている(Table-Bの呼び出し回数は実用上5回までで、現在の仕様を満たす事ができる)。

UTF用<Table-C> (1/2)

<#C>	Baseの更新履歴	参照するtable	符号	回数
0	0	D		1
1	1	D		1
2	2	D		1
3	3	D		1
4	4	D		1
5	5	D		1
6	6	D		1
7	7	D		1
8	0	E	+	2
9	1	E	+	2
10	2	E	+	2
11	3	E	+	2
12	4	E	+	2
13	5	E	+	2
14	6	E	+	2
15	7	E	+	2
16	0	E	+	3
17	1	E	+	3
18	2	E	+	3
19	3	E	+	3
20	4	E	+	3
21	5	E	+	3
22	6	E	+	3



&lt;Table-C&gt; (2/2)

<#C>	Baseの更新履歴	参照するtable	符号	回数
23	7	E	+	3
24	0	E	-	2
25	1	E	-	2
26	2	E	-	2
27	3	E	-	2
28	4	E	-	2
29	5	E	-	2
30	6	E	-	2
31	7	E	-	2
32	0	E	-	3
33	1	E	-	3
34	2	E	-	3
35	3	E	-	3
36	4	E	-	3
37	5	E	-	3
38	6	E	-	3
39	7	E	-	3
40	Table-Bの組み合わせで参照された最大値	D		1
41	Table-Bの組み合わせで参照された最小値	D		1
42	使用しない			
43	使用しない			
44	使用しない			

UTF用<Table-C>に対応する処理:

過去のBaseの更新履歴からのオフセットでUnicode上の符号位置を参照し、Baseの値を更新する。この更新処理は、Table-DとTable-Eの中で行われる。

回数の項目は、それぞれのTableを連続で参照する回数である。

<#C>が0~7の場合と、40、41の場合には、<Table-D>を参照する。

<#C>が8~39の場合には、<Table-E>を参照する。

UTF用&lt;Table-D&gt; (1/2)

<#C>	下位5ビットの値	オフセット
0	0x00	
1	0x01	
2	0x02	
3	0x03	
4	0x04	
5	0x05	
6	0x06	
7	0x07	
8	0x08	
9	0x09	
10	0x0A	
11	0x0B	
12	0x0C	
13	0x0D	
14	0x0E	
15	0x0F	
16	0x10	
17	0x11	
18	0x12	
19	0x13	
20	0x14	
21	0x15	
22	0x16	

UTF用&lt;Table-D&gt; (2/2)

<#C>	下位5ビットの値	オフセット
23	0x17	
24	0x18	
25	0x19	
26	0x1A	
27	0x1B	
28	0x1C	
29	0x1D	
30	0x1E	
31	0x1F	
32	0x00	-1
33	0x01	-2
34	0x02	-3
35	0x03	-4
36	0x04	-5
37	0x05	-6
38	0x06	-7
39	0x07	-8
40	0x08	-9
41	0x09	-10
42	使用しない	
43	使用しない	
44	使用しない	

UTF用<Table-D>に対応する処理:

<Table-C>の表で指定された過去のBaseの更新履歴を参照し、下位の5ビットを表で指定された値に置き換える。Offsetの指定がある場合には、この値に対し更にOffsetの値を加えた値によって、Unicodeの符号位置を指定し、Baseの値を更新する。

<Table-D>の処理の後は、<Table-A>に戻る。

UTF用&lt;Table-E&gt; (1/2)

<#C>	数値
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	8
8	9
9	10
10	11
11	12
12	13
13	14
14	15
15	16
16	17
17	18
18	19
19	20
20	21
21	22
22	23

UTF用&lt;Table-E&gt; (2/2)

<#C>	数値
23	24
24	25
25	26
26	27
27	28
28	29
29	30
30	31
31	32
32	33
33	34
34	35
35	36
36	37
37	38
38	39
39	40
40	41
41	42
42	使用しない
43	使用しない
44	使用しない

UTF用<Table-D>に対応する処理:

<Table-C>の表で指定された過去のBaseの更新履歴を参照し、<Table-C>の表で指定された回数だけ<Table-D>の表の値を読み出し、<Table-C>の指定に従って、オフセット値の加算または減算を行って得られた値でUnicodeの符号位置を指定し、Baseを更新し、<Table-A>に戻る。

Offset値の求め方:

参照回数が2の場合

$$\text{Offset} = D1 + D2 * 42;$$

参照回数が3の場合

$$\text{Offset} = D1 + D2 * 42 + D3 * 42 * 42;$$

Offset値の指定に対する注意事項:

負の場合には、Baseの値を超えてはならない。

<UTFエンコード>の利用上の注意事項:

<UTFエンコード>に必要なビット数は、データのパターンに依存して大きく変動する。従って、最終的なデータサイズを比較して、<UTFエンコード>を利用するのか、QRコードの漢字モードを利用するのか、バイナリモードでUTF-8でエンコードするのかを選ぶべきである。

ローマ字表記について:

日本語の固有名詞を、アルファベットとで表記する手段として、2021年現在で、ヘボン式ローマ字、日本式ローマ字、訓令式ローマ字、修正ヘボン式ローマ字、ISO 3602、99式ローマ字、JSLローマ字、などの様々な形式が混雑している。

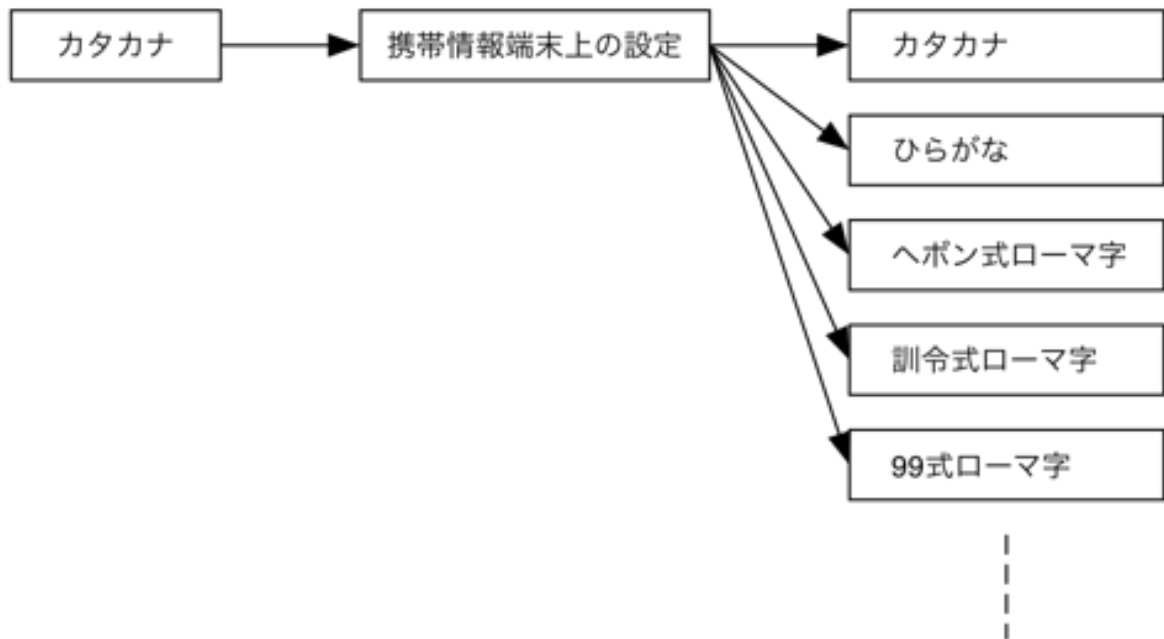
これらの形式は、それぞれにメリット、デメリットがあり、今後も、どれか一つに集約されるのでは無く、その用途と、運用する組織毎に、従来から利用されている方式の使用が継続され、全体としては、混雑して使用され続けていくと考えられる。その為、LPSおよびGMSでは、全ての方式を許容できる規格とする事にした。ローマ字は、それぞれが、最も適切判断する形式を利用し、表記するものとする。

ローマ字表記を行う場合には、<固有名詞切り替え>のエスケープ文字(/+)を利用し、固有名詞を表現するモードの中で、英数字モードや場合によっては、バイナリモードでUTF-8、もしくは、<UTFエンコード>を活用して、それぞれの方式で記述すれば良い。

但し、QRコードの英数字モードを主に利用してエンコードしている都合上、アクセント記号が入った場合は、必要となるデータ量が、大きくなる特性を持つ。マクロンを使用した長音表記だけは、2倍のサイズに抑える事ができるが、サーカムフレックスを利用する為には、UTF-8での表記が必要となる為、必要な、データの量が、マクロンよりも増加する事となる。

## LPSおよびGMSでのカタカナ表記のエンコード手法について

ローマ字には様々な記法があり、どの記法が適切であるかは、用途により異なる。そこで、LPSおよびGMSは、カタカナで表記し、携帯情報端末側の設定により、任意の表記方法に変化する方法を提案する。



ローマ字から元のカタカナには、逆変換ができない場合もあるが、カタカナから、ローマ字への変換規則は、それぞれのローマ字の方式で厳密に決められている為、カタカナから、ローマ字への一方向であれば、無理なく変換が可能になると考えられる。

QRコードの、漢字モードを利用して、全角カタカナで表記する事ができる。また、半角カタカナは、使用しないものとする。



<S式ローマ字>:

モードの指定	英語名	実際の記述	効果
<S式ローマ字>	<s-type-kana-encode>	/J	<S式ローマ字>によるエンコード開始

「/J」は<S式ローマ字>を使用した、カタカナが英数字と記号を利用して符号化されている事を意味するエスケープ文字として使用する。

本来のローマ字とは異なるが、ここでは、符号化方式を便宜上、<S式ローマ字>と呼ぶ事とする。

<S式ローマ字>の中で、再度「/J」が呼び出された場合には、呼び出し前の、元の状態に戻るものとする。

本規格においては、QRコードの漢字モードを利用して、カタカナで表記する方法が利用できるものとするが、QRコードのモード切り替えのオーバーヘッドが大きくなる為、より少ないデータ量で済む様に、英数字モードのままカタカナをエンコードできる仕組みを定義する事とする。

<S式ローマ字>にの詳細ついてできれば、本書にそのまま盛り込む事を検討していたが、<S式ローマ字>を実装する為に必要な表の行数が合計で、千行(30頁以上)を遥かに超えているため、詳細は別紙「S式ローマ字(2021年版)」に示すものとし、本書には、その実装の概要と仕組みに、表の一部と、例だけを記載する事とする。

## <S式ローマ字>の概要:

ローマ字表記は、主に、日本語の固有名詞を、アルファベットを用いて、海外の方に、発音を転写することを目的とした表記方法の一種と言える。

また、ローマ字では、海外の言語で利用されているアルファベット表記の発音に近付ける様に様々な表記方法が、今日まで、利用されているが、これらは、発音に重きを置いている為、表現に必要となる文字数が、多くなり、必然的に記録が必要となるデータ量が大きくなる特性を持つ。

しかしながら、LPSおよびGMSでは、媒体としてQRコードを利用する事を想定しているため、ローマ字で表記する事は、必要とするデータ量をできる限り小さく抑えたいという、相反する要求する課題を解決する符号化の手法である。

符号化の効率を最優先とし、一般的なローマ字では、使われていない、文字の組み合わせや、アラビア数字も組み合わせる事で、辞書圧縮の考え方を利用して、できる限り少ない文字数で、特に日本国内の地名、建物の固有名詞を符号化できる様に設計したつもりである。

辞書圧縮で有名なLZH等の手法では、過去に復号した情報を辞書として利用している。本規格では、<スタック定数>という概念を導入する事で、このアイデアも実装しているが、特にLPSでは、短い文字列しか利用されない事が想定されるため、過去に複合した情報を辞書として利用する方法が上手く機能しない可能性が高い、そこで、<S式ローマ字>では、地名、及び地名などの固有名詞を構成する綴りを2文字から3文字前後に分解した綴りを辞書として利用している。

<S式ローマ字>の仕組み:

「S式ローマ字(2021年版)」では、制御コード等を除き、主に8つの変換表を利用して英数字からカタカナへの復号化、または、カタカナから、英数字への符号化を行う。表がA,B,C,Zの4つの系列に分類される理由は、系列毎に組み合わせ可能な並びに制限が存在する為である。

<S式ローマ字>英数字カタカナ変換表の一覧表

#	表	系列	符号長	項目数	先頭の符号	中間の符号	終端の符号
1	A1	A	1	6			9, A, E, I, O, U
2	A2	A	2	105	B~Z (A, E, I, O, Uを除く)		A, E, I, O, U
3	A3	A	3	1,000 +α	B~Z (A, E, I, O, Uを除く)	B~Z (A, E, I, O, Uを除く)	A, E, I, O, U
4	B1	B	1	7			2~8
5	B2	B	2	147	B~Z (A, E, I, O, Uを除く)		2~8
6	B3	B	3	441	B~Z (A, E, I, O, Uを除く)	9	B~Z (A, E, I, O, Uを除く)
7	C2	C	2	441	B~Z (A, E, I, O, Uを除く)		B~Z (A, E, I, O, Uを除く)
8	D1	D	1	21	B~Z (A, E, I, O, Uを除く)		

<S式ローマ字>の符号表の組み合わせ:

符号の先頭では、どの系列(A,B,C,Z)から始めても良いが、後続の符号に対する制限が一部の系列には存在する。

A系列:

後続の系列に制限は無い

B系列:

後続の系列に制限は無い

C系列:

後続の系列は、B系列または、D系列に、制限される。

D系列:

後続は何も許されない。

英数字による符号の間に、空白や、ハイフン(-)、アスタリスク(\*)、エスケープ文字が挟まった場合に、この後に続く英数字の符号は、符号の先頭として扱われる。従って、C系列、または、D系列の後ろに、A系列やB系列の符号を続けたい場合には、間にアスタリスク(\*)を挟む事により、1文字分の符号長の増加と引き換えで、自由に続ける事ができる様になっている。

<S式ローマ字>の符号表一部抜粋:

A系列-1

#	符号	カタカナ	備考
1	A	ア	
2	E	エ	
3	I	イ	
4	O	オ	
5	U	ウ	
6	9	ー	(長音記号)

B系列 - B1

#	符号	カタカナ	備考
1	2	アン	安案庵
2	3	エン	縁円園苑演
3	4	イン	印院員隠
4	5	オン	音温恩
5	6	ウン	運雲
6	7	ン	(撥音記号)
7	8	ッ	(促音記号)

D系列 - D1

#	符号	カタカナ	備考
1	B	バ	馬場
2	C	チ	地
3	D	ダ	田
4	F	フ	府
5	G	ガ	河賀
6	H	ハ	派
7	J	ザ	座
8	K	キ	木
9	L	コ	湖
10	M	ツ	津
11	N	ノ	野
12	P	ワ	和
13	Q	ク	区
14	R	ギ	木
15	S	シ	市
16	T	タ	田
17	V	ブ	武
18	W	ゴ	後
19	X	カ	日
20	Y	ヤ	谷野屋
21	Z	ズ	図洲

## A系列 - A2 (一部抜粋)

#	符号	カタカナ	末尾の符号
1	BA	バ	A
2	BE	ベ	E
3	BI	ビ	I
4	BO	ボ	O
5	BU	ブ	U
6	CA	カー	A
7	CE	ガッ	E
8	CI	チー	I
9	CO	テレ	O
10	CU	サッ	U
11	DA	ダ	A
12	DE	デ	E
13	DI	ヂ	I
14	DO	ド	O
15	DU	ヅ	U
16	FA	ファ	A
17	FE	フェ	E
18	FI	フィ	I
19	FO	フォ	O
20	FU	フ	U
21	GA	ガ	A
22	GE	ゲ	E
23	GI	ギ	I
24	GO	ゴ	O
25	GU	グ	U

## B系列 - B2 (一部抜粋 - 1)

#	符号	カタカナ	備考
1	B2	アカ	赤
2	B3	アク	空く悪開く
3	B4	アサ	朝麻
4	B5	アシ	足葦
5	B6	アジ	味鱈
6	B7	アツ	圧
7	B8	アマ	甘天
8	C2	アラ	新荒
9	C3	アリ	有
10	C4	イガ	伊賀
11	C5	イシ	石医師
12	C6	イタ	板
13	C7	イチ	一壱位置市
14	C8	イヌ	犬
15	D2	イマ	今
16	D3	ウマ	馬
17	D4	ウラ	浦裏
18	D5	カメ	亀
19	D6	カワ	川河皮
20	D7	ガン	元
21	D8	キク	菊
22	F2	キシ	岸
23	F3	キヨ	
24	F4	クニ	邦国國
25	F5	クメ	久米

## B系列 - B2 (一部抜粋 - 2)

#	符号	カタカナ	備考
43	J2	ジ	字事時地寺路
44	J3	シー	
45	J4	シゲ	資茂
46	J5	シツ	シツ
47	J6	ジツ	
48	J7	シナ	
49	J8	シブ	
50	K2	ジャン	
51	K3	シュ	
52	K4	ジュ	
53	K5	シュウ	秋周終週集
54	K6	ジュン	潤純順準
55	K7	ショ	初書所
56	K8	ジョ	所
57	L2	ショウ	商賞勝小昭翔
58	L3	シロ	白城
59	L4	シン	新真進神
60	L5	ス	洲
61	L6	スミ	墨

## B系列 - B2 (一部抜粋 -3)

#	符号	カタカナ	備考
139	Y7	ヨコ	横
140	Y8	ヨネ	米

## C系列 - C2 (一部抜粋)

s	符号	カタカナ	備考
1	BB	アキ	秋
2	BC	アケ	明
3	BD	アメ	雨
4	BF	アヤ	綾
5	BG	イカ	以下烏賊
6	BH	イケ	池
7	BJ	イゴ	以後囲碁
8	BK	イズ	伊豆
9	BL	イセ	伊勢
10	BM	イド	井戸
11	BN	イナ	否
12	BP	イネ	稲
13	BQ	イノ	猪
14	BR	イモ	芋
15	BS	イヨ	伊予
16	BT	ウガ	宇賀
17	BV	ウシ	牛
18	BW	ウミ	海
19	BX	ウメ	梅
20	BY	ウリ	売り瓜
21	BZ	エキ	駅益液
22	CB	エゾ	蝦夷
23	CC	エチ	越
24	CD	エツ	関越
25	CF	エド	江戸

## A系列 - A3 (一部抜粋 - 1)

符号	カタカナ
BBA	ババ
BBE	ベベ
BBI	ビビ
BBO	ボボ
BBU	ブブ

## B系列 - B3 (一部抜粋 - 1)

#	符号	カタカナ
1	B9B	アイチ
2	B9C	アイヌ
3	B9D	アカギ
4	B9F	アカシ
5	B9G	アキツ
6	B9H	アキツシマ
7	B9J	アキハバラ

## A系列 - A3 (一部抜粋 - 2)

符号	カタカナ
STA	スタ
STE	ステー
STI	スチー
STO	ストー
STU	スター

## B系列 - B3 (一部抜粋 - 2)

#	符号	カタカナ
176	L9K	シナガワ
177	L9L	ジビキ
178	L9M	シブヤ
179	L9N	シマネ
180	L9P	シマント
181	L9Q	シモタカ

## A系列 - A3 (一部抜粋 - 3)

符号	カタカナ
VWA	ヴ
VWE	ヱ
VWI	ヰ
VWO	ヱ

## B系列 - B3 (一部抜粋 - 3)

#	符号	カタカナ
268	Q9T	トウカイドウ
269	Q9V	トウキョウ
270	Q9W	トウゲ

## A系列 - A3 (一部抜粋 - 4)

符号	カタカナ
ZZA	ゝ (濁点)
ZZE	゜ (半濁点)
ZZI	(小文字)
ZZO	(予約1)
ZZU	(予約2)

## B系列 - B3 (一部抜粋 - 4)

#	符号	カタカナ
300	S9H	ニッコウ
301	S9J	ニッポリ
302	S9K	ニッポン
303	S9L	ニホン



<S式ローマ字>の適用につて:

<S式ローマ字>では、外来語の発音についても、日本語の一部として、カタカナで表記されるものは、表記する事ができるが、あくまでも、英語表記が、原則であり、英語表記ができない、固有名詞を表現する為の手段である。

従って、外来語の発音については、<S式ローマ字>では無く、英語のものは直接英語で表記し、英語では無い、外来語の固有名詞が含まれている場合には、<固有名詞切替>のエスケープ文字(/+)を利用してこの中で英字の大文字だけを使用して表記するべきである。

しかしながら、もとは、外来語であつても、古くから日本に、取り入れられてきた言葉は、日本語の一部として、本来の外来語から、大きく発音が乖離して、定着した言葉も、固有名詞には含まれているため、必要に応じてこれらの言葉は、<S式ローマ字>で表記できるものとする。

<S式ローマ字>における、英数字から、カタカナへの復号手順:

<S式ローマ字>で符号化された文字列は、文字列の先頭から、変換表に記載されている文字列のパターンと単純にパターンマッチングを行う。

変換表上に、重なる文字列のパターンが複数存在する場合には、1番文字列が長いパターンが適用される事となる。

符号中にアスタリスク(\*)が含まれている場合には、カタカナに変換された後に、この記号は削除されるものとする。

<S式ローマ字>における、カタカナから、英数字文字列への符号化手順:

先頭から変換表に記載されてる文字列とパターンマッチングを行い最終的に必要となる文字数が最小の組み合わせのパターンを選ぶものとする。

カタカナから英数字の文字列に符号化する際に、短い文字列のパターンの組み合わせが、長い文字列のパターンと偶然一致してしまった際には、間にア

スタリスク(\*)を挿入する事により、文字数が若干上昇するが、この様な場合であっても問題なく符号化する事ができる。

人がこの符号を直接読むのではなく、あくまでカタカナ表記の符号化を目的とするため、カタカナに戻す前は、人が発音を推測できない文字の組み合わせになっていても、元のカタカナに完全に戻すことさえできれば、問題は生じないため、人がこの符号から直接発音を推測する事ができることよりも、符号化の効率を重視したものとなっている。

## <S式ローマ字>における、アラビア数字の混在

カタカナの中に数字や記号が混ざる頻度は低い為、<S式ローマ字>では、QRコードの英数字モードで用意されているアラビア数字や記号も、カタカナを表記するために、一般的なローマ字で利用されるアルファベットと同じ扱いとし符号の一部に組み込んでいる。その一方で、住所の表記では、ローマ字を示すアルファベットとアラビア数字と空白にハイフンの文字が混在する場合がある。そのため、出現頻度を考慮し、<S式ローマ字>の状態から抜けずに、アラビア数字を扱える様にする為、<S式ローマ字>の状態の中では、エスケープ文字の対応を一部変更するものとする。

### <S式ローマ字>のエスケープ文字表記

表記したい文字	役割	実際の記述
0	アラビア数字0を表す	/B
1	アラビア数字1を表す	/H
2	アラビア数字2を表す	/2
3	アラビア数字3を表す	/3
4	アラビア数字4を表す	/4
5	アラビア数字5を表す	/5
6	アラビア数字6を表す	/6
7	アラビア数字7を表す	/7
8	アラビア数字8を表す	/8
9	アラビア数字9を表す	/9
	<UTFエンコード>に切り替える	/U

一般的なローマ字は、主に英語圏の人が見て発音が分かり易い文字列の組み合わせにする為にする事と拗音が含まれる場合などの例外を除き、カタカナ一文字に対し、子音と母音の2文字の組み合わせに拘っている為、符号化の効率が下がっている。カタカナ2文字のよく利用する組み合わせに対し、英数字2文字で表現できれば、符号化の効率が約1.5倍になると考えられる。

<S式ローマ字>では、下記表に示す記号を特別な意味で使用する。

#### <S式ローマ字>の記号の用途

用途	文字
符号上の区切り文字/発音に基づいた表記に切り替え	*
発音上の区切り文字/ひらがな表記への切り替え	-
固有名詞以外の英単語かつ英数字小文字モードに切替	+

符号上の区切り文字アスタリスク(\*)は、カタカナに戻された後には、何も残る事が無い区切り文字であり、あくまでも符号化の都合で、必要にな場合にだけ使用されるメタ文字である。これに対して、発音上の区切り文字ハイフン(-)は、カタカナに変換された後も残る区切り文字である(カタカナ上では中点「・」に該当する文字となる)。また、カタカナから、ローマ字に変換される場合にも、残される(英数字記号のハイフン(-)に戻される)事を期待した文字となっている。

また、<S式ローマ字>の符号の先頭がアスタリスク(\*)で始まった場合には、綴りに基づいた表記ではなく、発音に基づいた表記であることを示す。

また、ここでの符号の先頭とは、<S式ローマ字>の先頭だけではなく、空白や、ハイフン、アスタリスク(\*)の直後も先頭と見做される。

この規定を設けた理由は、現在の一般的な文章上のカタカナ、ひらがなの綴りは、歴史的仮名遣いの影響を受けて、実際の発音と綴りに、差が生じている為である。

例えば「陽」をカタカナで書くと、「ヨウ」になるが、発音的には「ヨオ」に近くなる。これは「ウ」が長音を示している為である。

従って、日本のパスポートで用いられる一般的なヘボン式表記では、長音が省かれる為、「ヨウ」は、「YO」の2文字で綴られる。しかし、このローマ字表記をカタカナに戻すと「ヨ」の一文字になってしまい基の綴りには戻すこ

とができない、その為、”YOHの様に、”H”を長音記号の為に割り当てる綴りかたと、”Yō”の様に、マクロンを利用して長音を示す場合もある。

ハイフン(-)は、2重の場合には、ひらがなへの切り替えを示す。

もう一度、ハイフン(-)が2重に指定されると、カタカナ表記に戻る。

ひらがな表記は、固有名詞の補足で、名詞以外のものを表現する必要があった場合を想定した予約の規定である。

また、この表に空白を示す文字や、スラッシュ(/)、ピリオド(.)、パーセント(%)、コロン(:)が含まれていない理由は、文字列パラメータの規則がそのまま適用されるためである。

<S式ローマ字>では、これらのアラビア数字と記号も組み合わせた上で、更に、データ効率を引き上げる為に、次に示す目標に基づいて符号化表を設計している。

- (1) できる限り少ない文字数で、使用頻度の高い日本の固有名詞の地名をカタカナ表記できる事。
- (2) 歴史的な仮名遣いも含めてすべてのカタカナ表記を網羅できる事。

<S式ローマ字>の仕様作成の背景:

一般的なローマ字は、アルファベットの内、A, E, I, O, Uをア行、または母音とし、残りのアルファベットを子音に割り当て、母音を現すアルファベットとの組み合わせにより、表記する手法が採られている。

また、促音を示す小文字の「ッ」は子音に使用するアルファベットを2連続で表記することで示し、「ン」はnを2重に表記するか、nとmの組み合わせを利用する等の例もある。

また文末に限りn一文字で表記できる規則を持ち込む方式、子音の前に示すアルファベットを続くアルファベット文字の種類に合わせてmに変える方式や、n一文字を基本としナ行をが後に続く場合には、アポロストロフィ「'」を間に挟む方式もある。

また、拗音を示す小文字については、Xの文字と大文字の発音を示すアルファベットの符号の組み合わせで示す場合と、濁音と特定の拗音の組み合わせについてだけ、特定の子音を示すアルファベット1文字に母音を示すアルファベット1文字の組み合わせで表現できる場合もある。

また、長音については、省略する方式と、hの文字を母音の後に続けて表記する事で表現する方式や母音の繰り返して、表現する方式、母音を示すアルファベットを長母音を示すマクロン付きや、サーカムフレックス付きのものと表現する場合もある。この様に、様々な方式のローマ字が、日本では使用されているが、ア行や促音、拗音、長音等の特定の条件を除き、カタカナ1文字に対し、アルファベット2文字以上で符号かされる事となる為、文字数当たりの符号化の効率は低い方式と言える。

そこで、符号化の効率を上げるため、アブジャドと呼ばれるヘブライ語や、アラビア語に見られる母音を省略する記法のアイデアを一部だけ借りて、日本の地名をできる限り母音を省略して表記する方法を取り入れる事を考えた、これは、人の名前をフルネームで表記する代わりに、イニシャルを利用して表記した場合の考え方と同じである。

候補となる名前のリストが存在し、同一のイニシャルを持つ者がいない場合は、イニシャルから、フルネームを見つける事ができる。候補の名前のリストをテーブルとして持っていれば、受け取るデータはイニシャルだけで済む為、必要な文字数を減らす事ができる。この考え方は、辞書圧縮の一種とも考える事ができる。しかし、この方式では、やはり、イニシャルの衝突問題が発生し、辞書の規模が大きくなる割には、効率が上がらないという問題も発生する。

また、この手法は、通常のローマ字表記と混在させたい場合には、通常のローマ字記法では用いる子音のアルファベットの組み合わせとは被らない他の組み合わせだけを選別するか、組み合わせが被る場合には、通常の記法との間に区切りの文字を挟む事が必要となる。

また、アブジャドで、スペースやハイフン等の区切り文字を間に入れずに、複数発音を連結したい場合には、先頭の文字から、辞書上に登録されているパターンのリストと、比較を行い、一番長いパターンの組み合わせと一致したものを採用する形となる単純にこの規則だけを適用した場合、辞書の量を増やす度に、短い文字列のパターンは、利用できない機会が増える事になる問題が発生する。

辞書に単語が追加される度に、衝突の問題の発生から、少ない文字数で記載できていた組み合わせが利用できなくなるため、新しい文字列のパターンを辞書に追加すると追加前の辞書に基づいて作成された文字列のパターンと互換性が失われてしまう事となる。

この新しい文字列のパターンが追加される度に、短い組み合わせが利用出来なくなる、互換性問題を生じる問題を回避する方法は単純に、文字列の長いパターンは、短い文字列のパターンの組み合わせとは被らない文字列パターンとする事である。



つまり、パターンの先頭や中間に利用できる文字と末尾にだけ使用する文字の種類を分けておけば、専用の区切り文字を入れなくても、パターンの末尾を判別する事が可能となり、新たにより長いパターンが追加されても短いパターンとの衝突が回避され、過去の辞書に基づいて作成された文字列であっても、辞書に新たなパターンを追加するだけで、過去に登録されているパターンの対応を維持しておけば、互換性が保証される事となる。

当初の案としては、各鉄道会社の駅番号をそのまま取込む方法を考えていたが、新駅開業や廃止に伴い、番号の振り直しを行った例があったため、駅番号を利用する方式は避ける事とした。

また、アブジャド表記では、短い文字列では、辞書に登録したいパターンが増える程、イニシャル衝突の問題を避けられない問題が生じる為、人が直接読める文字列にする案は捨てる事にした、また、アルファベットだけでは、組み合わせが限られている為、アラビア数字も、符号の一部に取り込む事とした。

また、外来語由来の発音と、特に地名を構成する漢字の発音の組み合わせに分けて、符号化を行う事とした。

外来語由来の発音は、従来のローマ字をベースに、空いている符号化の空間を埋める形で追加を行うものとした。

<S式ローマ字>は、あくまでも、カタカナを英数字と記号を利用して符号化し、モード変換により発生する、オーバーヘッドの削減と、符号化効率を優先したものとなっている。この符号は、携帯情報端末内部のソフトウェア上で、一度カタカナに復号された後に、携帯情報端末の設定に応じて、「ひらがな」や各種のローマ字表記に変換されるか、音声に変換される事を前提としたものであるため、人が直接この符号を読める事は考慮していない。

あくまでも、主に、日本国内の地名などの固有名詞で、利用頻度が高いものを短く表記する為にカタカナの文字列を英数字記号を活用して符号化する為の手段である。

日本語では、カタカナを外来語の発音を表記する為の手段として用いるが、LPSおよびGMSでは、英語を基本とし、カタカナは、日本語の固有名詞の発音を表記し、様々な方式のローマ字表記に変換する為の、中間言語的な役割で主に用いる。「ひらがな」ではなくカタカナを利用する理由は、小文字を含めるとカタカナの方が文字の種類が多いためである。

#### カタカナ/99式ローマ字/アブジャド/<S式ローマ字>の比較例

日本語の固有名詞漢字表記	カタカナ	99式ローマ字	アブジャド	<S式ローマ字>符号化		
				例1	例2	例3
東京	トウキョウ	TOUKYOU	TKY	TOUKYOU	P6GC	Q9V
横浜	ヨコハマ	YOKOHAMA	YKHM	YOKOHAMA	Y7TC	YKA
渋谷	シブヤ	SIBUYA	SBY	SIBUYA	J8Y	L9M
品川	シナガワ	SINAGAWA	SNGW	SINAGAWA	J7FF	L9K
新宿	シンジュク	SINZYUKU	SNZYK	SI7JUKU	L4LC	M9J

例1は、

この例で、示す様に、<S式ローマ字>は、より少ない文字数で、カタカナを表現する事ができる。

<S式ローマ字>では、長音符号(「ー」)、単独の撥音(「ン」)、促音(小文字の「ッ」)にアラビア数字を割りあてている。

これは、特に促音で、同じ子音のアルファベットを繰り返し指定する一般的なローマ字の方式では、符号に利用できる空間が減らす原因となる為、このような仕組みを導入する事とした。

また、アラビア数字の0と1は、アルファベットのIとOとの判別が困難な為、あえて省いている。

2～6の母音と撥音(「ン」)の組み合わせは、母音を示すAEIOUの代わりに、アクセント記号付きの文字を利用する考え方と同じである。

また2～9の用途については、あくまで原則の主な用途であり、使用頻度を考慮して、あえて規則正を無くし、別のカタカナの符号化に割り当てている部分も存在する。

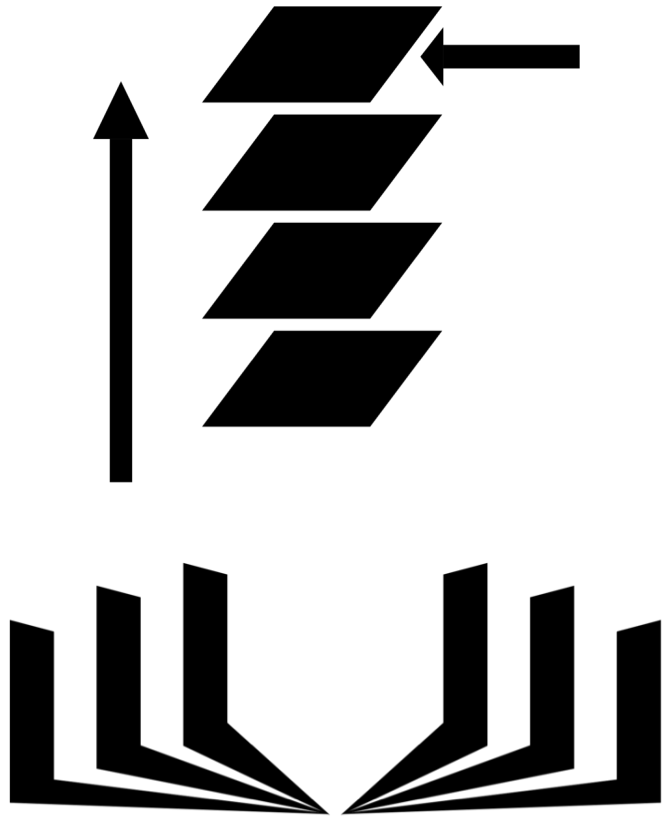
エスケープ文字列のモード切り替え一覧:

「/9」は将来的な予約として残されている。

残りのアラビア数字「/0」、「/1」は、マクロンを表すアルファベットの「/O」、「/I」との判別が困難な為、実装未遂に繋がる恐れがあるため、将来的にも使わないものとする。

モードの指定	英語名	実際の記述	効果
<caps切換>	<caps-change>	/-	英数字の小文字/大文字の切り替え
<1文字caps切換>	<one-character-caps-change>	/--	1文字だけ英数字の小文字/大文字の切り替え
<固有名詞>	<proper-noun>	/+	外来語や略語、固有名詞の指定の指定
<連続した略語/固有名詞の指定>	<continues-proper-noun>	/++	連続した外来語や略語、固有名詞の指定の指定
<S式ローマ字>	<s-type-kana-encode>	/J	<S式ローマ字>によるエンコード開始
<UTFエンコード>	<UTF-encode>	/8	<UTFのエンコード>によるエンコード開始

A ↔ a



### <文字列のスタック定数>

文字列のパラメータは、現れた順に、長さのスタックに積まれ行く、文字列のスタック定数は、文字列スタックに積まれたパラメータの値を定数として利用する為の仕組みである。

文字列スタックは、文字列のパラメータデータだけが保存されるスタックである。文字列パラメータが利用される度に、新たな文字列が登録される。

## 書式 1

**%<アラビア数字>%**

アラビア数字で示される値の位置から、スタックに、保存される値を参照する事ができる。

最後にスタックに保存されたプロパティからアルファベットで指定した番号の順に、指定したプロパティのアラビア数字で指定した順番で、スタックに保存される値を参照する事ができる。

## 書式 2

**%<アラビア数字>**

書式2は後続の文字列がアラビア数字以外の場合に利用する事ができる。

書式1は後続の文字に制限が無い。

書式1と書式2の違いは適用可能な条件の違いだけであり、機能的な制限は無い。

## 書式 3

**%%**

スタックに、直前に保存される値を参照する事ができる。

## アラビア数字

10進数で、スタックの位置を指定するために使用する。

- ※スタック定数は、長い文字数から構成されているパラメータが複数存在し、パラメータの示す値が共通であるならば、パラメータの値を使い回せる方が、文字数を削減できる為に用意された仕組みである。
- ※スタックには、プロパティのパラメータが読み込まれ、パラメータの1つの区切りに達すると1つ値が積まれる事になる
- ※一般的なソフトウェアのスタックの概念では、スタックに積む作業と、取り出す作業は、セットになっており、取り出しを行う事で、スタックの山は低くなるが、スタック定数では、スタックに積まれる一方となり、山は高くなる事は有っても、低くなる事は無い。
- ※%0%と%%が返す値は同じ値となる。
- ※スタック定数は、長い文字数から構成されているパラメータが複数存在し、パラメータの示す値が共通であるならば、パラメータの値を使い回せる方が、文字数を削減できる為に用意された仕組みである。
- ※スタックには、プロパティのパラメータが読み込まれ、パラメータの1つの区切りに達すると1つ値が積まれる事になる
- ※一般的なソフトウェアのスタックの概念では、スタックに積む作業と、取り出す作業は、セットになっており、取り出しを行う事で、スタックの山は低くなるが、スタック定数では、スタックに積まれる一方となり、山は高くなる事は有っても、低くなる事は無い。



### <定義済み文字列定数>

公共の場で施設を案内し記述する事や説明を記述する為に必要なデータを少ないデータ量で符号化できる様にする為、<定義済み文字列定数>を用意する事とした。

### <定義済み文字列定数>の書式:

書式1:

**%<アルファベット符号>%**

書式1の後続は、<定義済み文字列定数>以外の文字列が続く事となる。

書式2:

**%<アルファベット符号>**

書式2は後続の文字がアルファベットの「S」または「+」、アルファベットやアラビア数字以外の場合に使用できる。

書式1と書式2の違いは適用可能な条件の違いだけであり、機能的な制限は無い。

書式3:

**%<アルファベット符号>S**

<アルファベット符号>で符号化されている文字列の複数形の文字列の指定となる。複数形が存在しない文字列の場合には、<アルファベット符号>で符号化されている文字列の後ろに小文字の「s」が追加される。

書式4:

%<アルファベット符号>+

書式4の後続は、書式5～書式8のみが適用される。

書式5:

<アルファベット符号>%

書式5の後続は、<定義済み文字列定数>以外の文字列が続く事となる。

書式6:

<アルファベット符号>

書式6は後続の文字がアルファベットの「S」または「+」、アルファベットやアラビア数字以外の場合に使用できる。

書式5と書式6の違いは適用可能な条件の違いだけであり、機能的な制限は無い。

書式7:

<アルファベット符号>**S**

<アルファベット符号>で符号化されている文字列の複数形の文字列の指定となる。複数形が存在しない文字列の場合には、<アルファベット符号>で符号化されている文字列の後ろに小文字の「s」が追加される。

書式8:

<アルファベット符号>+

書式8の後続は、書式5～書式8のみが適用される。

アルファベット符号は、1文字目は「A」～「Z」の内、「O」と「L」を除いた、24文字を利用する。

2文字目以降は「A」～「Z」の内、「O」と「L」、「S」を除いた、23文字を利用する。

2文字目以降に「S」を割り当てていない理由は、複数形として「S」を続けて記述できる様にする為である。

書式1の場合は、後ろにアルファベットの文字列を繋げて記述する事ができる。

<アルファベット符号>は、1文字で最大24個、2文字で、最大552個、3文字で12,696個までの単語や、単語の一部分の文字列、略語、熟語を表現する事ができる。

<アルファベット符号>に、先頭に%または、+のヘッダが追加されるため、<アルファベット符号>1文字に対し、全体では2文字分のデータが必要になる為、単純に考えた場合、<アルファベット符号>1文字に対しては、3文字以上の単語を割り当て無くても、符号量の削減にはならないが、略語や固有名詞等、大文字、小文字の区別が必要な単語については、大文字小文字の切り替えの為の情報や固有名詞または、略語を示す為の切り替え情報も必要になり、この切り替え情報だけで3文字以上、のデータとなるため、この様な場合には3文字の長さの単語や略語は、

符号化する価値がある。3文字までの単語や単語の一部分は、23個。3文字までの略語は、15個を収録した。

4文字以上の単語だけで、300個以上、総数は2,000を超え、本文中に全て記載する事が適切な量では無いため、先頭から38個までを抜粋して記載する事とした。

詳細については、別紙「アルファベット符号一覧表2021年版」参照するものとする。

#### アルファベット符号一覧表2021年版より一部抜粋

Index	符号	対応文字列	文字数
1	A	ing	3
2	B	and	3
3	C	bay	3
4	D	bus	3
5	E	can	3
6	F	car	3
7	G	day	3
8	H	end	3
9	J	for	3
10	K	hot	3
11	L	how	3
12	M	key	3
13	N	low	3
14	P	new	3
15	Q	non	3
16	R	old	3
17	S	out	3
18	T	sea	3
19	U	sky	3

Index	符号	対応文字列	文字数
20	V	the	3
21	W	use	3
22	X	way	3
23	Y	you	3
24	Z	AED	3
25	AA	ATM	3
26	AB	FAQ	3
27	AC	FAX	3
28	AD	GMT	3
29	AE	GPS	3
30	AF	LAN	3
31	AG	Leo	3
32	AH	MAX	3
33	AJ	May	3
34	AK	MIN	3
35	AL	SNS	3
36	AM	TEL	3
37	AN	UTC	3
38	AP	WAN	3

メタ情報の提供:

メタ情報は、文字列のパラメータの中に埋め込む事が可能な情報である。

#### <メタ情報タグ>概要一覧

意味	書式
<メタ情報及びサービスに関するリスト>	%+<サービス符号>
<バリア情報や制限に関するリスト>	%-<制限情報符号>
<リストの区切り>	%*
<リストの終了>	\$\$

<メタ情報及びサービスに関するリスト>と<バリア情報や制限に関するリスト>には後続に<文字列>が続く種類のものと、単独で完結する種類の2種類に分けられる。

単独で完結する場合には、<リストの区切り>や<リストの終了>が直後に続く事は無い。また。単独で完結するものの後続に続く<文字列>は、メタ情報ではない。

これに対し、後続に、<文字列>が続く場合には、<文字列>の後に、<メタ情報タグ>に続ける必要がある。

<メタ情報タグ>に<リストの区切り>が選択された場合には、<文字列>の前の<メタ情報タグ>を選択した場合と同じ効果をもたらす。

単独で完結する<メタ情報タグ>の場合の書式:

書式1

<メタ情報及びサービスに関するリスト>

書式2

<バリア情報や制限に関するリスト>

後続に<文字列>が続く場合の書式:

書式1

<メタ情報及びサービスに関するリスト><文字列><メタ情報タグ>

書式2

<バリア情報や制限に関するリスト><文字列><メタ情報タグ>

例:

<メタ情報及びサービスに関するリスト>

<メタ情報及びサービスに関するリスト><文字列><リストの区切り><  
文字列>...<リストの区切り><文字列>...<リストの終了>

<メタ情報及びサービスに関するリスト><文字列><リストの区切り><  
文字列>...<リストの区切り><文字列>...<リストの終了>

<メタ情報及びサービスに関するリスト><文字列><リストの終了>

<メタ情報及びサービスに関するリスト><リストの終了>

<バリア情報や制限に関するリスト><文字列><リストの区切り><文字列>...<リストの区切り><文字列>...<リストの終了>

<バリア情報や制限に関するリスト><文字列><リストの終了>

<バリア情報や制限に関するリスト><リストの終了>

<メタ情報及びサービスに関するリスト><文字列><バリア情報や制限に関するリスト><文字列><リストの区切り><文字列>...<リストの区切り><文字列>...<リストの終了>

## サービス符号一覧

サービス符号	メタタグの説明
A	アクセビリティ
B	アクセビリティ、(アルファベット符号が後ろに続く)
C	開く扉の切り替え有り
D	階数名
E	非常用設備
F	非常用設備、(アルファベット符号が後ろに続く)
G	非常用、常用共通設備
H	許可の例外規定
W	現地の言語による表記
X	現地の言語による表記の終了
J	行き先
K	施設の外側
L	施設の内側
M	乗り物の外側
N	乗り物の内側
Y	場所の状態
Z	名称
P	地面の状況
Q	通路名
R	地名
S	提供されるサービス
T	提供されるサービス(アルファベット符号が後ろに続く)
U	外開き扉
V	許可されている行為



## 制限情報符号一覧

サービス符号	メタタグの説明
A	ステップ数
B	バリア
C	バリア(アルファベット符号が後ろに続く)
D	バリアの例外
E	禁止行為(禁止と書かれた文字列に対して適用する)
F	禁止行為(禁止と書かれた文字列に対して適用する)、(アルファベット符号が後ろに続く)
G	禁止行為(行為に対して適用する:否定文を作る)
H	禁止行為(行為に対して適用する:否定文を作る)、(アルファベット符号が後ろに続く)
Y	禁止行為の例外規定
J	高さ制限
K	重量の制限
L	人数制限
M	その為の制限
N	長さの制限
Z	内開き扉
P	幅の制限
Q	危険
R	危険、(アルファベット符号が後ろに続く)
S	警告
T	警告、(アルファベット符号が後ろに続く)
U	進入禁止
V	進入禁止、(アルファベット符号が後ろに続く)
W	安全上好ましくは無いが通行できる箇所(横断歩道、踏み切り、ガードレールが整備されていない歩道)

**EN → JP**

### <英語以外の言語の指定>

英語圏以外の地域で、地名等を表記する際に、英語圏には無い言葉が語源となっている場合に、語源となった語源の言語を指定できた方が有益な場合にだけ指定する為の仕組みである。

### <英語以外の言語の指定>の書式:

#### 書式1

**W:**<英語以外の外来語登録>.

#### 書式2

**W:**.

#### 書式3

**W:+**<英語以外の現地言語の登録>.

書式1は、/ + または、/ ++ で指定された英語以外の外来語対応する言語を英語表記または、英語表記の略号で指定する。

#### 例

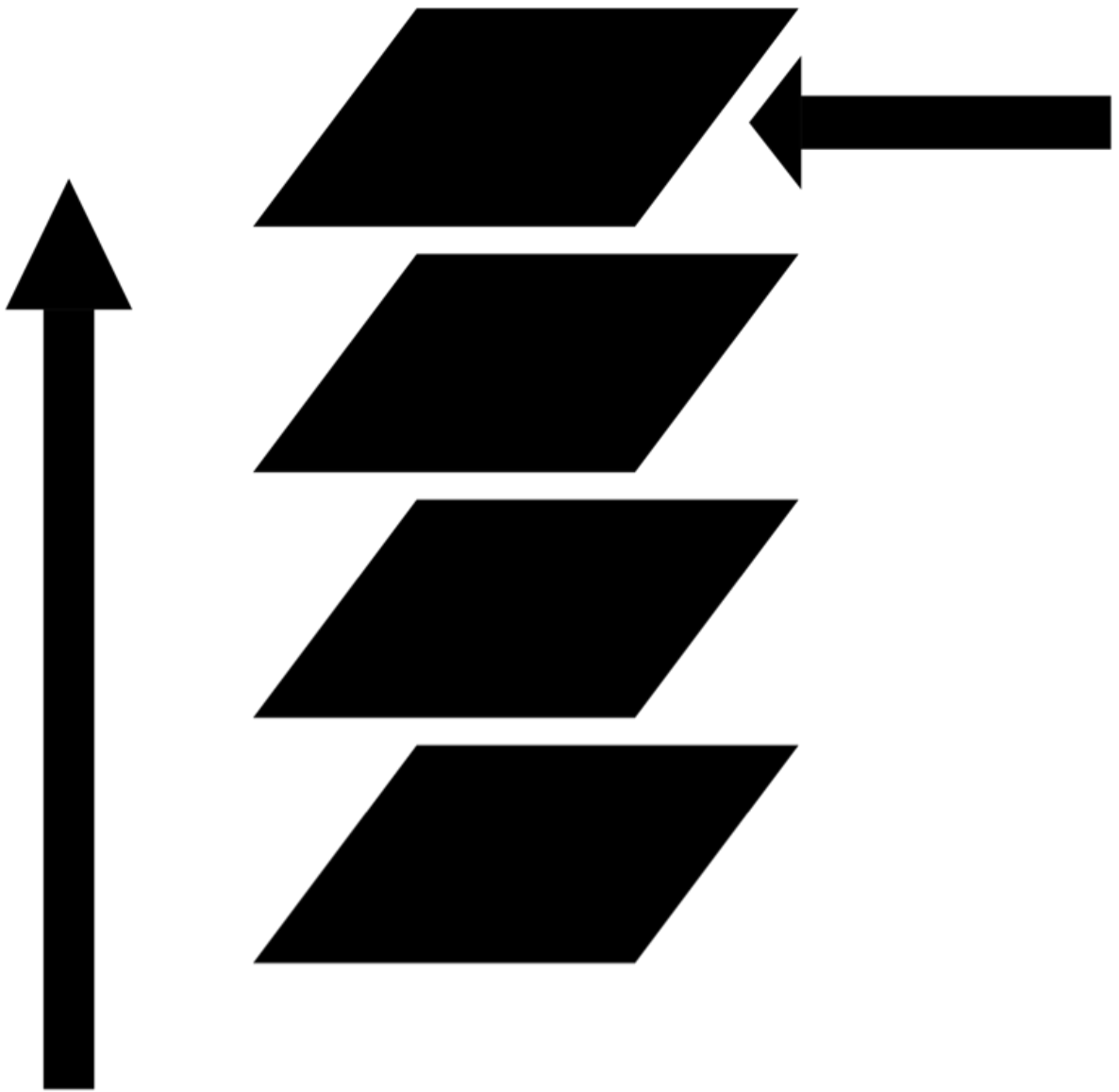
W:JAPAN.

W:JP.

書式2で指定した言語の指定を取り消す。

書式3は、%+Wで指定された、現地言語の併記で、書式1で指定した以外の言語を追加する必要がある場合に指定する。

現地の公用語が英語以外で2つ以上ある場合にだけ使用するものである。



## 8 定数の宣言:

プロパティ名	プロパティの名称	主な情報
X	<角度と長さの定数宣言のプロパティ>	角度と長さの定数宣言
Z	<文字列定数の宣言のプロパティ>	文字列定数の宣言

定数の宣言は、予め使用頻度の高い共通部分となる値を宣言しておき、スタック定数から値を参照するためのものである。定数の宣言のパラメータは、その種類に応じたスタックに、値が保存される。

定数の値は常にスタックに積まれ続け、伸び続ける。スタック定数の値が参照されても、スタックが縮む事は無い。スタック定数の値が参照された場合は、その値がそのままスタックの終わりに追加される。この仕組みにより、直近に利用された定数の値は、より少ない文字数で表現できる様になる。

また、マルチデータのスタックに保存された値は、アルファベットを利用した24進数で参照される。スタック定数は、角度、長さ、文字列の3つの種類に分類され、それぞれの値を専用のスタックに保存されるものとする)。

パラメータの分類事にスタックを分ける事で、24進数の1桁で参照できる定数を増やす事ができる。スタック定数は、説明の都合の概念上スタックと読んでいるが、その事態は、辞書圧縮を目的とした仕組みである。

角度または長さの定数の宣言:

プロパティ名	プロパティの名称	主な情報
X	<角度と長さの定数宣言のプロパティ>	角度と長さの定数宣言

スタック定数として、角度または、長さを他のプロパティから参照されるスタックに詰め込むことを目的としたプロパティである。

書式1

**X:**<パラメータ>.

書式2

**X:**<パラメータ 1>\*<パラメータ 2>\* ... <パラメータ n>.

パラメータ:

パラメータには、下記の4つの内、何れかの値が入る。

<角度の値>

<角度のスタック定数>

<長さの値>

<長さの値><長さのスタック定数>

パーセント(%)または、アルファベットから始まるものは、角度のパラメータとして、角度のスタックに積まれる。

アラビア数字から始まるものは、長さのパラメータとして、長さのスタックに積まれる。

文字列の定数の宣言:

プロパティ名	プロパティの名称	主な情報
Z	<文字列定数の宣言のプロパティ>	文字列定数の宣言

スタック定数として他のプロパティから参照される文字列の値をスタックに詰め込むことを目的としたプロパティである。

LPSやGMSでは、文字列を連結する手段は用意しているが、一部を抽出する手段を用意していない。

その代わりとして、共通の長い文字列がある場合に、定数の宣言により、共通の長い文字列を予め、文字列のスタック定数に入れておき、差分の文字列を追加する事で、抽出をしなくて済む様にするために用意された仕組みである。

他のプロパティの中では、部分的に異なるが全体としては、一致する長い文字列の共通部分だけを、指定することができない場合に活用される。

書式1

**Z:**<パラメータ>.

書式2

**Z:**<パラメータ 1>\$<パラメータ 2>\$ ... <パラメータ n>.

パラメータ:

文字列と文字列のスタック定数が入る。

文字列のスタック定数は必ず%から始まり%で終わる。

文字列のスタック定数と文字列を続けて記述する事で、文字列のスタック定数として保存されている文字列を連結し、文字列のスタックに積む事ができる。

パラメータの中で参照するスタック定数と結合する文字列の順番や個数などは自由に選ぶ事ができる。

※定数の宣言の中で、スタック定数を参照している場合は、参照先の文字列の意味と連結され、新たな文字列が作成されると解釈される。従って、文字列の連結以外に、(長さの中で扱う場合とは異なり)定数宣言の中では、演算などは行われたいものとする。



例:

定数の宣言を利用した際に、スタック上にどのような文字列が定数として積み上がって行くのかを、次の例で説明する。

この表の中ではスペースの都合上、改行されているが、実際のQRコードに保存されるデータには、改行コードは含まれない。

英数字モードの符号化の都合上、エスケープ文字を利用しているパラメータについては、実装上は、UTF-8または、UTF-16などの文字コードに、内部的には、変換されて扱われることになる。

この表で、スタック上のデータは、英数字モードの符号化の都合上、エスケープ文字を利用していたデータが内部的に、別の文字コードに符号化された状態を示している。また、#1から#3までの定数宣言は、QRコードのデータ上は、連続して並んでいるものとする。

#	定数の宣言	0	1	2	3	4	5
1	Z:JAPAN.	JAPAN					
2	Z:EHIME/C%% \$OSAKA/C%B%.	OSAKA,JAPAN	EHIME,JAPAN	JAPAN			
3	Z:TOKYO/C%C% \$SHIBUYA/C% \$SHINAGAWA/ C%B%.	SHINAGAWA, TOKYO,JAPAN	SHIBUYA, TOKYO,JAPAN	TOKYO ,JAPAN	OSAKA, JAPAN	EHIME, JAPAN	JAPAN

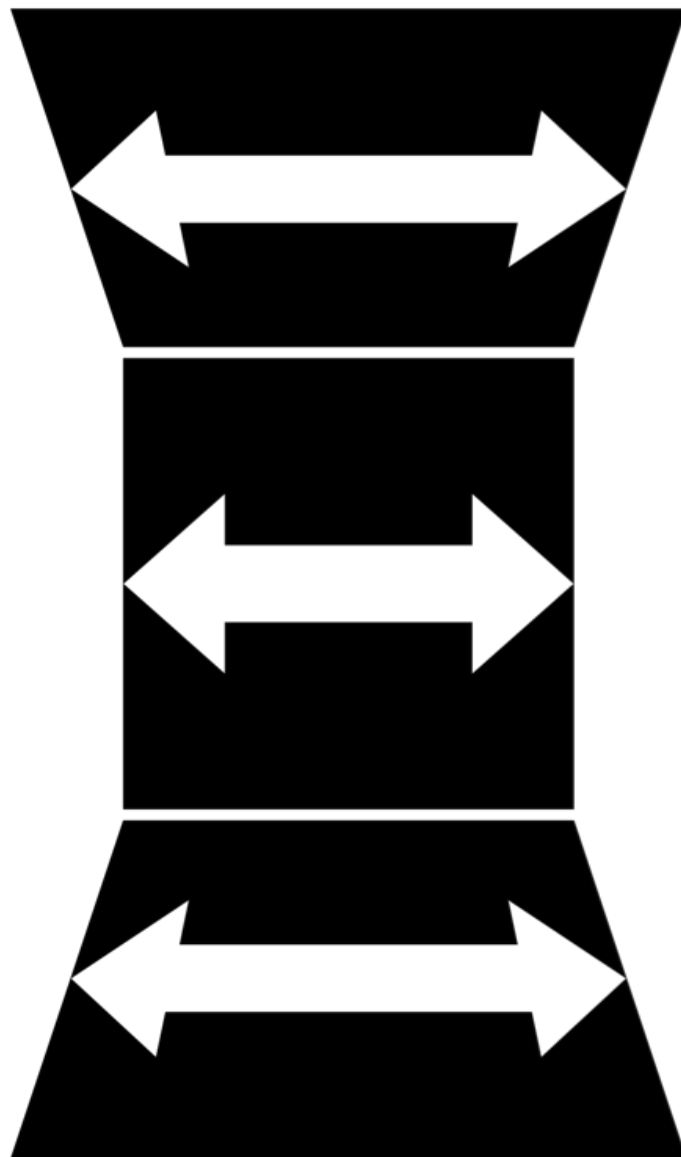
#1の定数宣言が終わると、JAPANという文字列がスタックの最後に保存される。

#2の始まりでは、JAPANという文字列が、スタックの一番上にあるので、パラメータ 1からは、%%でスタック定数として参照できるが、パラメータ 2からは、%1%で参照する事になる。

これは、パラメータ 2が始まる時点で、既に、スタック上にはパラメータ1の値が最後に入っているためである。

つまり、この表に書かれているスタックの状態は、各定数宣言を最後まで実行した際の状態であって、途中の状態は、省いている。

#3で示すように、同じ定数宣言の前のパラメータがスタックに入れた内容を、後のパラメータから、参照する事もできる。



マークの取付け角度と横幅、高さ:

プロパティ名	プロパティの名称	主な情報
M	<マークのプロパティ>	マークの取付け角度と横幅、高さ

<マークのプロパティ>は、LiDARなどを搭載していない情報携帯端末で、QRコードのマークの横幅を推測し易くする為に、マークの横幅と取り付け角度の情報を入れる事とする。

書式1

**M:**<マークの取付け角度><マークの横幅>\$<マークの床面からの高さ>.

書式2

**M:**<マークの取付け角度><マークの横幅>.

パラメータ:

<マークの取付け角度>:

QRコードの取付け角度を示す<角度のパラメータ>である。アルファベットで符号化したものか、角度のスタック定数を利用するものとする。マークの取付け角度とQRコードの向き関係の詳細については以降のページに示す。

<マークの横幅>:

QRコードの横幅を示す<長さのパラメータ>である。

<マークの床面からの高さ>:

QRコードの床面からの高さを示す<長さのパラメータ>である。

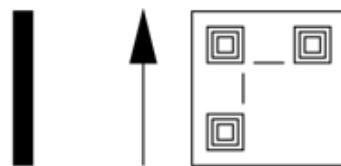
<マークの取付け角度>の詳細:

角度のパラメータである。

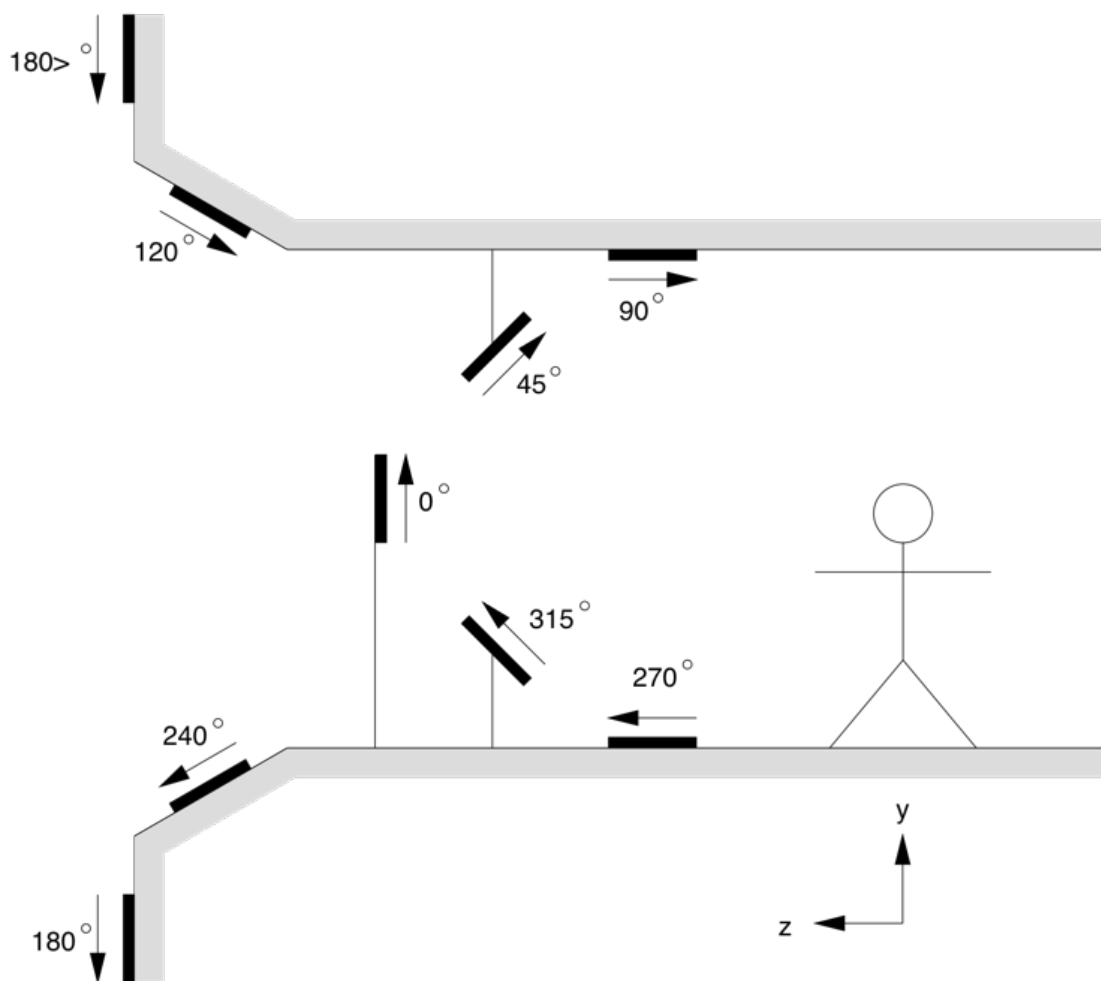
地面から垂直の壁に印刷されている状態を0度とする。

マークを側面とQRコードの向きの関係:

正面から見た場合には、下図に示した向きで、切り出しシンボルの位置が見えるものとする。



天井面に印刷されている状態を90度とする。



床面に印刷されている状態を270度とする。

<マークの取付け角度>が180度以上270度以下になる例としては、  
下り方向のエスカレーターの、ステップ脇の面に、LPSおよびGMSの情報を  
記録したQRコードを貼る場合が考えられる。

<マークの取付け角度>が180度以上270度以下になる例としては、  
上り方向のエスカレーターの、天井の面に、LPSおよびGMSの情報を記録し  
たQRコードを貼る場合が考えられる。

<マークの横幅>の詳細:

QRコードの横幅を示す<長さのパラメータ>である。

マークの横幅は、シールや紙に印刷した際の横幅を基準とする。

円筒形の円柱に適用する際の規定についての注釈:

円筒形の円柱に直接印刷した場合は、円周の長さとなり、直線の長さとはならない。QRコードが円筒形の柱の上に貼られら際に、イメージセンサーで捉えられる横幅は、データ上の横幅よりも小さくなると考えられるが、QRコードは、マーク全体が正方形となっているので、縦幅から歪みを補正し、補正が可能であると考えられるので、柱が円筒上形の場合であっても、印刷するデータは変えないものとする。但し、この規定の意味する事は、あくまでも、QRコードに記録するデータの話であって、円筒形の物体の上に、QRコードを直接貼り付ける事や印刷する事を推奨しているわけでは無い。QRコードには、円筒形の物体に、印刷されたものに対して、補正を行う仕組みを有しているが、必ずしも補正が行えて正しく認識できる保証は無い。従って、この様な場合には、円筒形物体に直接貼り付けるたり、印刷するのではなく、円筒形の物体の上に平らな板を取り付け、その上に、QRコードを貼り付けるか、平らな板の上に直接印刷する事を推奨する。

また、円筒形の円柱に対し、LPSおよびGMSのデータが記憶されたQRコードを貼り付けたり、印刷する事は推奨はしないが許される。

円筒の半径が大きくなるにつれ、その形状は、ほぼ平面と同じとなり、歪みが許容できる様になる為である。

<マークの床面からの高さ>の詳細:

<長さのパラメータ>である。

床面からQRコードの中心までの高さを指定するパラメータである。

このパラメータが省かれている場合には、下記の値を適用する。

$\theta$  = <マークの取り付け角度>

0度から90度以内まで:

$$y = 1.2 + 1.2 * \sin(\theta)$$

90度を超えてから180度の手前まで:

$$y = 2.4 - 0.6 * \cos(\theta)$$

180度以上から270度の手前まで:

$$y = 0.6 * \cos(\theta)$$

270度から360度手前まで:

$$y = 1.2 + 1.2 * \sin(\theta)$$

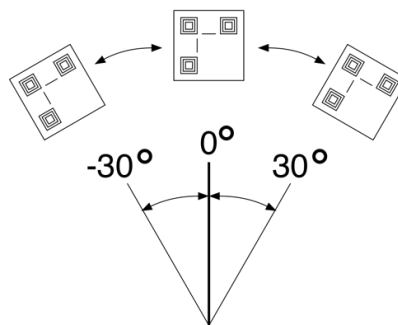
※QRコードにデジタルデータの保存されている<マークの取り付け角度>と<マークの横幅>の情報が分かれば、携帯情報端末の利用者の過去の使用履歴情報から求めた携帯情報端末の床面からの高さをもとに、イメージセンサーから求めたQRコードの光学的情報で求めた距離と角度の情報を合わせて、計算したQRコードの高さを、<マークの床面からの高さ>の情報と比較する事で、複数階に渡って見える場合に、どちらの階数を対象とした情報なのか、判別するための情報を提供する事を目的としたパラメータである。



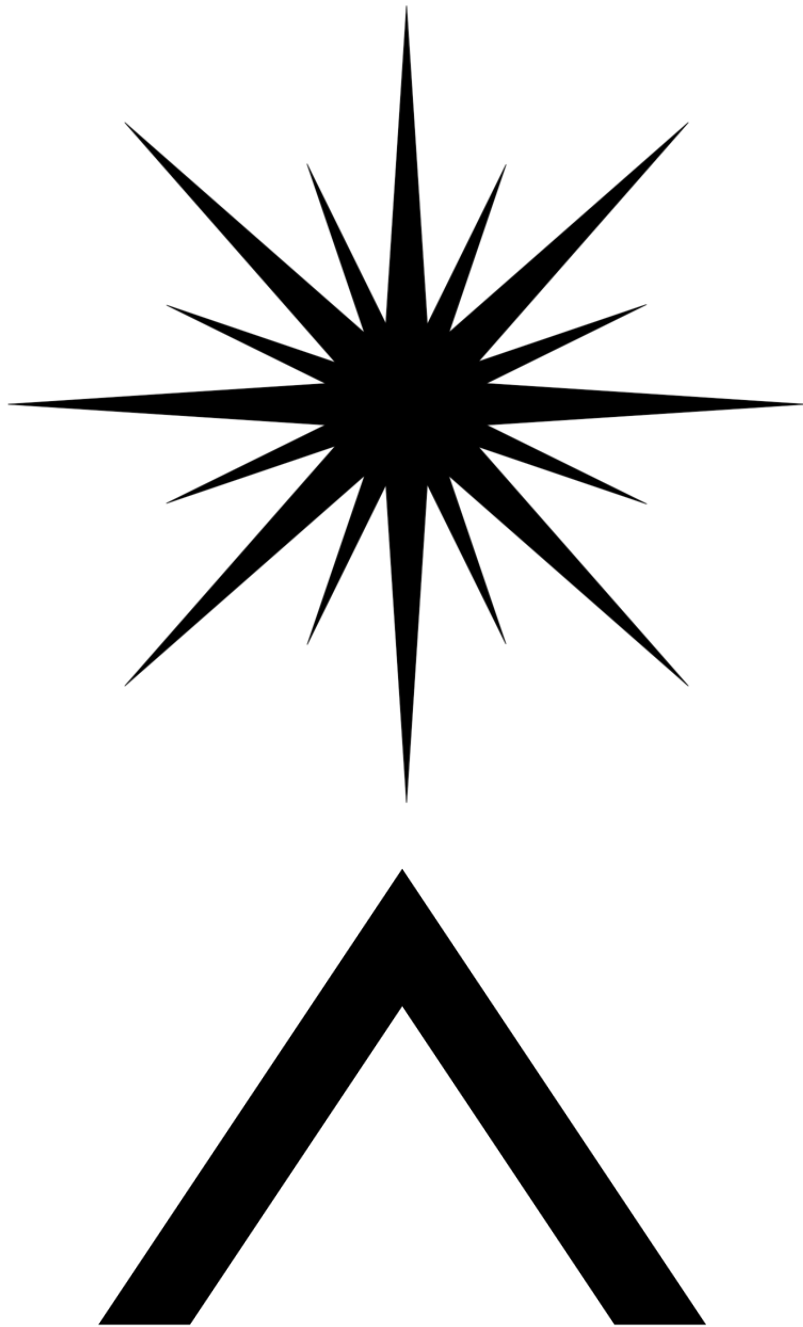
虚像による誤認識を防ぐための規定:

QRコードの取り付け角度が、0度から30度の範囲と、330度から360度(0度)の際は、マーク中央から見てz軸に対し時計回りまたは、反時計に、30度を超えてロールしているものは、虚像と見做し、認識させないものとする。  
また、鏡写しの画像も認識させないものとする。

認識されるロールの範囲



QRコードの規格自体は、如何なる角度に回転している場合も、正常に認識できる優れた性能を持つ規格であるが、LPSとして利用する場合は、虚像を排除する為、認識させないものとする。



地球上の方角:

プロパティ名	プロパティの名称	主な情報
D	<地球上の方角のプロパティ>	地球上の方角、標高

QRコードが貼られている<地球上の方角のプロパティ>である。

書式1

**D:**<地球上の方角><標高>.

書式2

**D:**<地球上の方角>.

パラメータ:

<地球上の方角>:

0度を北、90度を東、180度を南、270度を西とし、アルファベットでコード化したもので指定する。  
角度のパラメータである。

<標高>:

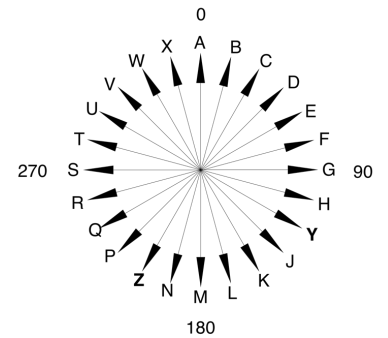
LPSのデータのデータが入ったQRコードの中心の高さを示す<長さのパラメータ>である。

<地球上の方角>:

角度のパラメータである。

0度を北、90度を東、180度を南、270度を西とし、アルファベットでコード化して、指定する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X



アルファベット1文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

例: A=0度、B=15度、D=45度、E=60度、G=90度、Y=120度、M=180度

アルファベット2文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

+360/24/24\*<2文字目の対応表で示される数値>

例:AB=0.625度、GQ=97.5+2.5=100度

1文字のAに対し2文字のAAはより高い有効精度の場合に使用される。

アルファベットn文字で表記する場合:

$$y = \sum_{k=1}^n \frac{x_k}{24^k}$$

y = 角度

$x_k$  = アルファベットで表記された対応する数値

k = アルファベットの左から数えた文字数

n = アルファベットの全体の文字数

角度とは $2\pi$ を分割する値に対する比率であり、度数法では360等分した際の割合で示すものである。LPSでは360等分では無く、 $24^n$ 等分するという考えに基づいている。その為、 $n = 1$ の場合には24等分と精度が低くなるが、 $n = 2$ の場合には576等分となり、度数方で整数の範囲で表記する場合よりも精度が良くなる。

従って実用上、アルファベット1文字から2文字で十分な精度が得られる事が期待できるが、必要に応じて3文字以上で、表記しても良い。LPSを利用する携帯情報端末が、アルファベット3文字以上で書かれたデータを受け取った際の処理として、3桁以降を値として細かく計算するか、切り捨て処理を行うかは実装の目的に合わせて判断するものとする。

※符号化する際にアルファベットの大文字のIとOはアラビア数字の1と0との判読が困難となるため、それぞれ大文字のYとZに置き換える。

※アルファベットから数値に復号する際は、アルファベットの大文字のIは数値8、アルファベットの大文字のOは、数値14として扱うものとする。

※アラビア数字ではなく、アルファベットでコード化する理由は、より少ない文字数で多くの情報を入れる事を可能にし、精度と情報量のトレードオフを可能にする為である。

例:

北の場合: D:A.

東の場合: D:G.

南の場合: D:M.

西の場合: D:S.

<標高>:

<長さのパラメータ>である。

メートルよりも細かな値を扱う必要がある場合には、ピリオドの代わりに、「M」を入れるものとする。

キロメートル単位の値を扱う場合には「K」を入れるものとする。

<床面からの高さ>と<標高>はQRコードの中心点を基準とする。

<標高>のパラメータの例:

0.12mの場合: H:0M12.

12mの場合: H:12.

-12mの場合: H:12S.

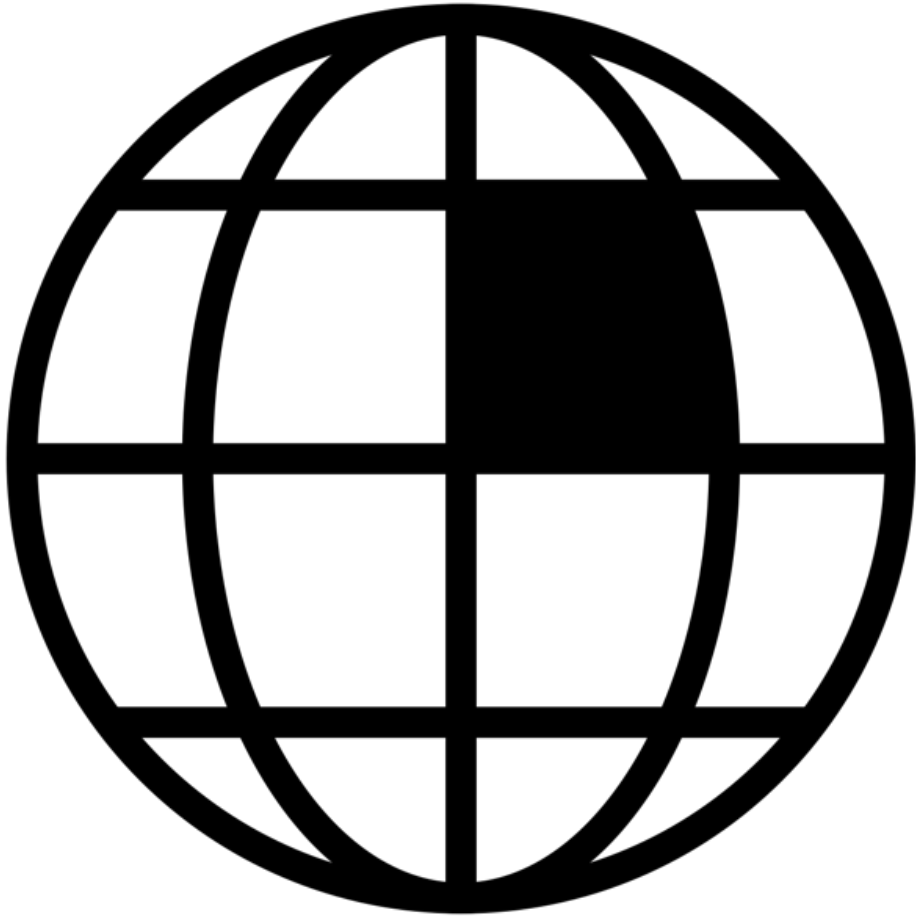
12.3mの場合: H:12M3.

12 kmの場合: H:12K.

12.3 kmの場合: H:12K3.

エレベーターの内部に掲載されるものに対しては、<標高>のパラメータは、省略するものとする。

- ※エレベーター内部に掲載されるものに対し省くという規定は、エレベーターの籠室内部に対する規定であり、エレベーターのシャフト内部に対する規定では無い。
- ※このプロパティは、ドローンや、外壁の清掃や検査を行う無人の装置で利用する事を考慮した規定である。
- ※ピリオドは、QRコードに登録されたデータを文字列として人が、目視で確認する作業が発生した場合に、見落としを減らす為に、代わりにアルファベット1文字を入れる事とする。
- ※平野だけでなく、山岳地帯し、大きな値をとる事ができる。





位置:

プロパティ名	プロパティの名称	主な情報
P	<地球上の位置のプロパティ>	地球上の位置

<位置のプロパティ>

書式1

**P:**<MGRS>.

書式2

**P:**<MGRS精度拡張書式>.

<MGRS>:

MGRS (Military Grid Reference System)の1メートル精度のものを利用する。また、MGRSは、最高で1m精度となるが、QRコードのマークを必ずしも丁度良く、MGRSの交点に配置する事ができないため、用途によっては、より細かなcm単位の精度を要求する事が考えられる。この様な要求に対応する為、cm精度への拡張も定義する。  
位置はQRコードのマークの中心を基点とする。

<MGRS>の書式:

ZZBGEEEEENNNNN

<MGRS>の各パラメータ:

文字列のパラメータである。

ZZ: UTMゾーン

B: 経度バンド

ZZBはグリッドゾーン指定 (Grid Zone Designator:GZD) と呼ばれる

GG: 100km格子

EEEEEE: X座標 (東距離) (5桁)

NNNNN: Y座標 (北距離) (5桁)

MGRSの詳細については、巻末の参考文献を参照。

<MGRS精度拡張書式>:

MGRS 100cm精度拡張書式:

ZZBGEEEEENN>NNN-en

MGRS 10cm精度拡張書式:

ZZBGEEEEENN>NNN-eenn

MGRS 1cm精度拡張書式:

ZZBGEEEEENN>NNN-eeenn

<MGRS精度拡張書式>のパラメータ:

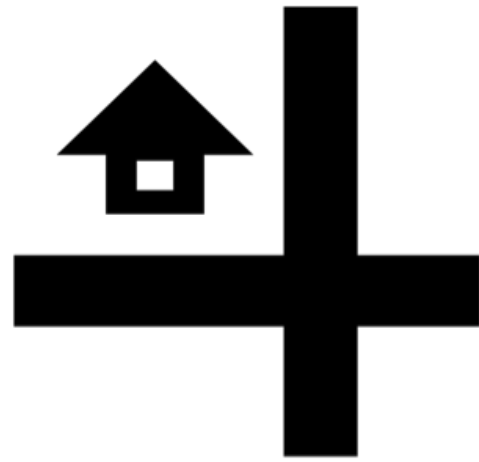
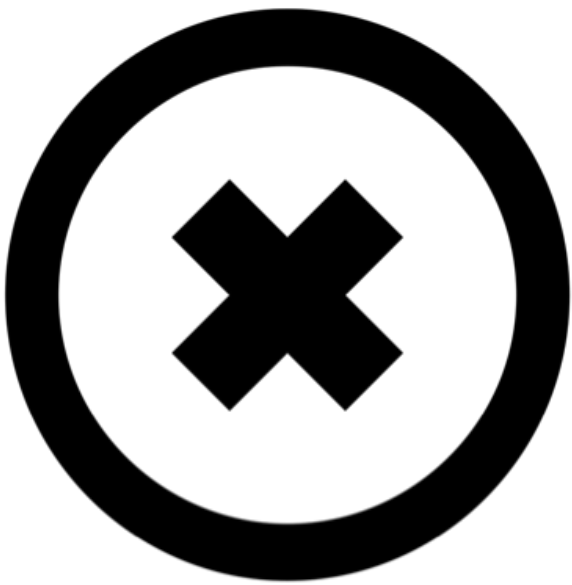
文字列のパラメータである。

ハイフンよりも前はMGRSをそのまま利用する。ハイフンより後ろが拡張部分とする。

eee : X座標 (東距離) (1桁～3桁)

nnn : Y座標 (北距離) (1桁～3桁)

※<MGRS精度拡張書式>は、MGRS本体のパラメータとハイフンに拡張精度部分も含めて1つの文字列として、文字列のスタック定数上では扱われるものとする。



領域:

プロパティ名	プロパティの名称	主な情報
A	<領域のプロパティ>	領域の有効半径

<領域のプロパティ>は、LPSやGMSがマークアップする対象の有効範囲と属性を示すものである。サブ領域のプロパティと合わせて、バリア情報の位置関係を指定する為に用いられるプロパティでもある。

領域のプロパティが持つパラメータは以下の4種類である。

<領域の基準点の位置座標>

<領域基準点の角度と有効半径>

<領域の有効精度>

<領域の属性>

各パラメータは、初期値を持っており、<領域のプロパティ>の中で、パラメータを指定する事で上書きする事ができる。

<領域のプロパティ>を複数回宣言した場合には、パラメータを省略した場合には、上書きされたパラメータの値がデフォルト値となる。

<領域の基準点の位置座標>の初期値:

マークの取り付け角度及び横幅のプロパティで指定された角度のパラメータにより変化する。

奥行き方向の初期値は下記の式で求められる。

$$\mathbf{x} = \mathbf{0}$$

$$\mathbf{y} = \mathbf{0}$$

$$\mathbf{z} = 1.2 * \cos(\theta)$$

$$\theta = \text{マークの取り付け角度}$$

地面と平行の角度にQRコードが取り付けられている場合は、QRコードは床に直接取り付けられ、真上から読み取られた位置を領域の基準点とする。

壁と平行に取り付けられている場合には、床から、1.2mの高さで、壁を正面から見て後ろに、1.2m離れた地点を初期値とする。

<領域基準点の角度と有効半径>の初期値:

このパラメータは2つのパラメータから構成され、それぞれが初期値を持っている。

<領域基準点の角度>:

0度

<領域の有効半径>:

1.2m

<領域の有効精度>の初期値:

<高さ方向の精度>: <領域の有効半径>の60分の1

<横方向の精度>: <領域の有効半径>の60分の1

<領域の属性>:

全て判別不明

領域は以下に示す16通りの書式を持つ。

#### 書式 1

基本形

**A:**<領域の基準点の位置座標>-<領域基準点の角度と有効半径>-<領域の有効精度>+<領域の属性>.

#### 書式 2

<領域の基準点の位置座標>の省略形

**A:**-<領域基準点の角度と有効半径>-<領域の有効精度>+<領域の属性>.

#### 書式 3

<領域基準点の角度と有効半径>の省略形

**A:**<領域の基準点の位置座標>--<領域の有効精度>+<領域の属性>.

#### 書式 4

<領域の有効精度>の省略形

**A:**<領域の基準点の位置座標>-<領域基準点の角度と有効半径>+<領域の属性>.

#### 書式 5

<領域の属性>の省略形

**A:**<領域の基準点の位置座標>-<領域基準点の角度と有効半径>-<領域の有効精度>.

書式 6

<領域の基準点の位置座標>と<領域の有効精度>の省略形

**A:**-<領域基準点の角度と有効半径>+<領域の属性>.

書式 7

<領域の基準点の位置座標>と<領域の属性>の省略形

**A:**-<領域基準点の角度と有効半径>-<領域の有効精度>.

書式 8

<領域基準点の角度と有効半径>と<領域の有効精度>の省略形

**A:**<領域の基準点の位置座標>+<領域の属性>.

書式 9

<領域基準点の角度と有効半径>と<領域の属性>の省略形

**A:**<領域の基準点の位置座標>--<領域の有効精度>.

書式 10

<領域の基準点の位置座標>と<領域基準点の角度と有効半径>の省略形

**A:**--<領域の有効精度>+<領域の属性>.

書式 11

<領域の有効精度>と<領域の属性>の省略形

**A:**<領域の基準点の位置座標>-<領域基準点の角度と有効半径>.



書式 12

<領域の基準点の位置座標>単独形

**A:**<領域の基準点の位置座標>.

書式 13

<領域基準点の角度と有効半径>単独形

**A:--**<領域基準点の角度と有効半径>.

書式 14

<領域の有効精度>単独形

**A:--**<領域の有効精度>.

書式 15

<領域の属性>単独形

**A:+**<領域の属性>.

※ <領域の属性>のパラメータの前の区切り文字「+」は、他のパラメータとは異なるので、他のパラメータと区別できる従って、間の区切り文字の個数を1つに抑える事ができる。

書式 16

全てデフォルトのパラメータを使用

**A:.**

<領域の基準点の位置座標>:

QRコードの真下の位置を基点に、オフセットを加える<座標のパラメータ>である。地面平行な面に対するオフセットの座標を指定する。

円座標系を利用した場合の例を下記に示す。

QRコードよりも、2m手前を基準点としたい場合の例:

M2

QRコードよりも、3.5m先を基準点としたい場合の例:

A3M5

QRコードよりも、1m右を基準点としたい場合の例:

G1

QRコードよりも、5m左を基準点としたい場合の例:

S5

平面直交座標系を利用した場合の例を下記に示す。

QRコードよりも、2m手前を基準点としたい場合の例:

2S\*0

QRコードよりも、3.5m先を基準点としたい場合の例:

3M5\*0

QRコードよりも、1m右を基準点としたい場合の例:

0\*1

QRコードよりも、5m左を基準点としたい場合の例:

0\*5S

円座標系を利用した場合の例を下記に示す。

QRコードよりも、2m手前を基準点としたい場合の例:

2S

QRコードよりも、3.5m先を基準点としたい場合の例:

3M5

<領域基準点の角度と有効半径>:

このパラメータは2つのパラメータから構成されており、下記に示す書式を持っている。

<領域基準点の角度と有効半径>の書式 1:

<領域基準点の角度><領域の有効半径>

<領域基準点の角度と有効半径>の書式 2:

<領域基準点の角度>

<領域基準点の角度と有効半径>の書式 3:

<領域の有効半径>

<領域基準点の角度と有効半径>のパラメータ:

<領域基準点の角度>:

<角度のパラメータ>である。

このパラメータを指定しない書式3を選んだ場合は、初期値の0度を指定した事となる。

<領域の有効半径>:

<長さのパラメータ>である。

このパラメータを指定しない書式2を選んだ場合には、1.2mを指定した事となる。

<領域の有効半径>:

そのLPSマークが領域として指定できる半径を示す<長さのパラメータ>である。

初期値は、1.2mであり、正の値だけを指定する事ができる。

<領域の有効半径>の内側では、<領域の属性>で指定された属性が適用される。

<領域の有効半径>の内側では、サブ領域は、高い精度を持って指定され事が必要とされる(高い精度の意味合いは、その場所に於いて、LPSおよびGMSで、提供される情報だけを利用し、視覚に障害を持つ人に、対し安全に、適切な指示を携帯情報端末が与えるのに、必要十分な精度という意味合である)。

<領域の有効半径>の外側の部分は、<領域の属性>の指定に拘らず、属性は常に、判別不明として扱われる。

<領域の有効半径>の外側の部分は、内側よりも、低い精度となり、精度が保証されない曖昧な内容を含む事が許容される。

値の例:

12mの場合: 12

12.3mの場合: 12M3

※GPSが利用できない場所に於いて、携帯情報端末内蔵の加速度センサーを利用し、積算量から位置を求める事になる。LPSは、GPSを補完し、QRコードの読み取り位置での精度を高める効果をもたらすが、1つのLPSを読み取った場合は、距離が離れて行くほど、得られる精度は低下する。1つのQRコードで提供できる情報量は、限られる為、特にLPSに於いては、QRコードの周囲の限られた範囲内で、利用者に役立つ情報だけに限定する事で、情報量の削減を行う事

が求められる。有効半径を指定する事で、有効半径外の部分が精度が期待できない曖昧な部分が含まれている事を厳密に指定する事ができる。

<領域基準点の角度>:

<領域基準点の角度と有効半径>の中で指定できる、<角度のパラメータ>である。Y軸に対する相対角度を指定する。このパラメータにより、X軸、Z軸の向きが変化する。このパラメータの指定が省略された場合は、初期値の0度を指定した事となる。

<領域の有効精度>:

書式 1:

<水平方向の精度>\$<高さ方向の精度>

書式 2:

<水平方向の精度>

書式 3:

\$<高さ方向の精度>

パラメータから省略された場合:

<高さ方向の精度>: <領域の有効半径>の60分の1

<横方向の精度>: <領域の有効半径>の60分の1

書式2と書式3で指定されなかったパラメータに、対しては、初期値が適用されるものとする。

<水平方向の精度>と<高さ方向の精度>は<長さのパラメータ>である。

<領域の有効半径>内の保証している精度を指定する事ができる。

<領域の属性>:

<領域の属性>は、領域有効半径内の背景領域とサブ領域が、公開(公共空間)して良い場所か、非公開(私的空間)の場所か、どちらか、分からない判別不明の場所であるかを判別する為のパラメータである。

アルファベットとアラビア数字の組み合わせにより、各領域の番号に対応する属性を指定する。

<領域の属性>のパラメータの中では、領域に下記の番号付けを行う

領域 #1: 背景領域 (領域のプロパティの有効半径内)

領域 #2: サブ領域#1

領域 #3: サブ領域#2

領域 #4: サブ領域#3

...

領域 #x: サブ領域#(x-1)

領域番号が小さいものと、大きなものが重なった場合は、大きな番号の領域の属性が優先される。

<領域の属性>のパラメータは、アルファベットの大文字で始まり、各アルファベットの文字は、1文字で1個から最大3個までの領域の属性を指定することができる。

アルファベットの後にアラビア数字を続けて、記述する事で10進数により、最後に指定した属性を更に追加で何文字分繰り返すのかを指定することができる。また、アラビア数字の後には、アルファベットで再度残りの属性を指定できる。

領域番号の個数に対し、領域のパラメータで指定した属性の個数が不足している場合には、最後に指定した属性がそのまま残りの全てに適用される。

また、<領域の属性>のパラメータが存在しない場合は、属性が判別不明である事を示す(符号Zを指定した事になる)。

#### アルファベットと属性の対応

符号	1	2	3
A	g	n	
B	g	g	
C	g	z	
D	g	n	z
E	g	g	z
F	g	z	g
G	g		
H	g	n	g

符号	1	2	3
J	n	n	
K	n	g	
L	n	z	
M	n	n	z
N	n		
P	n	g	z
Q	n	n	g
R	n	g	n

符号	1	2	3
S	z	n	
T	z	g	
U	z	z	
V	z	n	z
W	z	g	z
X	z	z	n
Y	z	z	g
Z	z		

g: 公開 (公共空間)

n: 非公開 (私的空間)

z: 判別不明

例:

領域属性のプロパティにAを指定した場合の例:

#1: 背景領域: g: 公開 (公共空間)

#2: サブ領域#1: n: 非公開 (私的空間)

#3: サブ領域#2: n: 非公開 (私的空間)

#4: サブ領域#3: n: 非公開 (私的空間)

...

<領域の属性>のプロパティにS3Eを指定した場合の例:

- #1: 背景領域: z: 判別不明
- #2: サブ領域#1: n: 非公開 (私的空間)
- #3: サブ領域#2: n: 非公開 (私的空間) #1
- #4: サブ領域#3: n: 非公開 (私的空間) #2
- #5: サブ領域#4: n: 非公開 (私的空間) #3
- #6: サブ領域#5: g: 公開 (公共空間)
- #7: サブ領域#6: g: 公開 (公共空間)
- #8: サブ領域#7: z: 判別不明
- #9: サブ領域#8: z: 判別不明

...

LPSは、基本的に公開空間の指定に用いるものとするが、集合住宅の廊下や、病院の待合室、工場の搬出入路など、外部から、特定の来場者の侵入を許可するが、基本的に、行動を非公開とする領域や、公共施設であっても、公開部分と、関係者専用の非公開部分、その境界に隣接する場所が考えられる。

この属性は、マップ情報を提供するプロバイダに対し、移動をトラッキングして得られた携帯情報端末利用者の移動情報の公開を禁止する事を通知する為の情報を示す。

公開(公共空間)領域の属性の効力は、LPSの有効半径の領域内で、かつ有効半径内で、サブ領域の指定がある場合には、サブ領域の指定に従う。同じ地点で重なるサブ領域がある場合は、後から出てきたサブ領域の指定に従うものとする。



LPSの場合は、記載されているサブ領域の情報が複数の階に繋がっているサブ領域が定義されている場合は、そのサブ領域の指定も有効となる。

現在の階数に限定されている場合には、公開(公共空間)領域の属性の指定は、その階にのみ有効となり、異なる階数に対しては、影響をもたらさないものとする。

これに対しGMSでは、階数のプロパティは複数宣言する事で、複数の階数にまたがる情報を記載する事ができ、複数階数プロパティや領域のプロパティも、記載できる。

いずれ場合も、<領域の有効半径>を超えた場合は、判別不明として扱われる。

LPSの情報を含んだQRコードを複数認識し、領域の指定に矛盾が生じた場合には、床面に対して高い位置に存在し、物理的に高い位置に存在するもの、物理的にマークの横幅が大きいもの、対象の地点に近くのあるもの、作成日時情報が新しいもの、GPSが利用でき場合には、GPSから求められる位置情報との矛盾が無いもの、GPSが利用できない場所では、加速度センサーからの累積で求められた位置情報との誤差が少ない値の情報を持つものを正確な情報とし扱うものとする。

非公開(私的空間)は、トラッキングしてはならない領域という意味である。

但し、非公開(私的空間)を地図的な情報が、外部に漏れてはならない秘密の領域と解釈してはならない。地図的な情報が外部に漏れる事が不利益<sup>1</sup>と外部のネットワークに接続可能という条件が加われば、本質的に情報が流出する可能性が高くなる。また、公開(公共空間)領域と隣接する場合、LPSに含まれている情報自体が外部のサーバーに渡る場合がある。

従って、隣接部分に於いても、秘密にしたい領域については、細かな情報をLPS上に記載してはならない。公開(公共空間)との境界だけが分かる外側の境界の形状だけを情報に入れ、その他の情報は、記載しないべきである。

但し、公開(公共空間)と物理的な進入を防ぐ仕切りが無く、連続して繋がっている場所に於いて、視覚に障害を持ったものが、侵入してしまった際に、命の安全に関わる障害(例えば、足を踏み入れたら自力で抜け出す事が不可能な柵の無い溜池がある場合など)については、正しくLPS上に記載するべきである。

※位置情報サービスは、非常に有益なものであるが、特に屋内では、プライバシーの問題が発生すると考えられる。プライバシーの問題が、LPSの普及の障害とならないようにする為、LPSを利用したサービスを提供するプロバイダに対し、位置情報をトラッキングする事で得られたデータを公開する事が不適切な領域を通知できる仕組みを定義する。

※技術的な視点から見た場合、カメラや加速センサー、WiFi機能を内蔵している情報携帯端末は、位置のトラッキングは可能である。仮に、本ドキュメントが提案するQRコードが印刷されて居なくても、加速度センサーにより、積算され蓄積された移動量の情報に、後から、GPSの座標の情報を入手する事ができれば、後追いで、それまでの移動経路を逆算することができる。また、カメラ画像とニューラルネットワークによる画像認識による特徴のフィルタリングの技術と十分なデータの蓄積が在れば、確度は下がるが、位置の特定は可能となる。更にWiFiにアクセスできる端末は、WiFiの電波強度の分布という指紋の情報をもとに、データベースと照合する事により、位置を特定する事ができる。従って、本、ドキュメントが提案するQRコードを利用した、位置座標と方角を計測するシステムが導入されなくても、条件さえ整えば、情報携帯端末は、位置の特定や推定がある程度の範囲で実現可能であると言えるので、LPSは、<領域の属性>により、非公開を指定できる事は、本システムを導入する前よりも、より座標情報のトラッキングを利用したサー

ビスを提供する業者に、自発的に個人のプライバシーの保護を守らせる事に  
寄与する仕組みであると言える。

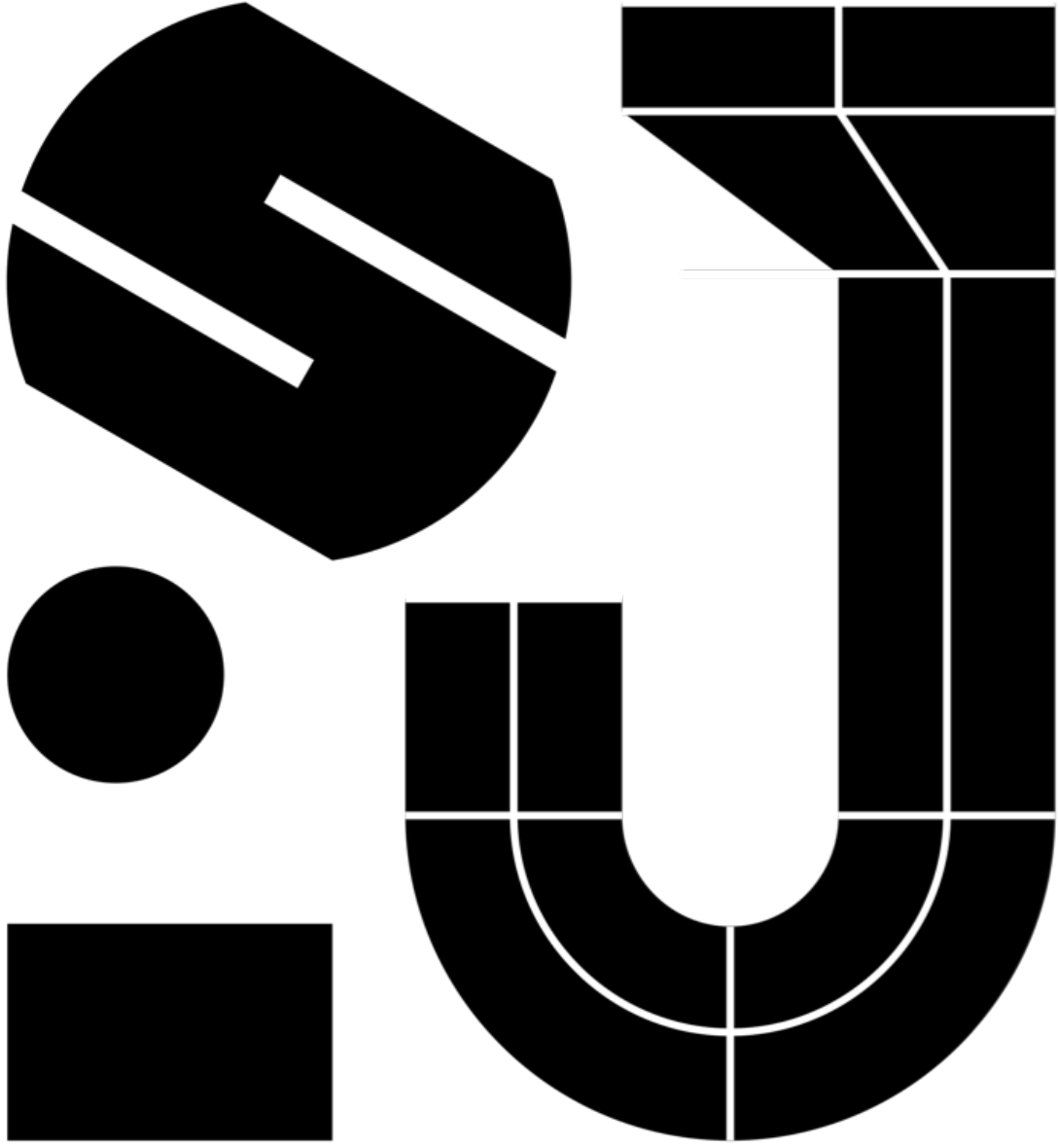
※非公開領域に指定された位置情報は、公益の側面から必要に応じて開示を求  
められる場合が考えられる。この様な場合には、印刷された標識の所有者  
と、トラッキングを受けた個人の両方に、トラッキングを行った目的と、ト  
ラッキングの範囲、位置や時間、情報にアクセスした組織の情報も含めて通  
知する事が、義務となると考えられる。

※非公開領域に指定された位置座標のトラッキングを行う事が望ましい事案の  
一例としては、救命に関わる事態が挙げられる。情報携帯端末の所有者が、  
命に関わる事態に陥った場合は、救命隊に、自分の位置を正確に知らせる為  
に、位置情報並びに、移動経路などのトラッキング情報を伝達できる事が必  
要となる。

※また、この逆に、公開とされている領域であっても、位置情報のトラッキン  
グは無差別に許されるべきものではない。個人が特定できるデータに関して  
は、その個人の同意が必要となるべきであり、その時々や場所により同意す  
る範囲をその個人が選べるべきである。例えば、自分の家の周囲1km範囲で  
の行動については、公共の道路上であってもトラッキングを禁止する事がで  
きるべきである。

※<領域の属性>の指定するべき範囲は、将来的な周囲の変化に対応できる様  
に、限定するべきである。つまり、QRコードが印刷された案内板の所有者が  
管理できる範囲内のみを公開または非公開と指定し、その他の領域は、判別  
不明な領域として扱うべきである。

※<領域の属性>のプロパティが含まれない場合は、公開か非公開か不明な判別  
不明の領域として扱われるものとする。



サブ領域:

プロパティ名	プロパティの名称	主な情報
Q	<サブ領域長方形のプロパティ>	サブ領域長方形
C	<サブ領域円形のプロパティ>	サブ領域円形
J	<サブ領域多接続形のプロパティ>	サブ領域多接続形
S	<サブ領域多頂点形のプロパティ>	サブ領域多頂点形
T	<サブ領域樹木形のプロパティ>	サブ領域樹木形
H	<サブ領域凹形のプロパティ>	サブ領域凹形

サブ領域は、対象の形状及び位置と向き指定するためのプロパティであり、LPSでは、サブ領域で指定された、階数のプロパティ、位置と向きをサブ領域は、重なった背景領域の属性を上書きする事ができる。

サブ領域は、表現したい形状をできる限り、少ないデータ量で表現する為に、形状毎に、プロパティやパラメータが別になっている。

サブ領域と、領域の属性の有効範囲:

サブ領域が、領域の基準点から、有効半径で示された領域から、はみ出した領域の属性は、常に不明として扱われる。

サブ領域を複数指定され、サブ領域同士が重なり合う場合は、後から指定された領域の属性が有効となる。

サブ領域の座標:

サブ領域の位置は、領域の基準点からの相対座標で指定する。

サブ領域中の位置、前の頂点位置と角度を基準点とする考え方は、プログラミング言語LOGOのTurtle Graphicsの考え方そのものである。

サブ領域の角度:

サブ領域の角度は、下記の置換表と計算式によりアルファベットでコード化したもので指定する。

アルファベット1文字で表記する場合:

$n =$  アルファベットの全体の文字数

サブ領域の形状:

サブ領域の形状は、長方形、接続形、円形、多頂点形の4種類を定義する。

基本の書式だけを下記に示す。

長方形:

**Q:**<長方形の角度><長方形の横幅>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

円形:

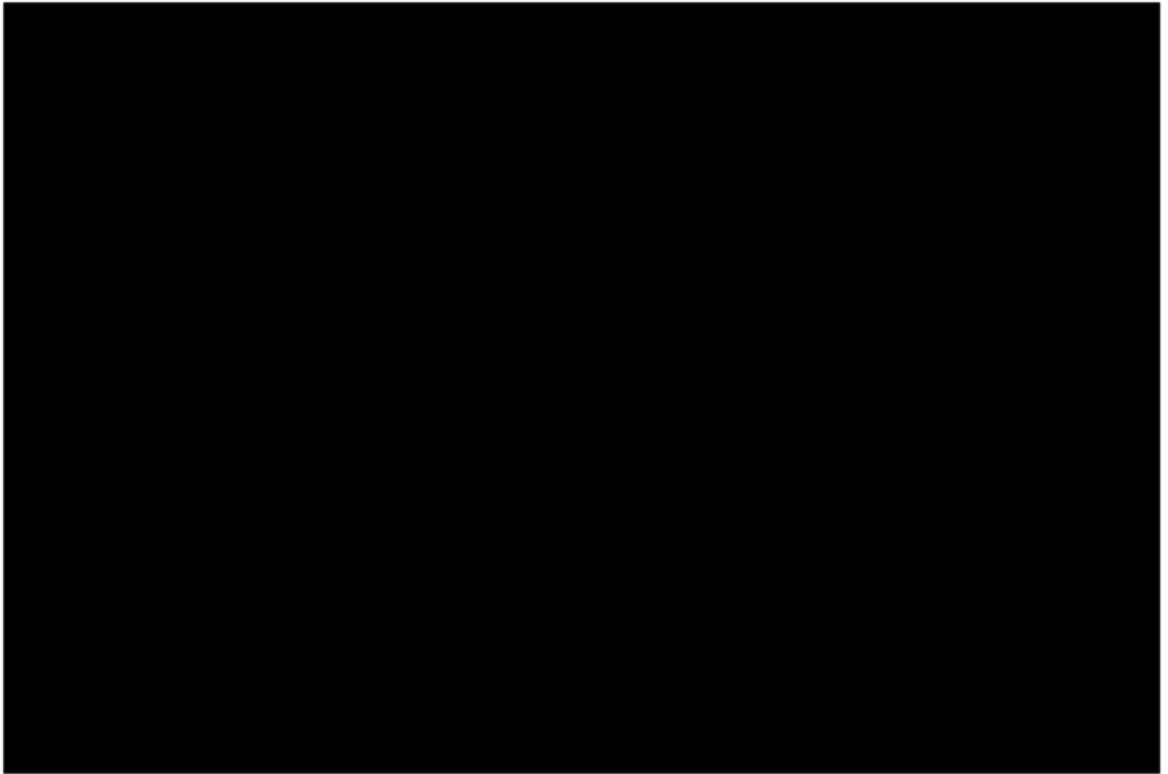
**C:**<扇の展開角1><円の厚み>.<扇の展開角2><楕円の長軸の半径>.<楕円の角度><円の中心点の向き><円の中心点に対する座標>**\$**<領域基準点からの座標と高さ>.

接続形:

**J:**<節の座標1><節の角度と曲率1><節の横幅1><補間の種類1><節の座標2><節の角度と曲率2><節の横幅2><補間の種類2>...<節の座標n><節の角度と曲率n><節の横幅n><補間の種類n>.

多頂点形:

**S:**<頂点の座標1><頂点の角度と曲率1><補間の種類1><頂点の座標2><頂点の角度と曲率2><補間の種類2>...<頂点の座標n><頂点の角度と曲率n><補間の種類n>**\$**<高さ>.





長方形:

プロパティ名	プロパティの名称	主な情報
Q	<サブ領域長方形のプロパティ>	サブ領域長方形

長方形はサブ領域を指定するためのプロパティの1つである。

サブ領域は、領域のプロパティの有効半径内で、形状を指定し、領域を分割する為のものである。

長方形の書式:

長方形は書式は1から16まであり、書式 1が基本形、書式 2から16までが、書式1からパラメータを省略した派生形である。

書式 1:

基本形式

**Q:**<長方形の角度><長方形の横幅>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

書式 2:

<長方形の角度>の省略形式

**Q:**<長方形の横幅>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

書式 3:

<長方形の横幅>の省略形式

**Q:**<長方形の角度>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

書式 4:

<長方形の角度>と<長方形の横幅>の省略形式

**Q:**.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

書式 5:

<長方形の奥行き>の省略形式

**Q:**<長方形の角度><長方形の横幅>**\$**<領域基準点からの座標と高さ>.

書式 6:

<長方形の角度>と<長方形の奥行き>の省略形式

**Q:**<長方形の横幅>\$<領域基準点からの座標と高さ>.

書式 7:

<長方形の横幅>と<長方形の奥行き>の省略形式

**Q:**<長方形の角度>\$<領域基準点からの座標と高さ>.

書式 8:

<長方形の角度>と<長方形の横幅>と<長方形の奥行き>の省略形式

**Q:**\$<領域基準点からの座標と高さ>.

書式 9:

<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度><長方形の横幅>.<長方形の奥行き>.

書式 10:

<長方形の角度>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の横幅>.<長方形の奥行き>.

書式 11:

<長方形の横幅>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度>-<長方形の奥行き>.

書式 12:

<長方形の角度>と<長方形の横幅>と<領域基準点からの座標と高さ>  
の省略形式

**Q:**-<長方形の奥行き>.

書式 13:

<長方形の奥行き>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度><長方形の横幅>.

書式 14:

<長方形の角度>と<長方形の奥行き>と<領域基準点からの座標と高さ>  
>の省略形式

**Q:**<長方形の横幅>.

書式 15:

<長方形の横幅>と<長方形の奥行き>と<領域基準点からの座標と高さ>  
>の省略形式

**Q:**<長方形の角度>.

書式 16:

<長方形の角度>と<長方形の横幅>と<長方形の奥行き>と<領域基準  
点からの座標と高さ>の省略形式

**Q:**.

長方形のパラメータ:

<長方形の角度>:

領域のプロパティで指定された<領域基準点の角度>に対する相対角度を指定する為の<3軸回転角度のパラメータ>である。

省略された場合は、相対角度は0度として扱われる。

角度のパラメータは、アルファベットで符号化されたものと、パーセントマーク(%)から始まる角度のスタック定数が利用される。

<長方形の横幅>:

長方形の横幅を指定するための、<長さのパラメータ>である。

省略された場合には、<領域基準点からの座標>から、<長方形の角度>に対し、直交する角度に引かれた線に対し生じる、<領域の有効半径>の円との2つの交点の距離が、<長方形の横幅>として、扱われる。

<長方形の奥行き>:

長方形の奥行きを指定するための、<長さのパラメータ>である。

省略された場合には、<領域基準点からの座標>から、<長方形の角度>が示す方向に向かって直線を引き、<領域の有効半径>の円との交点までの距離の値が、<長方形の奥行き>として扱われる。

<領域基準点からの座標と高さ>:

書式1:

<領域基準点からの座標>\$<高さ>

書式2:

<領域基準点からの座標>

書式3:

\$<高さ>

<領域基準点からの座標>:

<領域の基準点の位置座標>と<領域基準点の角度>の値で示された座標に対する、相対座標で長方形の基準点の位置を指定する<座標のパラメータ>である。

省略された場合には、長方形の基準点の位置は、<領域の基準点の位置座標>と一致する事になる。

(+)を前に付けた場合は、前のサブ領域の<領域基準点からの座標>からの差分で示す。前のサブ領域の<領域基準点からの座標>から差分がゼロの場合には(+)だけ指定すれば良い。

<高さ>:

<高さ>は、<長さのパラメータ>である。高さを指定した場合には、高さの制限がある事を示す。

※<領域基準点からの座標>の指定に対し、仮に、円座標系を利用して指定しても、円座標系に含まれている角度の値は、<領域基準点の角度>と<長方形の角度>の組み合わせで得られる長方形の基点の角度には影響を及ぼさない。

※LPSおよびGMSで実際に実装した場合には、QRコードがある方向と同じ方向を向いた長方形が存在する可能性が高くなるため、仮に、座標系で指定した角度を反映する仕様にした場合、<長方形の角度>のパラメータで角度の補正を行う可能性が高くなり、パラメータを省略出来ずにデータの量が増加する事が考えられる。

※LPSおよびGMSでは、仮に円座標系で、データ量が大きくなる切りの悪い角度が利用され、その角度を、<長方形の角度>として、利用したい場合には、角度のスタック定数をの仕組みにより、以前に利用した、データ量が大きくなる角度を、直後に再度指定する場合には、少ないデータ量で指定できる仕組みが利用できる為、この様な場合であっても、表現に、必要となるデータの量は抑える事ができる。

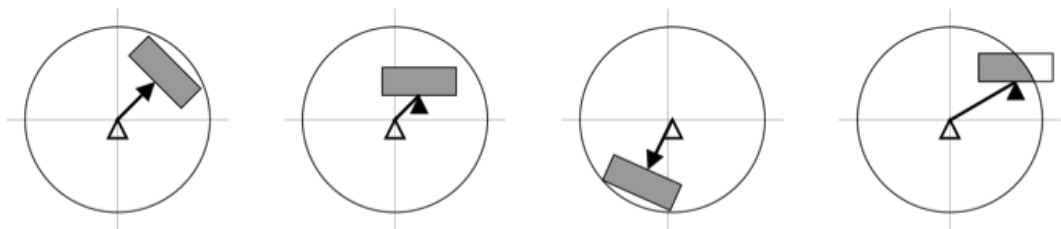
長方形の書式の詳細:

長方形は1から16の書式を持つ。

書式 1:

基本形式

**Q:**<長方形の角度><長方形の横幅>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.

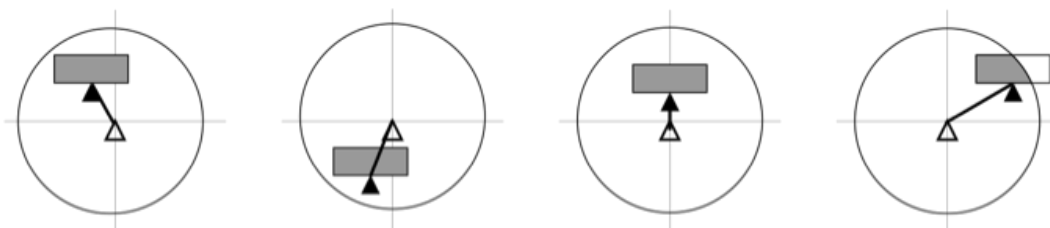


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 2:

<長方形の角度>の省略形式

**Q:**<長方形の横幅>.<長方形の奥行き>**\$**<領域基準点からの座標と高さ>.



横幅を持つ境界線。有効半径を超えた部分は無効となる。

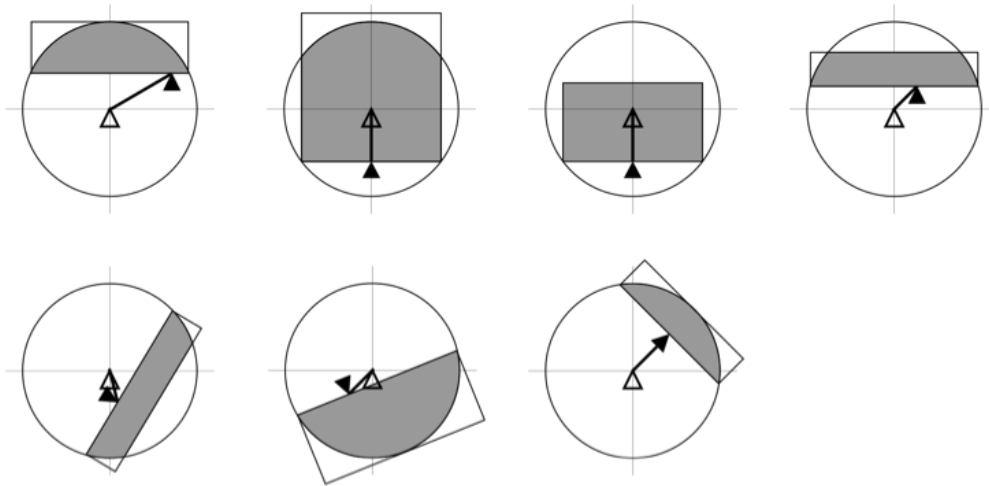
<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。



書式 3:

<長方形の横幅>の省略形式

**Q:**<長方形の角度>-<長方形の奥行き>\$<領域基準点からの座標と高さ>.

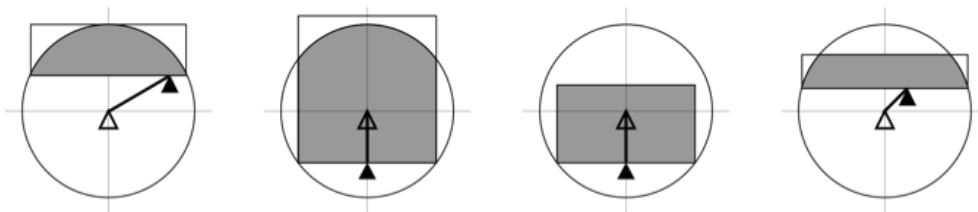


横幅を持つ境界線。有効半径を超えた部分は無効となる。

書式 4:

<長方形の角度>と<長方形の横幅>の省略形式

**Q:-**<長方形の奥行き>\$<領域基準点からの座標と高さ>.

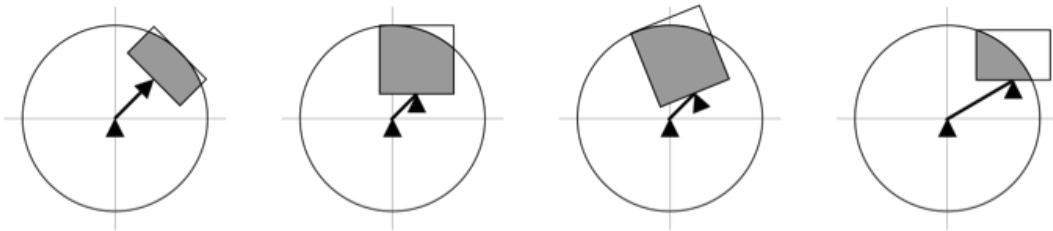


横幅を持つ境界線。有効半径を超えた部分は無効となる。

書式 5:

<長方形の奥行き>の省略形式

**Q:**<長方形の角度><長方形の横幅>**\$**<領域基準点からの座標と高さ>.

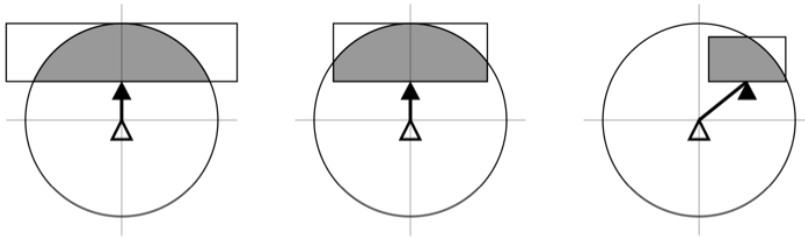


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 6:

<長方形の角度>と<長方形の奥行き>の省略形式

**Q:**<長方形の横幅>**\$**<領域基準点からの座標と高さ>.

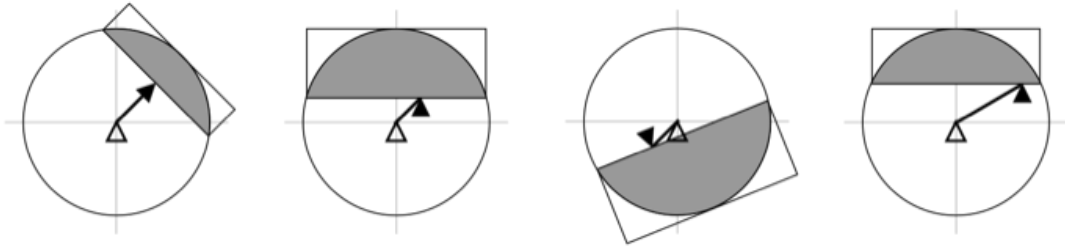


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 7:

<長方形の横幅>と<長方形の奥行き>の省略形式

**Q:**<長方形の角度>\$<領域基準点からの座標と高さ>.

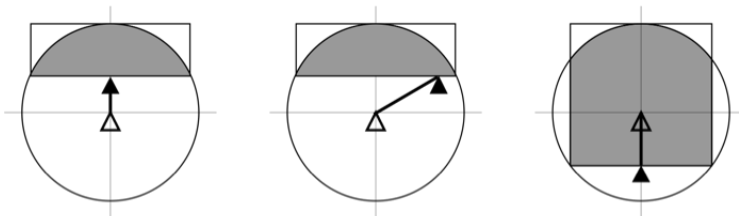


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 8:

<長方形の角度>と<長方形の横幅>と<長方形の奥行き>の省略形式

**Q:**\$<領域基準点からの座標と高さ>.

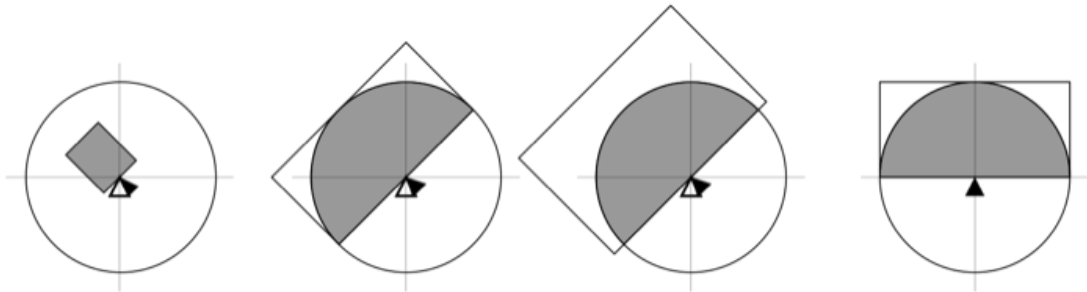


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 9:

<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度><長方形の横幅>.<長方形の奥行き>.

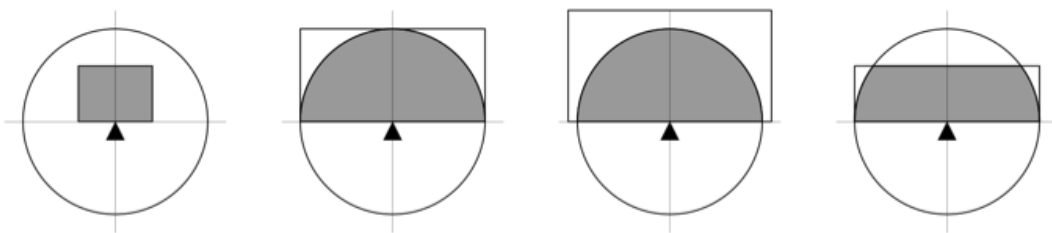


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 10:

<長方形の角度>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の横幅>.<長方形の奥行き>.

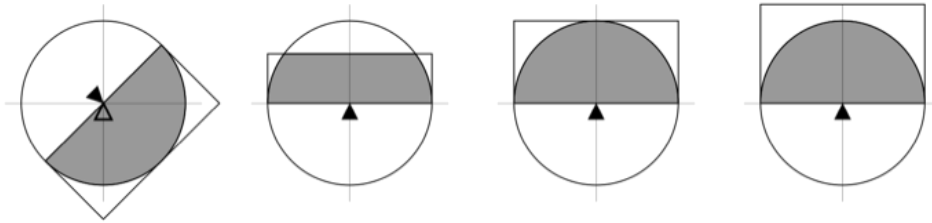


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 11:

<長方形の横幅>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度>-<長方形の奥行き>.

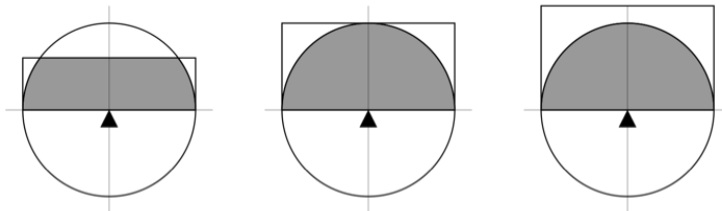


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 12:

<長方形の角度>と<長方形の横幅>と<領域基準点からの座標と高さ>の省略形式

**Q:**-<長方形の奥行き>.

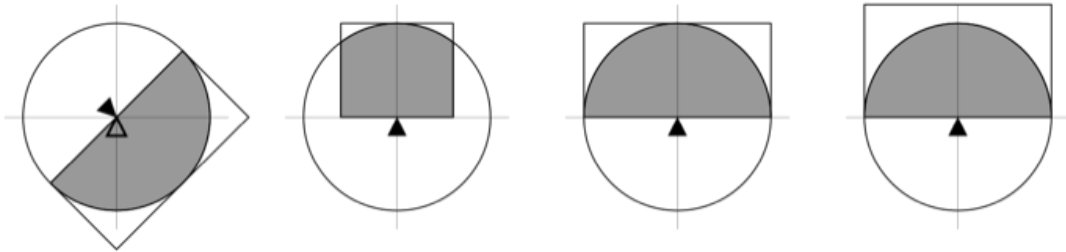


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 13:

<長方形の奥行き>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度><長方形の横幅>.

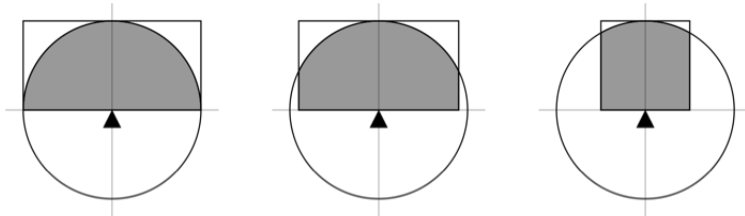


<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

書式 14:

<長方形の角度>と<長方形の奥行き>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の横幅>.



<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

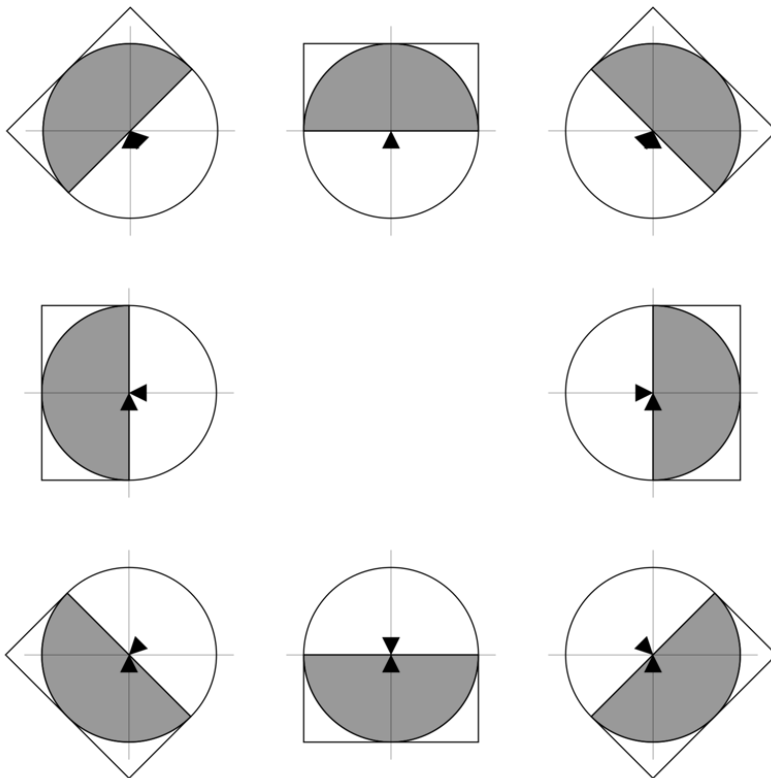
書式 15:

<長方形の横幅>と<長方形の奥行き>と<領域基準点からの座標と高さ>の省略形式

**Q:**<長方形の角度>.

パラメータ:

<長方形の角度>:角度のパラメータである。



<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

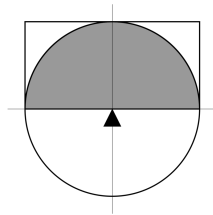


書式 16:

<長方形の角度>と<長方形の横幅>と<長方形の奥行き>と<領域基準点からの座標と高さ>の省略形式

**Q:**

<領域の基準点の位置座標>の位置を基点に、<領域基準点の角度>の角度で、<領域の有効半径>をそのまま利用して、境界を指定する。



「Q」の1文字で済むため、最もデータの量を減らす事ができる。

<領域の有効半径>を超えた部分は<領域の属性>無効となる。また、精度も保証できない。

<長方形の角度>:

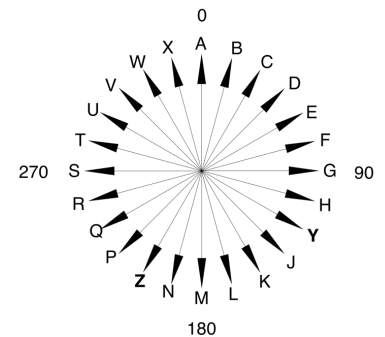
<長方形の角度>は、<角度のパラメータ>である。<長方形の角度>の項目が省略されている場合は下記のように扱うものとする。

長方形の角度は、サブ領域の角度がそのまま適用される。

<長方形の横幅>\*<長方形の奥行き>も省略される。

長方形の角度は、下記の置換表と計算式によりアルファベットでコード化したもので指定する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X



アルファベット1文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

例: A=0度、B=15度、D=45度、E=60度、G=90度、Y=120度、M=180度

アルファベット2文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

+360/24/24\*<2文字目の対応表で示される数値>

例: AB=0.625度、GQ=97.5+2.5=100度

1文字のAに対し2文字のAAはより高い有効精度の場合に使用される。

アルファベットn文字で表記する場合:

$$y = \sum_{k=1}^n \frac{x_k}{24^k}$$

$y$  = 角度

$x_k$  = アルファベットで表記された対応する数値

$k$  = アルファベットの左から数えた文字数

$n$  = アルファベットの全体の文字数

角度とは $2\pi$ を分割する値に対する比率であり、度数法では360等分した際の割合で示すものである。LPSでは360等分では無く、 $24^n$ 等分するという考えに基づいている。その為、 $n = 1$ の場合には24等分と精度が低くなるが、 $n = 2$ の場合には576等分となり、度数方で整数の範囲で表記する場合よりも精度が良くなる。

従って実用上、アルファベット1文字から2文字で十分な精度が得られる事が期待できるが、必要に応じて3文字以上で、表記しても良い。LPSを利用する携帯情報端末が、アルファベット3文字以上で書かれたデータを受け取った際の処理として、3桁以降を値として細かく計算するか、切り捨て処理を行うかは実装の目的に合わせて判断するものとする。

※符号化する際にアルファベットの大文字のIとOはアラビア数字の1と0との判読が困難となるため、それぞれ大文字のYとZに置き換える。

※アルファベットから数値に復号する際は、アルファベットの大文字のIは数値8、アルファベットの大文字のOは、数値14として扱うものとする。

※アラビア数字ではなく、アルファベットでコード化する理由は、より少ない文字数で多くの情報を入れる事を可能にし、精度と情報量のトレードオフを可能にする為である。

<長方形の横幅>:

<長方形の横幅>は、<長さのパラメータ>である。

<長方形の横幅>が省略されている場合は、下記の様に扱うものとする。

<長方形の横幅>は、<領域の有効半径>を満たす長さとする。

\*<長方形の奥行き>も省略される。

単位はメートルを基本とする。

メートルよりも細かな値を扱う必要がある場合には、ピリオドの代わりに、「M」を入れるものとする。キロメートル単位の値を扱う場合には「K」を入れるものとする。

例:

0.12mの場合: M12

※ 先頭の「0」は省略できる。

12mの場合: 12

※メートル精度の場合は「M」は、省略できる。

12.3mの場合: 12M3

123mの場合: 120

100mの場合: K1

※ 先頭の「0」は省略できる。

12 kmの場合: 12K

12.3 kmの場合: 12K3

<長方形の奥行き>:

<長方形の奥行き>は、<長さのパラメータ>である。

<長方形の奥行き>が省略されている場合は、下記の様に扱うものとする。

<長方形の奥行き>は、<領域の有効半径>を満たす長さとする。

単位はメートルを基本とする。

メートルよりも細かな値を扱う必要がある場合には、ピリオドの代わりに、

「M」を入れるものとする。キロメートル単位の値を扱う場合には「K」を入れるものとする。

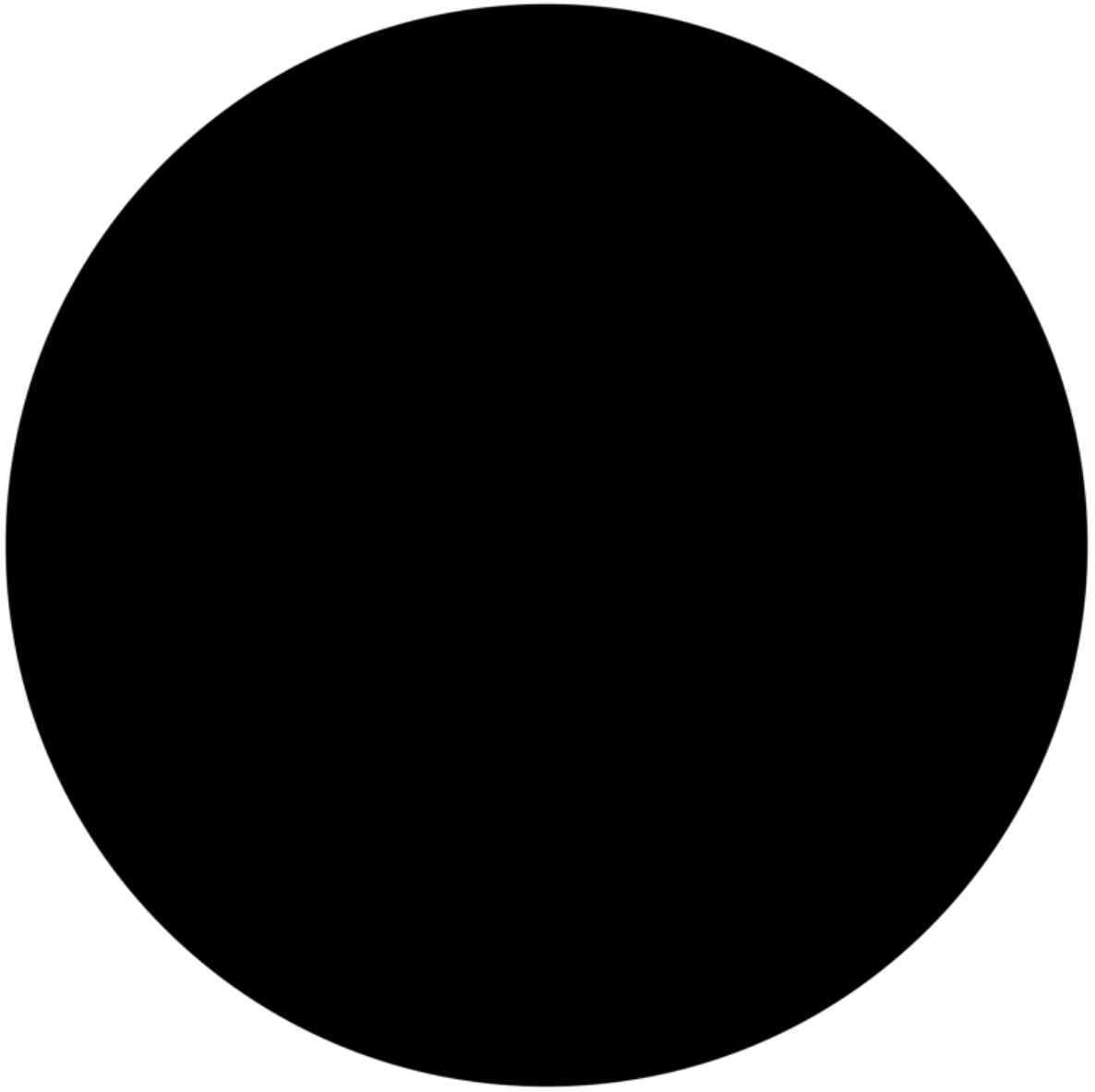
例:

12mの場合: 12

※メートル精度の場合は「M」は、省略できる。

12.3mの場合: 12M3

123mの場合: 120



円形:

プロパティ名	プロパティの名称	主な情報
C	<サブ領域円形のプロパティ>	サブ領域円形

円形は、円の形状、同心円の形状、扇の形状、楕円形を表現するためのサブ領域のプロパティである。

円形の書式概要:

書式 (楕円扇形基本形式)

**C:**<扇の展開角1><円の厚み>.<扇の展開角2><楕円の長軸の半径>.<楕円の角度><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

書式 (楕円扇形省略形式)

**C:**<扇の展開角1>.<楕円の長軸の半径>.

書式 (楕円同心円形基本形式)

**C:**<円の厚み>.<楕円の長軸の半径>.<楕円の角度><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

書式 (楕円同心円形省略形式)

**C:**<円の厚み>.<楕円の長軸の半径>.

書式 (楕円形基本形式)

**C:**<楕円の長軸の半径>.<楕円の角度><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

書式 (楕円形省略形式)

**C:**-<楕円の長軸の半径>.

書式 (扇形基本形式)

**C:**<扇の展開角1><円の厚み>-<扇の展開角2>-<円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

書式 (扇形省略形式)

**C:**<扇の展開角1>.

書式 (同心円形基本形式)

**C:**<円の厚み><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

書式 (同心円形省略形式)

**C:**<円の厚み>.

書式 (円形基本形式)

**C:**<円の中心点に対する角度><円の半径>\$<領域基準点からの座標と高さ>.

書式 (円形省略形式)

**C:**<円の厚み><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.



円形のパラメータの概要:

<円の中心点の向き>:

<領域基準点からの座標>が円の中心点に対し、内側なのか、外側なのかを示すパラメータである。

<円の中心点に対する座標>:

<座標系のパラメータ>である。

<座標系のパラメータ>は、<長さのパラメータ>と<角度のパラメータ>の組み合わせで構成されている。

<円の半径>:

<長さのパラメータ>である。

<円の中心点に対する角度>:

<角度のパラメータ>である。

<円の厚み>:

<長さのパラメータ>である。

<扇の展開角1>と<扇の展開角2>

<角度のパラメータ>である。

<楕円の角度>

<3軸回転角度のパラメータ>である。

<楕円の長軸方向の半径>

<長さのパラメータ>である。

<領域基準点からの座標と高さ>

<領域基準点からの座標>と<高さ>を組み合わせたパラメータである。

<領域基準点からの座標>:

<座標系のパラメータ>である。

(+)を前に付けた場合は、前のサブ領域の<領域基準点からの座標>からの差分で示す。前のサブ領域の<領域基準点からの座標>から差分がゼロの場合には(+)だけ指定すれば良い。

<座標系のパラメータ>は、<長さのパラメータ>と<角度のパラメータ>の組み合わせで構成されている。

<高さ>:

<長さのパラメータ>である。

円形のパラメータの詳細:

書式 (楕円扇形基本形式)

**C:**<扇の展開角1><円の厚み>.<扇の展開角2><楕円の長軸の半径>.<楕円の角度><円の中心点の向き><円の中心点に対する座標>\$<領域基準点からの座標と高さ>.

<円の中心点の向き>:

<領域基準点からの座標>が円の中心点に対し、内側なのか、外側なのかを示すパラメータである。

外側の場合には、プラスマーク(+)、内側の場合には、ドルマーク(\$)で表す。

指定を省略した場合:

外側を指定したものとして扱われる。

<円の中心点に対する座標>と<領域基準点からの座標>も省略される。

<円の中心点に対する座標>:

<座標系>のパラメータである。

<円の中心点に対する座標>のパラメータは、長さのパラメータと角度のパラメータの組み合わせで構成されている。<円の中心点に対する角度>と<円の半径>のパラメータに変換される。

<円の中心点に対する座標>のパラメータが省略された場合には、領域の<領域の基準点の位置座標><領域基準点の角度と有効半径>の情報から、<領域基準点の角度><領域の有効半径>の組み合わせで求められる。即ち、<領域の基準点の位置座標>から、<領域の有効半径>の円を書き、<領域の基準点の位置座標>から、<領域基準点の角度>の方向に引いた直線と円の交点の座標が用いられる。

<円の中心点に対する角度>:

<角度のパラメータ>である。

<円の中心点に対する座標>からx-z平面上のy軸の<角度のパラメータ>として求められる。

<円の半径>:

<長さのパラメータ>である。

<円の中心点に対する座標>からx-z平面上の<長さのパラメータ>として求められる。

<円座標系>をした場合には、角度と長さがそのまま反映される。<円筒座標系>を利用する場合も、同様に、高さの値は無視される。

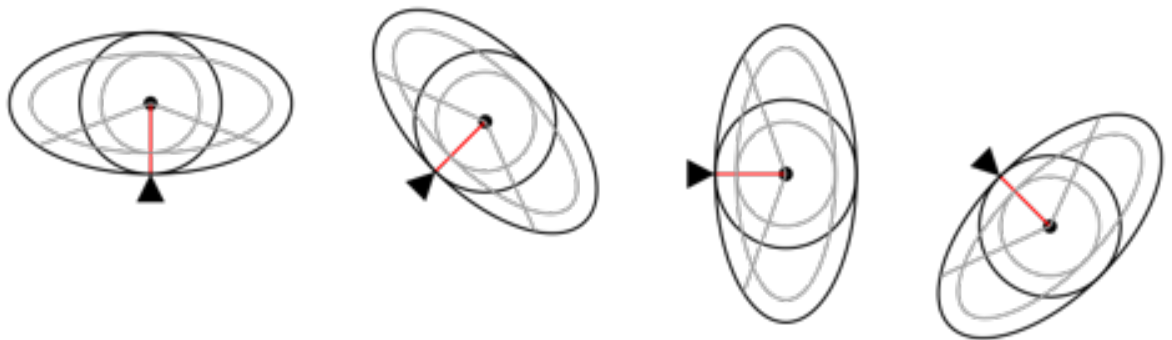
<円の中心点に対する角度>:

<円の中心点に対する座標>のパラメータから求められる角度のパラメータである。

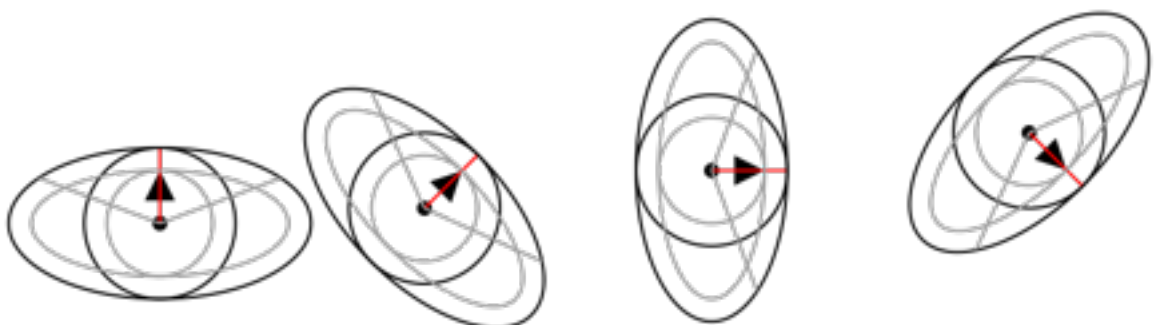
<領域基準点からの座標>を中心として、<領域基準点の角度>から見た、円の中心点の相対角度を指すパラメータである。

<円の中心点の向き>が外側の場合は、このパラメータの角度の方向に、円の中心点が存在する。内側の場合は、このパラメータの180度反対方向に、円の中心点が存在する。

<円の中心点の向き>が外側の際に0度、45度、90度、135度の例



<円の中心点の向き>が内側の際に0度、45度、90度、135度の例



<円の中心点に対する角度>のパラメータが省略されている場合には、相対角度に0度を指定したものとして解釈され、<領域基準点の角度>がそのまま反映される事となる。

<円の半径>:

長さのパラメータである。

<円の半径>のパラメータを省略した場合には、<領域の有効半径>の値が指定されたものとする。

円の内側から外側を見ている場合に、<領域基準点からの座標>は、<円の厚み>で指定された、内側の円周上に存在する事となる。

楕円の場合の規定:

<円の半径>のパラメータは、<楕円の長軸方向の半径>のパラメータが指定されている場合は、楕円の短軸方向の半径として解釈される。

<扇の展開角1>

角度のパラメータである。

<扇の展開角1>が省略されている場合:

<扇の展開角2>も省略され、両方ともに0度が指定されたものとする。

<扇の展開角1>と<扇の展開角2>の両方に対して0度を指定した場合には、扇形ではなく、円形を指定した事となる(円形は、扇形の特別な場合として考えるものとする)。

<扇の展開角1>のパラメータが指定されている場合:

<扇の展開角1>は、中心点から見て時計回りに、扇が展開されている角度を指定する。

<扇の展開角2>と合わせた角度が360度以上の値になる角度は指定する事はできない。

<扇の展開角2>のパラメータを省略した場合:

<扇の展開角2>には<扇の展開角1>のパラメータの値が適用されるため、<扇の展開角1>には、180度よりも小さな角度しか、指定する事ができない。

#### <円の厚み>

長さのパラメータである。

円の外側から内側方向への長さを指定する。

<円の厚み>のパラメータの値は0から<円の半径>までの正の値を取ることができる。

<円の厚み>のパラメータを省略した場合:

<円の厚み>に0を指定したものと扱われるものとする。

#### <扇の展開角2>

角度のパラメータである。

#### <楕円の長軸方向の半径>

長さのパラメータである。



<領域基準点からの座標と高さ>:

書式1:

<領域基準点からの座標>\$<高さ>

書式2:

<領域基準点からの座標>

書式3:

\$<高さ>

<領域基準点からの座標>:

<領域の基準点の位置座標>と<領域基準点の角度>の値で示された座標に対する、相対座標で長方形の基準点の位置を指定する<座標のパラメータ>である。

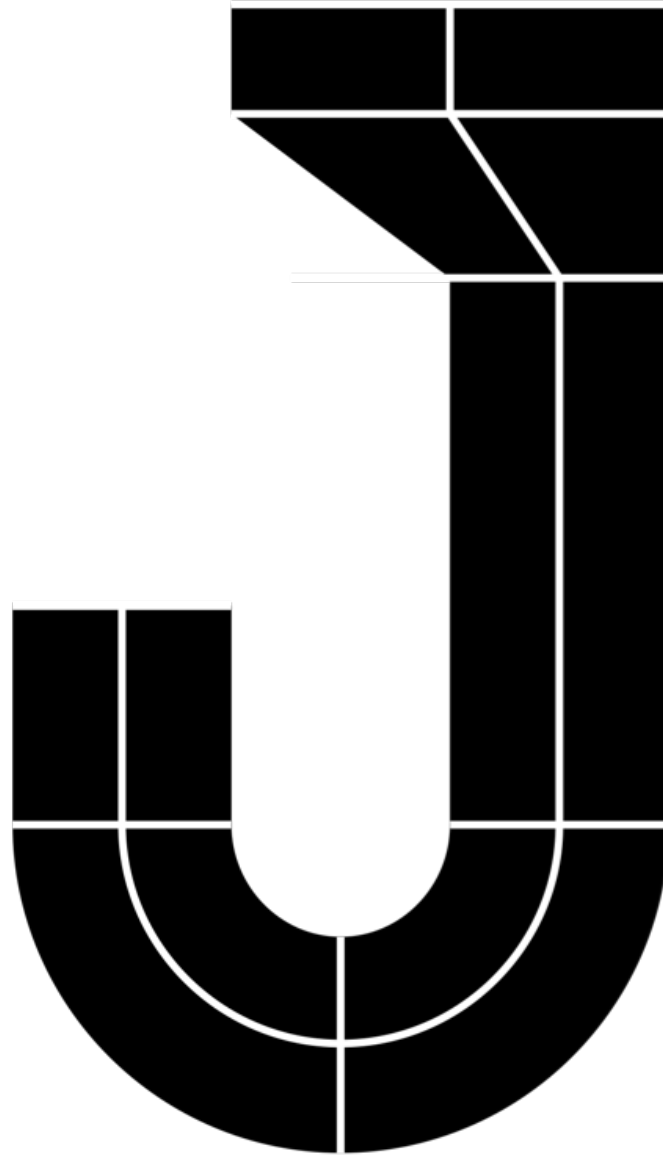
省略された場合には、円形の基準点の位置は、<領域の基準点の位置座標>と一致する事になる。

(+)を前に付けた場合は、前のサブ領域の<領域基準点からの座標>からの差分で示す。前のサブ領域の<領域基準点からの座標>から差分がゼロの場合には(+)だけ指定すれば良い。

<高さ>:

<高さ>は、<長さのパラメータ>である。高さを指定した場合には、高さの制限がある事を示す。乗り越える事を想定しているが、車椅子等の妨げになる場合など、高さが意味を持つ場合に指定する。柱や柵などは、高さに関係なく、進行を妨げる存在なので、この様なものに対しては、高さは省略できるものとする。

<サブ領域円形のプロパティ>では、円形、楕円形、扇形、2重円、これらの組み合わせだけを取扱い、その他の円錐曲線については、このプロパティでは、取り扱わないが、<サブ領域多頂点形のプロパティ>を利用する事で、楕円形や、扇形を近似したり、その他の曲線を近似する事もできる。



接続形:

プロパティ名	プロパティの名称	主な情報
J	<サブ領域多接続形のプロパティ>	サブ領域多接続形

接続形は、連続した曲がりくねった道、曲がり階段などを表現するための形状である。従って、接続という言葉は、連節という文字に置き換えてもある程度意味が通じるものとなっている。このサブ領域は、道以外の形状を表現する為にも利用する事ができる。

書式:

**J:**<節の座標1><節の角度と曲率1><節の横幅1><補間の種類1><節の座標2><節の角度と曲率2><節の横幅2><補間の種類2>...<節の座標n><節の角度と曲率n><節の横幅n><補間の種類n>.

<節の座標<sub>n</sub>>:

節の位置を示す<座標のパラメータ>である。

節の個数がn個の場合:

1個目とn個目は、誤差の累積を防止する為に、<領域の基準点の位置座標>から、<領域基準点の角度>で指定された位置を基点とした、相対座標で示すものとする。

これに対し、その間の<節の座標>は、情報量を削減する為に前の節の座標からの相対座標で指定する。

<節の座標1>は、<領域の基準点の位置座標>との相対座標が、0の場合には省略する事ができる。

また、間の<節の座標>も、前の<節の座標>との相対座標が、0  
の場合には省略する事ができる。

<節の角度と曲率<sub>n</sub>>:

<節の角度と曲率<sub>n</sub>>のパラメータは、下記に示す8つの書式を持つ

書式1:

-<yaw角度とyaw曲率>\*<pitch角度とpitch曲率>\*<roll角度とroll曲率>

書式2:

-\*<pitch角度とpitch曲率>\*<roll角度とroll曲率>

書式3:

-<yaw角度とyaw曲率>\*\*<roll角度とroll曲率>

書式4:

-<yaw角度とyaw曲率>\*<pitch角度とpitch曲率>

書式5:

-<yaw角度とyaw曲率>

書式6:

-\*<pitch角度とpitch曲率>

書式7:

-\*\*<roll角度とroll曲率>

書式8:

全て省略し、何も記載しない。

<yaw角度とyaw曲率>の書式:

書式1:

<yaw角度><yaw曲率>

書式2:

<yaw角度>

前の節と同じ<yaw曲率>の場合に指定する書式である。

書式3:

<yaw曲率>

前の節と同じ<yaw角度>の場合に指定する書式である。

<yaw角度>:

<角度のパラメータ>である。

<領域基準点の角度>に対する相対角度である。

0度～180度未満: 右カーブ

180度～360度未満: 左カーブ

節の座標に重ねて、この角度の方向に直線を引くとは、<節の横幅>を示す直線と重なる事になる。

<yaw曲率>:

<長さのパラメータ>である。

yaw軸 (Y軸)に対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。

<pitch角度とpitch曲率>の書式:

書式1:

<pitch角度><pitch曲率>

書式2:

<pitch角度>

前の節と同じ<pitch曲率>の場合に指定する書式である。

書式3:

<pitch曲率>

前の節と同じ<pitch角度>の場合に指定する書式である。

<pitch角度>:

<角度のパラメータ>である。

0度～180度未満: 上向のカーブ

180度～360度未満: 下向のカーブ

<pitch曲率>:

<長さのパラメータ>である。

pitch軸対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。

<roll角度とroll曲率>の書式:

書式1:

<roll角度><roll曲率>

書式2:

<roll角度>

前の節と同じ<roll曲率>の場合に指定する書式である。

書式3:

<roll曲率>

前の節と同じ<roll曲率>の場合に指定する書式である。

<roll角度>:

<角度のパラメータ>である。

進行方向に対するroll軸の角度

0度～180度未満: 時計回りの傾きのカーブ

180度～360度未満: 反時計回りの傾きのカーブ

<roll曲率>:

<長さのパラメータ>である。

roll軸に対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。



<節の横幅<sub>n</sub>>:

書式1

+<長さのパラメータ>

<長さのパラメータ>により(道の)横幅を示す。

書式2

省略し、何も記載しない。

省略された場合には、前の節の値が適用される。

<節の横幅1>が省略された場合には、<領域の有効半径>の値がそのまま適用される。

<補間の種類<sub>n</sub>>:

節と節の間の補間方法を2文字の文字列で指定する。

yaw軸、pitch軸、roll軸それぞれで、補間方法を選択できる。

補間の種類

対応する文字列	yaw軸	pitch軸	roll軸
\$0	線形補間	線形補間	線形補間
\$1	線形補間	線形補間	前の節の値
\$2	線形補間	前の節の値	線形補間
\$3	線形補間	前の節の値	前の節の値
\$4	前の節の値	線形補間	線形補間
\$5	前の節の値	線形補間	前の節の値
\$6	前の節の値	前の節の値	線形補間
\$7	前の節の値	前の節の値	前の節の値

線形補間の場合は、前の節の値と次の節の値の両方の値から間の値を埋める事ができる。

前の節の値で補間の場合には、次の節の座標まで、前の節の座標の値を適用する。

<節の角度と曲率 $n$ >は、直線の場合には省略する事ができる。



多頂点形:

プロパティ名	プロパティの名称	主な情報
S	<サブ領域多頂点形のプロパティ>	サブ領域多頂点形

サブ領域多頂点形は、多角形などの複雑な形状を表現する為のサブ領域である。

多頂点形:

書式

**S:**<頂点の座標1><頂点の角度と曲率1><補間の種類1><頂点の座標  
2><頂点の角度と曲率2><補間の種類2>...<頂点の座標n><頂点の角度  
と曲率n><補間の種類n><多頂点形の角度と高さ>.

<頂点の座標 $n$ >:

頂点の位置を示す<座標のパラメータ>である。

頂点の個数が $n$ 個の場合:

1個目は、<領域の基準点の位置座標>から、<領域基準点の角度>に<多頂点形の角度>を加えて指定された位置を基点とした、  
相対座標で示すものとする。

これに対し、その他の<頂点の座標>は、情報量を削減する為に  
前の<頂点の座標>からの相対座標で指定する。

<頂点の座標 $1$ >は、<領域の基準点の位置座標>との相対座標  
が、0の場合には省略する事ができる。

また、間の<頂点の座標>も、前の<頂点の座標>との相対座標  
が、0の場合には省略する事ができる。

<頂点の座標 $1$ >については、先頭にプラス(+)  
マークを付ける事で、前の<サブ領域>の最後に指定した座標からの相対座標で指定する事もできる。また、前の<サブ領域>の最後に指定した座標からの相対座標との差がゼロの場合には、プラス(+)  
マークだけを記述するだけで良い。

<頂点の角度と曲率<sub>n</sub>>:

<頂点の角度と曲率<sub>n</sub>>のパラメータは、下記に示す8つの書式を持つ

書式1:

-<yaw角度とyaw曲率>\*<pitch角度とpitch曲率>\*<roll角度とroll曲率>

書式2:

-\*<pitch角度とpitch曲率>\*<roll角度とroll曲率>

書式3:

-<yaw角度とyaw曲率>\*\*<roll角度とroll曲率>

書式4:

-<yaw角度とyaw曲率>\*<pitch角度とpitch曲率>

書式5:

-<yaw角度とyaw曲率>

書式6:

-\*<pitch角度とpitch曲率>

書式7:

-\*\*<roll角度とroll曲率>

書式8:

全て省略し、何も記載しない。

<yaw角度とyaw曲率>の書式:

書式1:

<yaw角度><yaw曲率>

書式2:

<yaw角度>

前の頂点と同じ<yaw曲率>の場合に指定する書式である。

書式3:

<yaw曲率>

前の頂点と同じ<yaw角度>の場合に指定する書式である。

<yaw角度>:

<角度のパラメータ>である。

<領域基準点の角度>に対する相対角度である。

0度～180度未満: 右カーブ

180度～360度未満: 左カーブ

<yaw曲率>:

<長さのパラメータ>である。

yaw軸 (Y軸)に対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。

<pitch角度とpitch曲率>の書式:

書式1:

<pitch角度><pitch曲率>

書式2:

<pitch角度>

前の頂点と同じ<pitch曲率>の場合に指定する書式である。

書式3:

<pitch曲率>

前の頂点と同じ<pitch角度>の場合に指定する書式である。

<pitch角度>:

<角度のパラメータ>である。

0度～180度未満: 上向のカーブ

180度～360度未満: 下向のカーブ

<pitch曲率>:

<長さのパラメータ>である。

pitch軸対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。



<roll角度とroll曲率>の書式:

書式1:

<roll角度><roll曲率>

書式2:

<roll角度>

前の頂点と同じ<roll曲率>の場合に指定する書式である。

書式3:

<roll曲率>

前の頂点と同じ<roll曲率>の場合に指定する書式である。

<roll角度>:

<角度のパラメータ>である。

進行方向に対するroll軸の角度

0度～180度未満: 時計回りの傾きのカーブ

180度～360度未満: 反時計回りの傾きのカーブ

<roll曲率>:

<長さのパラメータ>である。

roll軸に対するカーブの半径を示す数値である。

0の場合は、直線である事を示す。

<補間の種類<sub>n</sub>>:

節と節の間の補間方法を2文字の文字列で指定する。

yaw軸、pitch軸、roll軸それぞれで、補間方法を選択できる。

補間の種類-1

対応する文字列	yaw軸	pitch軸	roll軸
\$0	線形補間	線形補間	線形補間
\$1	線形補間	線形補間	前の節の値
\$2	線形補間	前の節の値	線形補間
\$3	線形補間	前の節の値	前の節の値
\$4	前の節の値	線形補間	線形補間
\$5	前の節の値	線形補間	前の節の値
\$6	前の節の値	前の節の値	線形補間
\$7	前の節の値	前の節の値	前の節の値

<多頂点形の角度と高さ>の書式:

書式1

+<多頂点形の角度><多頂点形の高さ>

書式2

+<多頂点形の角度>

書式3

+<多頂点形の高さ>

書式4

全てを省略

<多頂点形の角度>:

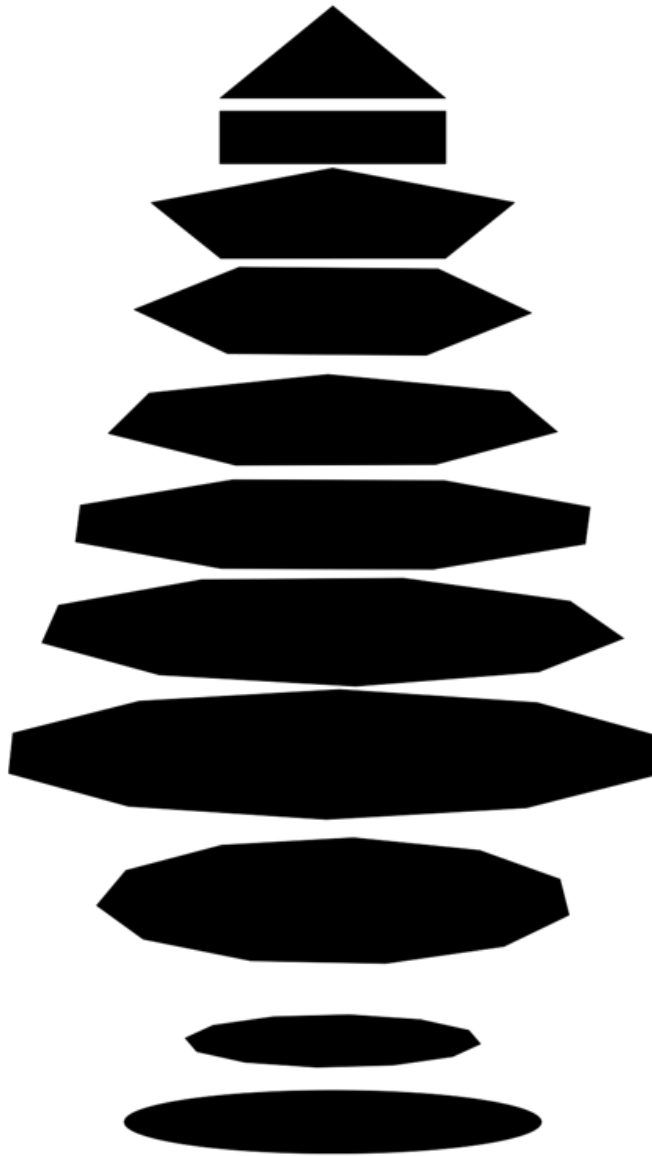
<多頂点形の角度>は<領域基準点の角度>に対する相対角度であり、<3軸回転角度のパラメータ>である。<多頂点形の角度>のパラメータが省略された場合には、<多頂点形の角度>に対し、0度を設定した事になり、<領域基準点の角度>がそのまま適用される事になる。

<多頂点形の高さ>:

<多頂点形の高さ>は、<長さのパラメータ>である。

乗り越える事を想定しているが、車椅子等の妨げになる場合など、高さが意味を持つ場合に指定する。

柱や柵などは、高さに関係なく、進行を妨げる存在なので、この様なものに対しては、高さは省略できるものとする。



樹木形:

プロパティ名	プロパティの名称	主な情報
T	<サブ領域樹木形のプロパティ>	サブ領域樹木形

樹木や、ドア、取手などの連続した、立体形状を表現する為のサブ領域である。

#### 書式1

T:<形状1>-<3軸回転角1><距離1>+<形状2>-<3軸回転角2><距離2>+...+<形状n-1>-<3軸回転角n-1><距離n-1>+<形状n>-<3軸回転角n>\$<領域基準点からの座標>.

#### 書式2

T:<形状1>-<3軸回転角1><距離1>+<形状2>-<3軸回転角2><距離2>+...+<形状n-1>-<3軸回転角n-1><距離n-1>+<形状n>-<3軸回転角n>.

樹木形は、中心点を基準に座標が指定される。

書式1で指定する<領域基準点からの座標>は、樹木形の中心点を指定するものである。

書式2では、<領域基準点からの座標>が省略されているため、中心点は、<領域の基準点の位置座標>と一致する。

<形状x>の書式:

xは1からnまでとなる。

書式1

<形状指定子><幅>

書式2

<形状指定子><横幅>\*<縦幅>

書式3

<形状指定子>\*<縦幅>

書式4

<形状指定子>

<横幅>の書式:

<長さのパラメータ>

<縦幅>の書式:

<長さのパラメータ>

縦横比同一の場合には、書式1を利用する。

横幅と縦幅を変更したい場合には、書式2を

横幅がそのまま、縦幅に適用される。

縦横比が異なる場合には、書式1~4を利用する。

<形状指定子>の書式:

下記の一覧表の符号が適用される。

符号	#	形状	補間	符号	#	形状	補間
A	1	球	直線	M	13	球	曲線
B	2	円	直線	N	14	円	曲線
C	3	三角形	直線	Z	15	三角形	曲線
D	4	4角形	直線	P	16	4角形	曲線
E	5	5角形	直線	Q	17	5角形	曲線
F	6	6角形	直線	R	18	6角形	曲線
G	7	7角形	直線	S	19	7角形	曲線
H	8	8角形	直線	T	20	8角形	曲線
Y	9	9角形	直線	U	21	9角形	曲線
J	10	10角形	直線	V	22	10角形	曲線
K	11	11角形	直線	W	23	11角形	曲線
L	12	12角形	直線	X	24	12角形	曲線

<形状x>を省略した場合は<形状x-1>の形状が維持される。

<形状1>で、<形状指定子>を省略した場合は、符号「A」を選択した事となる。

三角形から12角形までは、辺側が手前、頂点側が奥となる。

補間が曲線の場合は、次の形状までの間を、<3軸回転角x>に従って、曲線で補間を行う。

補間が直線の場合は、次の形状までの間を直線で補間を行う。

<3軸回転角 $x$ >の書式:

$x$ は1から $n$ までとなる。

<3軸回転角度のパラメータ>が適用される。

<距離 $x$ >の書式:

$x$ は1から $n-1$ までとなる。

<長さのパラメータ>が適用される。

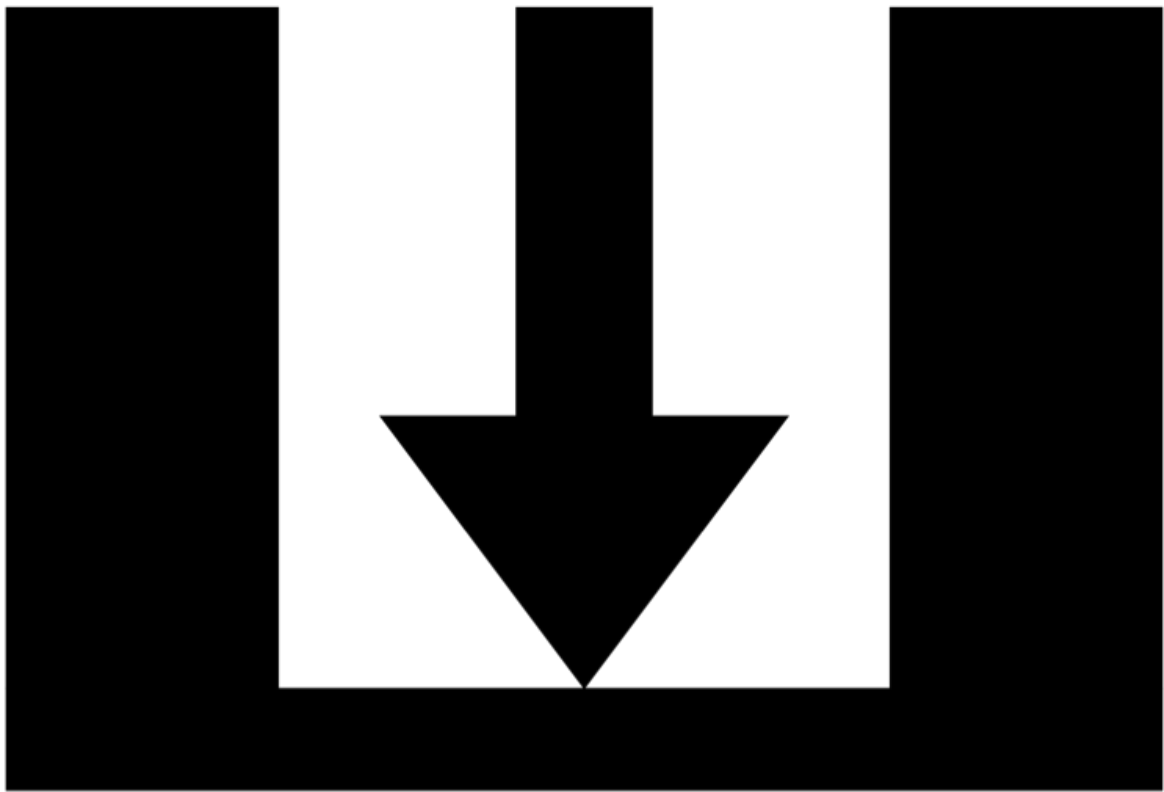
<領域基準点からの座標>:

<領域の基準点の位置座標>と<領域基準点の角度>の値で示された座標に対する、相対座標で長方形の基準点の位置を指定する<座標のパラメータ>である。

省略された場合には、長方形の基準点の位置は、<領域の基準点の位置座標>と一致する事になる。

(+)を前に付けた場合は、前のサブ領域の<領域基準点からの座標>からの差分で示す。前のサブ領域の<領域基準点からの座標>から差分がゼロの場合には(+)だけ指定すれば良い。





凹形:

プロパティ名	プロパティの名称	主な情報
H	<サブ領域凹形のプロパティ>	サブ領域凹形

凹みや穴を表現するためのサブ領域である。

書式1

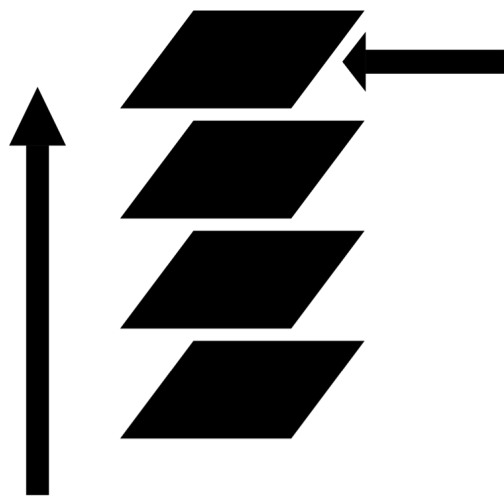
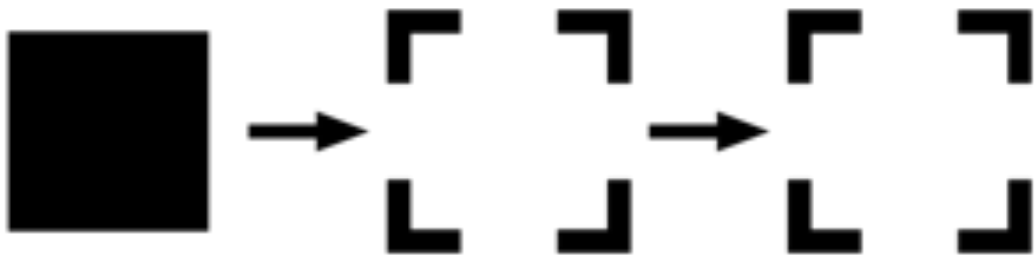
H:<形状1>-<3軸回転角1><距離1>+<形状2>-<3軸回転角2><距離2>+...+<形状n-1>-<3軸回転角n-1><距離n-1>+<形状n>-<3軸回転角n>\$<領域基準点からの座標>.

書式2

H:<形状1>-<3軸回転角1><距離1>+<形状2>-<3軸回転角2><距離2>+...+<形状n-1>-<3軸回転角n-1><距離n-1>+<形状n>-<3軸回転角n>.

凹形のパラメータは、樹木形と同じパラメータを持つが、その役割は、くり抜く事である。

他のサブ領域は、論理和であり、追加だけを行うが、凹形は、削る事だけが可能である。他のサブ領域で作成されたボクセルの一部を削りだし、穴を開ける事を目的としている。



繰り返しの指定:

プロパティ名	プロパティの名称	主な情報
R	<繰り返し指定のプロパティ>	サブ領域の繰り返し指定

<繰り返し指定のプロパティ>は、先に指定されたサブ領域を繰り返し (repeat)、複製す事と、サブ領域間の、関係性(relationship)を指定する為のプロパティである。

このプロパティにより、同じ形状のサブ領域が複数存在する際に、記述に必要となるデータ量を削減する事や、どのサブ領域がどのように接続されているかを指定する事ができる。

サブ領域を指定すると、サブ領域のスタックに積まれていく、サブ領域をスタック番号で指定する場合には、最後に積まれサブ領域が0となる。

<繰り返し指定のプロパティ>により、複製された、サブ領域も、後続の<繰り返し指定のプロパティ>による複製元のサブ領域の1つとなる。

書式1:

**R:**<オフセット1>\$<繰り返し回数と参照するサブ領域の範囲>+<オフセット2><リンクとスケールの指定>.

書式2:

**R:**<オフセット1>\$<繰り返し回数と参照するサブ領域の範囲><リンクとスケールの指定>.

書式3:

**R:**<オフセット1><リンクとスケールの指定>.

書式4:

**R:**\$<繰り返し回数と参照するサブ領域の範囲><リンクとスケールの指定>.

書式5:

**R:**<リンクとスケールの指定>.

書式1:

**R:**<オフセット1>\$<繰り返し回数と参照するサブ領域の範囲>+<オフセット2><リンクの指定>.

書式1は、複数回繰り返しを指定する場合のための書式である。

複製されたサブ領域の座標は以下の式により求められる。

初回の座標

=<オフセット1>のオフセット量  
+<領域の基準点の位置座標>の座標

繰り返し部分座標

=<オフセット1>のオフセット量  
+ <繰り返し回数> の繰り返し回数 \* <オフセット2>のオフ  
セット量  
+<領域の基準点の位置座標>の座標

※<領域の基準点の位置座標>は、<領域のプロパティ>のパラメータである。

書式2:

**R:**<オフセット1>\$<繰り返し回数と参照するサブ領域の範囲><リンクの指定>.

書式2は、複数回繰り返しを指定する場合のための書式である。

複製されたサブ領域の座標は以下の式により求められる。

初回の座標

$$\begin{aligned} &= \text{<オフセット1>のオフセット量} \\ &\quad + \text{<領域の基準点の位置座標>の座標} \end{aligned}$$

繰り返し部分座標

$$\begin{aligned} &= \text{<オフセット1>のオフセット量} \\ &\quad + \text{<繰り返し回数>の繰り返し回数} * \text{<オフセット1>のオフ} \\ &\quad \text{セット量} \\ &\quad + \text{<領域の基準点の位置座標>の座標} \end{aligned}$$

※ <領域の基準点の位置座標>は、<領域のプロパティ>のパラメータである。

書式3:

**R:**<オフセット1><リンクの指定>.

書式3は、繰り返しを行わない場合の指定である。

直前に指定されたサブ領域を下記に示す座標に1つだけ複製する。

複製先の座標

=<オフセット1>のオフセット量

+<領域の基準点の位置座標>の座標



書式4:

**R:\$**<繰り返し回数と参照するサブ領域の範囲><リンクの指定>.

書式4は、<繰り返し回数と参照するサブ領域の範囲>に対し、<リンクの指定>を行う為の形式である。

書式4では、<サブ領域>の複製は行わないものとし、<繰り返し回数と参照するサブ領域の範囲>で、<繰り返し回数>は、0、または、省略する必要がある。

書式5:

**R:**<リンクの指定>.

書式5は、リンクの指定だけを行う場合の指定である。

直前に指定されたサブ領域とこの後指定されるサブ領域との関係をお指定するためのものである。

書式5では、<サブ領域>の複製は行わないものとする。

<オフセット>の書式:

書式1:

<相対座標>+<回転角度>

書式2:

<相対座標>

書式3:

+<回転角度>

<相対座標>:

<座標のパラメータ>

<回転角度>:

<3軸回転角度のパラメータ>

<繰り返し回数と参照するサブ領域の範囲>の書式:

書式1

<繰り返し回数><参照するサブ領域の範囲1><参照するサブ領域の範囲2>... <参照するサブ領域の範囲n>

書式2

<繰り返し回数>

書式3

<参照するサブ領域の範囲1><参照するサブ領域の範囲2>... <参照するサブ領域の範囲n>

<繰り返し回数>:

アラビア数字を使用し10進数で参照する。

省略した場合は0回の繰り返しとなる。

<参照するサブ領域の範囲n>の書式:

<参照するサブ領域のスタック番号n><参照するサブ領域の個数n>

<参照するサブ領域のスタック番号n>:

アルファベットを利用した24進数で、参照するサブ領域のスタック番号を指定する。

<参照するサブ領域の個数n>:

アラビア数字を使用し10進数で参照するサブ領域の個数を指定する。

アラビア数字の値+1個の個数のサブ領域を参照する。

サブ領域は、最後に指定されたものから順番に参照される。

参照する個数が1個の場合はこのパラメータを省略することができる。

基点は、一番最初に指定された<参照するサブ領域のスタック番号n>とする。

<繰り返し回数>はアルファベット、<参照するサブ領域の個数>はアラビア数字で、利用しているもじの種類が異なる為、並べて記述しても明確に区別する事ができるので、間に区切り文字は入れないものとする。

<リンクとスケールの指定>の書式:

書式1

-<リンクの種類><スケールの指定>

基本形では、<リンクの種類>と<スケール>の両方を指定する事ができる。

書式2

-<リンクの種類>

基本形から<スケール>を省略したもの。

この場合、スケールは1:1となる。

書式3

-<スケールの指定>

基本形から<リンクの種類>を省略したもの。リンクの種類は、直前の<リンクの指定>が有効となる。

書式4

全て省略

リンクの種類は、直前の<リンクの指定>が有効となる。

スケールは1:1となる。

<リンクの種類>:

<リンクの種類>は、<リンクの種類>の一覧表の符号で指定する。

リンクには、トポロジー的なリンクと、移動変形がある。

トポロジー的なリンクでは、サブ領域間が、どの様に接続されているのかを指定する事ができる。

無し 繋がっていない

連続 直線上に繋がっている

分岐 放射状に繋がっている

合流 分かれているものが一点に繋がる

分岐と合流の違いは、向きにある。

分岐は、前方から後方のサブ領域には繋がっているが後方のサブ領域自身は、接続されない。

合流は、前方の分かっていたサブ領域どうしを接続する事ができる。

移動変形は、扉やシャッター、浮き橋の様に、移動するものや、柳の様に風などの影響で大きく変更するものが、占める場所を指定する事を目的としたものである。

<リンクの種類>の一覧表

符号	基点	繰り返しの前との関係	繰り返しに続く外との関係	属性	説明
A	無し	無し	無し	上書き	単純複製
B	コピー元	連続	連続	上書き	コピー元から延長
C	コピー元	連続	無し	上書き	コピー元から延長
D	繰り返しの先頭	連続	連続	上書き	繰り返し先頭から延長
E	繰り返しの先頭	連続	無し	上書き	繰り返し先頭から延長
F	コピー元	分岐先	分岐先	上書き	コピー元からの放射
G	コピー元	分岐先	無し	上書き	コピー元からの放射
H	繰り返しの先頭	分岐先	分岐先	上書き	繰り返し先頭からの放射
Y	繰り返しの先頭	分岐先	無し	上書き	繰り返し先頭からの放射
J	合流	-	無し	上書き	合流
K	合流	-	連続	上書き	合流
L	合流	-	分岐先	上書き	合流
M	無し	無し	無し	維持	単純複製
N	コピー元	連続	連続	維持	コピー元から延長
Z	コピー元	連続	無し	維持	コピー元から延長
P	繰り返しの先頭	連続	連続	維持	繰り返し先頭から延長
Q	繰り返しの先頭	連続	無し	維持	繰り返し先頭から延長
R	コピー元	分岐先	分岐先	維持	コピー元からの放射
S	コピー元	分岐先	無し	維持	コピー元からの放射
T	繰り返しの先頭	分岐先	分岐先	維持	繰り返し先頭からの放射
U	繰り返しの先頭	分岐先	無し	維持	繰り返し先頭からの放射
V	合流	-	無し	維持	合流
W	合流	-	連続	維持	合流
X	合流	-	分岐先	維持	合流
-	-	-	-	維持	移動変形

<スケールの指定>:

コピー元からの変形量を指定する。

書式1:

<初期スケール>+<繰り返しスケールの差分量>

書式2:

+<繰り返しスケールの差分量>

書式3:

<初期スケール>

<初期スケール>:

リピートの先頭にだけ適用されるスケールである。

省略された場合は、コピー元の大きさと一致する事になる。

<繰り返しスケールの差分量>:

リピートの先頭のスケールに対して足される、スケールの差分の量である。差分の量が正の値であれば、リピートの回数が増える程、サブ領域は大きくなる。負の値の場合には、逆に小さくなる。

省略された場合には、リピートの先頭と同じ大きさが維持される事となる。



<初期スケール>及び<繰り返しスケールの差分量>の書式:

書式1:

<スケール1>

書式2:

<スケール1>\*<スケール2>

書式3:

<スケール1>\*<スケール2>\*<スケール3>

<スケール<sub>n</sub>>の書式:

書式1:

<スケール整数部><スケール指定子><スケール小数部>

書式2:

<スケール整数部><スケール指定子>

<スケール整数部>:

アラビア数字で指定する。

<スケール小数部>:

アラビア数字で指定する。

<スケール指定子>:

小数点記号の代わりに、<スケール<sub>n</sub>>がX軸、Y軸、Z軸内のどのパラメータを指定するのかを指定する為の符号であり、係数及び軸の対応は、下記の一覧表の指定に従うものとする。

符号	X軸の係数	Y軸の係数	Z軸の係数
A	1	1	1
B	0.5	0.5	0.5
C	0.25	0.25	0.25
D	1	0.25	0.25
E	0.25	1	0.25
F	0.25	0.25	1
G	1	0.5	0.5
H	0.5	1	0.5
J	0.5	0.5	1
K	1	0.75	0.75
L	0.75	1	0.75
M	0.75	0.75	1

符号	X軸の係数	Y軸の係数	Z軸の係数
N	*	0.5	0.5
P	0.5	*	0.5
Q	0.5	0.5	*
R	-1	*	*
S	*	-1	*
T	*	*	-1
U	*	1	1
V	1	*	1
W	1	1	*
X	1	*	*
Y	*	1	*
Z	*	*	1

表の中に「\*」が入っている項目は、その軸には影響を与えない事を意味している。

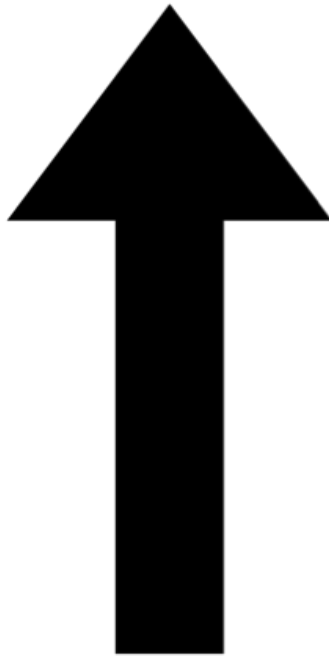
X, Y, ZとはそれぞれX軸、Y軸、Z軸の1軸だけを指定し、その他の軸には影響を与えない。Aの場合は、3軸全てに掛かる係数を指定する事ができる。Vを選んだ場合は、X軸、Z軸に対して掛かる係数を指定する事ができる。

係数が指定されなかった軸は、スケールは1となり変化しない。

<初期スケール>及び<繰り返しスケールの差分量>は、書式2では、2つまで、書式3では3つまで指定する事ができるので、同じ軸に複数の係数が掛ける組み合わせもあり得る。この場合には係数どうし掛け合わせた結果が利用される事になる。

R, S, Tでは、それぞれX軸、Y軸、Z軸の各1軸だけに、負の値を指定する事ができる。

スケールが負の値になった場合は、サブ領域が反転する事になる。これを利用する事で、鏡写しの形状を指定する事ができる。



バリア:

プロパティ名	プロパティの名称	主な情報
B	<バリア情報のプロパティ>	バリア情報

<バリア情報のプロパティ>は、進行を妨げる障害物、壁や柱、ベンチ、テーブル、などの情報を示す為のプロパティである。

バリアという意味では、階段や段差も含まれるが、こちらの情報は、<バリア情報のプロパティ>ではなく、<階数名のプロパティ>の中で取り扱うものとする。

バリアの種類としては、物理的バリア、視覚的なバリア、言語的なものが存在するが、LPSやGMSでは、物理的なバリアの存在を、ベクトル情報として、示す事により、視覚的なバリアを低くする事も狙っている。

また、LPSやGMSでは、対象を英語のデータとして提供する事により、視覚的なバリアと、言語的なバリアを下げる事も狙っている。

物理的なバリアに対するプロパティの使い分け:

<階数名のプロパティ>で直接示することができるもの:

段差、坂道、階段、エスカレーター

<バリア情報のプロパティ>中で指定することができるもの:

車道、ゲート、ドア、踏切、横断歩道、柵、壁、柱、ベンチ、  
このまま進む事が命の危険に繋がる場所、崖、用水路、川、池、  
踏み切りなど、進行の障害になるもの。

#### 障害の属性:

段差、坂道、階段、エスカレーター、横断歩道、車道、踏切、ゲート、ドア、壁、柱、ベンチ、このまま進む事が命の危険に繋がる場所など、進行の障害になるものの有無やその特性を示す為パラメータである。

#### 歩道と路面の特徴:

問題なしから、スリップ、泥濘み、凹凸(段差)、視覚障害者誘導ブロック、格子蓋、溝蓋、マンホール  
車椅子やベビーカーの進行を妨げるもの

#### 物理的な進行の障害物:

柵、壁、柱、ベンチ、車止め、ドア、ゲート

#### このまま進む事が命の危険に繋がる場所:

車道、線路  
踏み切り  
崖、海、川、池

#### 歩道の中で特別なもの:

横断歩道

<バリア情報のプロパティ>の影響範囲:

<バリア情報のプロパティ>と<階数名のプロパティ>は、同じグループの設定であり、互いに、設定を上書きし合う。

<バリア情報のプロパティ>は、後続の<領域のプロパティ>と<サブ領域のプロパティ>に影響を与える。

別の、<バリア情報のプロパティ>または<階数名のプロパティ>が現れるまで、後続の全ての<領域のプロパティ>と<サブ領域>に影響を与える。前方のプロパティには、一切の影響を与えない。

<サブ領域のプロパティ>がある場合は、サブ領域の方向が<バリア情報のプロパティ>の方向を示す向きとなる。

<サブ領域のプロパティ>が無い場合は、<領域のプロパティ>で示した方向となる。

領域の指定が無い場合は、QRコードが設置されている位置と方向がそのまま基準点なる。



書式:

**B:**<障害物の属性><障害物名>.

書式2の障害の追加情報の中身は、障害の属性に合わせて変わるものとする。

<障害物名>:

障害物名は、文字列のパラメータである。このパラメータは、省く事もできる。

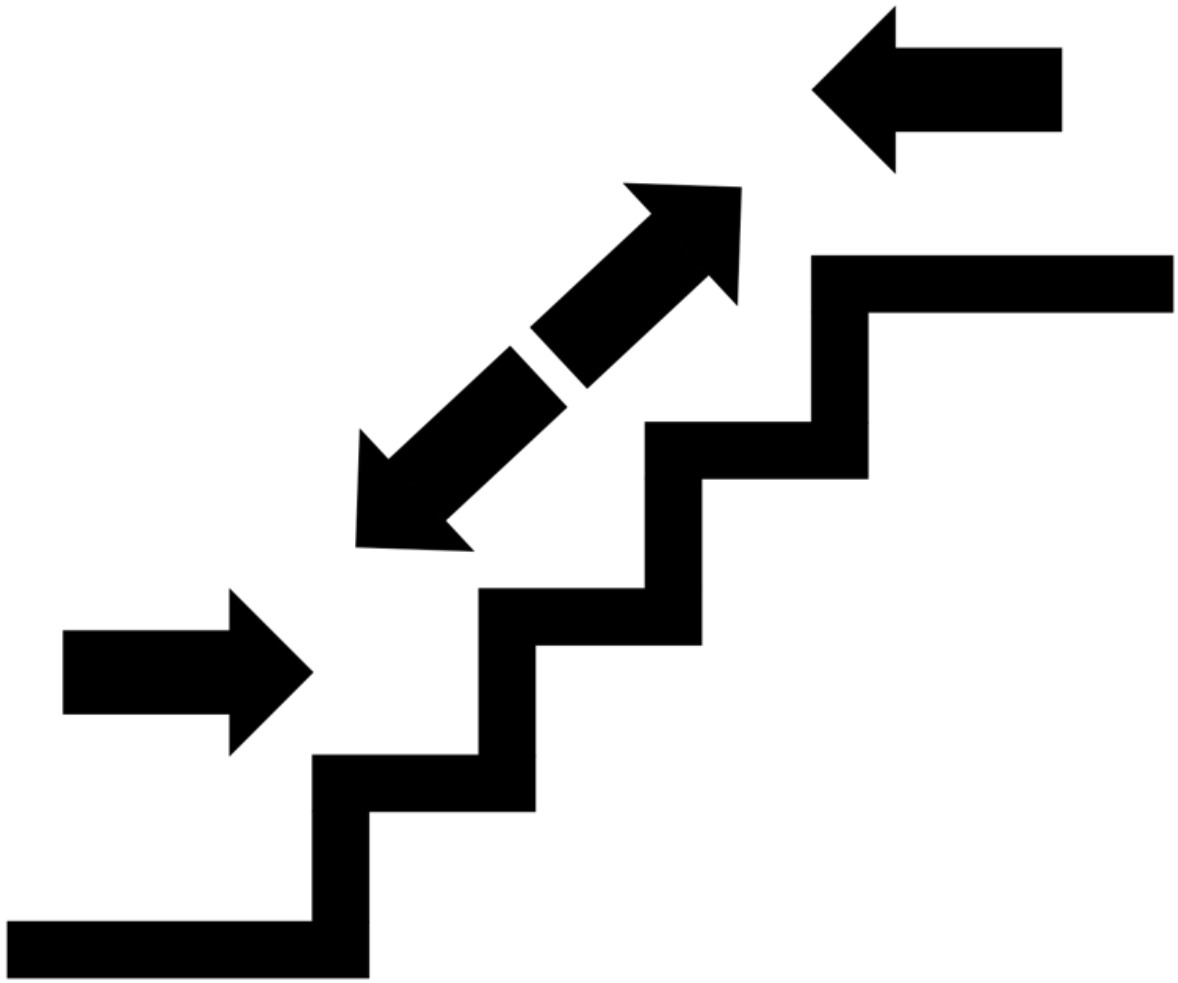
<障害物の属性>:

階数の属性は、<文字列のスタック定数>の一部である<メタ情報タグ>を利用して記述するパラメータである。バリアに関する補足情報やサービス(アクセビリティ等)に関する情報を記載する事を目的としている。このパラメータは、省く事もできる。

バリアフリーの適用状態は、道の両端を繋ぐ橋の完成状態と同じである。橋は、その道を繋ぐ両端まで、橋桁が開通するまで、人は渡り切る事ができない。途中で、橋桁が欠けて向う岸の道までの移動が妨げられているならば、橋はその役割を失ってしまう事になる。

橋と同様に、バリアフリーが求められる人にとっては、目的地迄の経路の全てがバリアフリーとして完成されている事が求められる。

もし、目的地までの経路の一部にでも、バリアが残っているならば、目的地まで、辿り着く事ができなくなる。



階数:

プロパティ名	プロパティの名称	主な情報
F	<階数名のプロパティ>	階数の名称と特徴

多層構造の建物内で、建物方向の位置(階数名)を示す為のプロパティである。

建物の、通常の階数だけでなく、階段、坂道(スロープ)、エスカレーター(動く歩道)、エレベーターの中で利用する場合も考慮した仕様となっている。

また、階の名前を英数字の文字列で示す。

階段や、段差、坂道(スロープ)、エスカレーター、エレベーターを特別階、それ以外を通常階と分類する。また、階段や坂道(スロープ)の途中に、踊り場が設けられている場合があるが、LPS上では、踊り場を、階段や坂道(スロープ)どうしを結ぶ通常階の扱いに分類する事もできる。

アスタリスク(\*)記号とドル(\$)記号は、階数同士の区切りと、階段、坂道(スロープ)、エスカレーター、エレベーターなどの種類、位置などを指定する為に利用する仕様とする。

特に、エスカレーターの外側から、降り口にした場合や、通常階から、階段の下り階に、意図せずに進入した場合は、危険が伴う為、LPSでは、領域のプロパティとサブ領域のプロパティを利用して、階段やエスカレーターの形状をベクター情報として提供できる仕組みとした(この仕組みはGMSのサブセットである)。

後続に、<領域のプロパティ>や<サブ領域のプロパティ>がある場合は、サブ領域の方向が階段や坂道(スロープ)、エスカレーターのなどの方向を示す向きとなる。

<領域のプロパティ>や<サブ領域のプロパティ>が無い場合は、QRコードが設置されている位置と方向がそのまま基準点なる。

<階数名のプロパティ>の影響範囲:

別の、<バリア情報のプロパティ>または<階数名のプロパティ>が現れるまで、後続の全ての<領域のプロパティ>と<サブ領域>に影響を与える。前方のプロパティには、一切の影響を与えない。

階数のプロパティに使用されるエスケープ文字:

LPSとGMS共通のエスケープ文字に加えて、階数のパラメータ内では、下記のエスケープ文字を追加するものとする。

階数のパラメータ内固有のエスケープ文字の記述方法:

「\$」は、「/\$」と記述する。

書式1

**F:**<階の種別><通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

書式2

**F:**<階の種別><通路の属性><通路名>\$<階数の属性1><階数名1>.

書式3

**F:**<階の種別><通路の属性><通路名>.

書式4

**F:**<通路の属性><通路名>.

書式5

**F:.**

書式1では、階数名1に上の階、階数名2に下の階を記入する。

書式4、書式5は、通常階専用の書式である。

上下の階が、通常階として定義している場合にその間に挟まれる階段等の特殊階で書式3を利用する事ができる。

書式2は、現在の階から、別の階に移動する階段等の設備で、移動先の階の情報を通常階として定義する事を省きたい場合に、情報量を節約する為に利用される。

<階の種別>:

階の種別は、地形の特徴を示す階数固有のパラメータである。

詳細については、後述する。

<通路の属性>:

通路の属性は、<文字列のスタック定数>の一部である<メタ情報タグ>を利用して記述するパラメータである。バリアに関する補足情報やサービス(アクセビリティ等)に関する情報を記載する事を目的としている。このパラメータは、省く事もできる。

通路の属性は、階段に、適用した場合には、通路の属性となる。

坂道や、エスカレータ、エレベータの場合にも同様である。

<通路名>:

通路の名は、文字列のパラメータである。このパラメータは、省く事もできる。通路名は、階段に、適用した場合には、階段の名前となる。坂道や、エスカレータ、エレベータの場合にも同様である。

<階数の属性<sub>x</sub>>:

階数の属性は、<文字列のスタック定数>の一部である<メタ情報タグ>を利用して記述するパラメータである。バリアに関する補足情報やサービス(アクセビリティ等)に関する情報を記載する事を目的としている。このパラメータは、省く事もできる。その階数や領域の特性を記述する為のものである。

<階数名<sub>x</sub>>:

階数名は、文字列のパラメータである。

現在の階数名または地名、到達する階数などを指定するために使用される。このパラメータは、省く事もできる。

## 階の種別:

階の種別は、階数固有のパラメータであり、LPSまたは、GMSのデータが保存されたQRコードが示す位置の、階が、通常の階であるのか、それとも、階段や、坂道(スロープ)、段差、エスカレーター、エレベーターであるか、また、坂道(スロープ)、段差が、車椅子で通れるものなのかを示す。階の種別は、ドル(\$)マーク、アスタリスク(\*)マークから始まり、アルファベット1文字で終わるものとする。従って、ドル(\$)マーク、アスタリスク(\*)マークが先頭に無い場合は、通常の階である事が判別することができる。

階数名の種別一覧表 (1/5) : 階段

符号	施設	車椅子への影響	位置
\$A	階段	移動不能	登り口手前の場合
\$B	階段	移動不能	降り口手前の場合
\$C	階段	移動不能	上の階に近い場合
\$D	階段	移動不能	中間の階にいる場合
\$E	階段	移動不能	下の階に近い場合

続く



階数の種別一覧表 (2/5) : 段差

符号	施設	車椅子への影響	位置
\$F	段差	移動不能	段差の側
\$G	段差	自力移動可能	段差の側

続く

階数名の種別一覧表 (3/5) : 坂道(スロープ)

符号	施設	車椅子への影響	位置
\$H	坂道(スロープ)	移動不能	登り口手前の場合
\$Y	坂道(スロープ)	移動不能	降り口手前の場合
\$J	坂道(スロープ)	移動不能	上の階に近い場合
\$K	坂道(スロープ)	移動不能	中間の階にいる場合
\$L	坂道(スロープ)	移動不能	下の階に近い場合
\$M	坂道(スロープ)	介助者が必要	登り口手前の場合
\$N	坂道(スロープ)	介助者が必要	降り口手前の場合
\$Z	坂道(スロープ)	介助者が必要	上の階に近い場合
\$P	坂道(スロープ)	介助者が必要	中間の階にいる場合
\$Q	坂道(スロープ)	介助者が必要	下の階に近い場合
\$R	坂道(スロープ)	自力移動可能	登り口手前の場合
\$S	坂道(スロープ)	自力移動可能	降り口手前の場合
\$T	坂道(スロープ)	自力移動可能	上の階に近い場合
\$U	坂道(スロープ)	自力移動可能	中間の階にいる場合
\$V	坂道(スロープ)	自力移動可能	下の階に近い場合

続く

階数名の種別一覧表(4/5): エスカレーター

符号	施設	車椅子への影響	位置
\$W	エスカレーター 上り方向	移動不能	入口手前の場合
\$X	エスカレーター 上り方向	移動不能	出口手前の場合
*A	エスカレーター 上り方向	移動不能	途中-入口に近い場合
*B	エスカレーター 上り方向	移動不能	途中-中間にいる場合
*C	エスカレーター 上り方向	移動不能	途中-出口に近い場合
*D	エスカレーター 下り方向	移動不能	入口手前の場合
*E	エスカレーター 下り方向	移動不能	出口手前の場合
*F	エスカレーター 下り方向	移動不能	途中-入口に近い場合
*G	エスカレーター 下り方向	移動不能	途中-中間にいる場合
*H	エスカレーター 下り方向	移動不能	途中-出口に近い場合

続く

階数名の種別一覧表 (5/5) : エレベーター

符号	施設	車椅子への影響	位置
*Y	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*J	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*K	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*L	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が決まっていない場合
*M	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*N	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*Z	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*P	エレベーターの扉用印刷物	自力移動可能	通常運行で到達可能な階が決まっていない場合
*Q	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*R	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*S	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*T	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が決まっていない場合
*U	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まってドアが閉まっている場合
*V	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まってドアが開いている場合
*W	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが上昇している場合
*X	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが降下している場合

通常階にいる場合:

階の種別の先頭文字に、ドル(\$)マーク、アスタリスク(\*)マークが入らない場合は、通常階を示す。

通常階は、書式4を利用し、<通路名>に現在の階数名または地名を入れるものとする。階数名は文字列のパラメータである。

通常階は、建物だけでなく、列車、バス、タクシー、航空機なども含まれる。

列車、バス、タクシー、航空機の場合には、サービスの名前や方面などを入れるものとする。

また、階数名の1文字目には、ドル(\$)マーク、アスタリスク(\*)マークを入れてはならない。

書式4

**F:**<通路の属性><通路名>.

書式5

**F:.**

階段の場合:

下記の符号が、階の種別として用いられる。

符号	施設	車椅子への影響	位置
\$A	階段	移動不能	登り口手前の場合
\$B	階段	移動不能	降り口手前の場合
\$C	階段	移動不能	上の階に近い場合
\$D	階段	移動不能	中間の階にいる場合
\$E	階段	移動不能	下の階に近い場合

階段は、車椅子での、移動不能をデフォルトの値としている。

車椅子の移動を可能な設備がある場合や介助者がいれば、車椅子での移動が容易な場合には、<通路の属性>に補足情報を記述することができる。

各階を結ぶ階段の手前(進入口)の場合:

符号	施設	車椅子への影響	位置
\$A	階段	移動不能	登り口手前の場合
\$B	階段	移動不能	降り口手前の場合

通常階から階段を見た場合に使用される事を想定した属性である。

書式2の利用を推奨する。

書式2を利用した場合には、階数名1には、上の階数名または下の階数名が入る。

登り口手前の場合:

**F:\$A**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: 階段の持つ属性

<通路名>: 階段の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

降り口手前の場合:

**F:\$B**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: 階段の持つ属性

<通路名>: 階段の名称

<階数の属性1>: 下の階数の持つ属性

<階数名1>: 下の階数名

各階を結ぶ階段の途中の場合:

符号	施設	車椅子への影響	位置
\$C	階段	移動不能	上の階に近い場合
\$D	階段	移動不能	中間の階にいる場合
\$E	階段	移動不能	下の階に近い場合

階段上で利用する事を想定した属性である。

書式1の利用を推奨する。書式1を使用した場合には、階数名1には、上の階数名、階数名2には、下の階数名が入る。

上の階に近い場合:

**F:\$C**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

中間の階にいる場合:

**F:\$D**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

降り口手前の場合:

**F:\$E**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

<通路の属性>: 階段の持つ属性

<通路名>: 階段の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

<階数の属性2>: 下の階数の持つ属性

<階数名2>: 下の階数名

段差の場合:

下記の符号が、階の種別として用いられる。

符号	施設	車椅子への影響	位置
\$F	段差	移動不能	段差の側
\$G	段差	自力移動可能	段差の側

書式3を利用する。

**F:**<階の種別><通路の属性><通路名>.

車椅子での移動不能の場合:

**F:\$F**<通路の属性><通路名>.

車椅子で自力移動が可能の場合:

**F:\$G**<通路の属性><通路名>.

<通路の属性>:

介助者が居れば、車椅子での移動が可能等の補足情報を入れる  
事ができる。

省略可能なパラメータである。

<通路名>:

地上では、現在地名、建物では、階数名等を入れるものとする。  
省略可能なパラメータである



坂道(スロープ)の場合:

下記の符号が、階の種別として用いられる。

符号	施設	車椅子への影響	位置
\$H	坂道(スロープ)	移動不能	登り口手前の場合
\$Y	坂道(スロープ)	移動不能	降り口手前の場合
\$J	坂道(スロープ)	移動不能	上の階に近い場合
\$K	坂道(スロープ)	移動不能	中間の階にいる場合
\$L	坂道(スロープ)	移動不能	下の階に近い場合
\$M	坂道(スロープ)	介助者が必要	登り口手前の場合
\$N	坂道(スロープ)	介助者が必要	降り口手前の場合
\$Z	坂道(スロープ)	介助者が必要	上の階に近い場合
\$P	坂道(スロープ)	介助者が必要	中間の階にいる場合
\$Q	坂道(スロープ)	介助者が必要	下の階に近い場合
\$R	坂道(スロープ)	自力移動可能	登り口手前の場合
\$S	坂道(スロープ)	自力移動可能	降り口手前の場合
\$T	坂道(スロープ)	自力移動可能	上の階に近い場合
\$U	坂道(スロープ)	自力移動可能	中間の階にいる場合
\$V	坂道(スロープ)	自力移動可能	下の階に近い場合

\* 介助者が居れば、車椅子での移動が可能な場合などの条件を指定する場合は、書式1~4を利用し、<通路の属性>の項目の中に記載するものとする。

詳細については、以降のページに記載する。

坂道(スロープ)に対する種別では、位置と、車椅子利用者に与える影響で分類している。

施設に対する位置は、登り口手前場合、降り口手前の場合、上の階に近い場合、上の階に近い場合、中間の階にいる場合にいる場合に分類している。

車椅子利用者に与える影響として、移動不能、介助者が必要、自力移動可能に分類している。

車椅子で移動不能:

介助者がいても、車椅子を利用して安全に移動することができない事を示す。

車椅子で介助者が必要:

車椅子利用者が自力で移動する事は困難であるが、介助者がいれば移動可能な事を示す。

車椅子で自力移動可能:

車椅子を利用者が自力で安全に移動できる事を示す。

車椅子で自力移動可能な条件は、その地域の特性及び、車椅子の改良と共に時代によっても変換して行くものと考えられる為、本規格の中では基準を設けないが、その施設の特徴と地域の基準に合わせて、LPS及びGMSのデータを作成する者が、適切に選択する事が求められる。また、選択に当たっては、実際に、車椅子の利用者が立ち会いを行う事が求められる。

坂道(スロープ)の手前(進入口)の場合:

符号	施設	車椅子への影響	位置
\$H	坂道(スロープ)	移動不能	登り口手前の場合
\$Y	坂道(スロープ)	移動不能	降り口手前の場合
\$M	坂道(スロープ)	介助者が必要	登り口手前の場合
\$N	坂道(スロープ)	介助者が必要	降り口手前の場合
\$R	坂道(スロープ)	自力移動可能	登り口手前の場合
\$S	坂道(スロープ)	自力移動可能	降り口手前の場合

坂道(スロープ)の手前(進入口)の場合には、書式2の利用を推奨する。階数名1には、上の階数名または、下の階数名が入る。

坂道(スロープ)の外側から登り口手前の場合:

符号	施設	車椅子への影響	位置
\$H	坂道(スロープ)	移動不能	登り口手前の場合
\$M	坂道(スロープ)	介助者が必要	登り口手前の場合
\$R	坂道(スロープ)	自力移動可能	登り口手前の場合

通常階から坂道(スロープ)を見た場合に使用される事を想定した属性である。

書式2の利用を推奨する。

**F:**<階の種別><通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で移動不能:

**F:\$H**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で介助者が必要:

**F:\$M**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で自力移動可能:

**F:\$R**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: 坂道(スロープ)の持つ属性

<通路名>: 坂道(スロープ)の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

坂道(スロープ)の外側から降り口手前の場合:

符号	施設	車椅子への影響	位置
<b>\$Y</b>	坂道(スロープ)	移動不能	降り口手前の場合
<b>\$N</b>	坂道(スロープ)	介助者が必要	降り口手前の場合
<b>\$S</b>	坂道(スロープ)	自力移動可能	降り口手前の場合

書式2の利用を推奨する。

**F:**<階の種別><通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で移動不能:

**F:\$Y**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で介助者が必要:

**F:\$N**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

車椅子で自力移動可能:

**F:\$S**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: 坂道(スロープ)の持つ属性

<通路名>: 坂道(スロープ)の名称

<階数の属性1>: 下の階数の持つ属性

<階数名1>: 下の階数名

坂道(スロープ)の途中の場合:

符号	施設	車椅子への影響	位置
\$J	坂道(スロープ)	移動不能	上の階に近い場合
\$K	坂道(スロープ)	移動不能	中間の階にいる場合
\$L	坂道(スロープ)	移動不能	下の階に近い場合
\$Z	坂道(スロープ)	介助者が必要	上の階に近い場合
\$P	坂道(スロープ)	介助者が必要	中間の階にいる場合
\$Q	坂道(スロープ)	介助者が必要	下の階に近い場合
\$T	坂道(スロープ)	自力移動可能	上の階に近い場合
\$U	坂道(スロープ)	自力移動可能	中間の階にいる場合
\$V	坂道(スロープ)	自力移動可能	下の階に近い場合

書式1の利用を推奨する。

詳細については、以降のページに記載する。

上の階に近い場合:

符号	施設	車椅子への影響	位置
\$J	坂道(スロープ)	移動不能	上の階に近い場合
\$Z	坂道(スロープ)	介助者が必要	上の階に近い場合
\$T	坂道(スロープ)	自力移動可能	上の階に近い場合

書式1の利用を推奨する。

車椅子で移動不能:

**F:\$J**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で介助者が必要:

**F:\$Z**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で自力移動可能:

**F:\$T**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

<通路の属性>: 坂道(スロープ)の持つ属性

<通路名>: 坂道(スロープ)の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

<階数の属性2>: 下の階数の持つ属性

<階数名2>: 下の階数名

中間の階の場合:

符号	施設	車椅子への影響	位置
\$K	坂道(スロープ)	移動不能	中間の階にいる場合
\$P	坂道(スロープ)	介助者が必要	中間の階にいる場合
\$U	坂道(スロープ)	自力移動可能	中間の階にいる場合

書式1の利用を推奨する。

車椅子で移動不能:

**F:\$K**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で介助者が必要:

**F:\$P**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で自力移動可能:

**F:\$U**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

<通路の属性>: 坂道(スロープ)の持つ属性

<通路名>: 坂道(スロープ)の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

<階数の属性2>: 下の階数の持つ属性

<階数名2>: 下の階数名



下の階に近い場合:

符号	施設	車椅子への影響	位置
\$L	坂道(スロープ)	移動不能	下の階に近い場合
\$Q	坂道(スロープ)	介助者が必要	下の階に近い場合
\$V	坂道(スロープ)	自力移動可能	下の階に近い場合

書式1の利用を推奨する。

車椅子で移動不能:

**F:\$L**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で介助者が必要:

**F:\$Q**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

車椅子で自力移動可能:

**F:\$V**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

<通路の属性>: 坂道(スロープ)の持つ属性

<通路名>: 坂道(スロープ)の名称

<階数の属性1>: 上の階数の持つ属性

<階数名1>: 上の階数名

<階数の属性2>: 下の階数の持つ属性

<階数名2>: 下の階数名

エスカレーター(動く歩道)の場合:

下記の符号が、階の種別として用いられる。

符号	施設	車椅子への影響	位置
\$W	エスカレーター 上り方向	移動不能	入口手前の場合
\$X	エスカレーター 上り方向	移動不能	出口手前の場合
*A	エスカレーター 上り方向	移動不能	途中-入口に近い場合
*B	エスカレーター 上り方向	移動不能	途中-中間にいる場合
*C	エスカレーター 上り方向	移動不能	途中-出口に近い場合
*D	エスカレーター 下り方向	移動不能	入口手前の場合
*E	エスカレーター 下り方向	移動不能	出口手前の場合
*F	エスカレーター 下り方向	移動不能	途中-入口に近い場合
*G	エスカレーター 下り方向	移動不能	途中-中間にいる場合
*H	エスカレーター 下り方向	移動不能	途中-出口に近い場合

エスカレーターは、入口から入って、出口へ抜けて行く一方通行の設備である。入口は、可動部の最初のステップが現れる一番手前の位置(入口手前)を示し、出口は、可動部のステップが隠れる一番終端の部分(出口手前)を示す。

出口は、エスカレーターの進行方向を0度として扱うものとする。

また、出口の反対側の向きは、一方通行の為、進入禁止として扱われるものとする。

従って、LPSの情報を記録した、QRコードが、要所に、配置され、全てのQRコードを読み取る事ができた場合、次の位置を示す符号が順番に、読み取れる事となる。

(1)通常階

(2)入口手前、(3)途中-入口に近い、(3)途中-中間、(4)途中-出口に近い

(5)出口手前

(6)通常階

注意:

QRコードの読み取りが可能かどうかは、様々な条件によって変わってくるため、あくまでLPSやGMSは、他の手段を補助する手段の一つである。このLPSやGMSに安全を担保させてはならない。仮に先の例で、(1)～(6)に対応するQRコードが全て用意されていても、条件が悪ければ全て読み取れ無い可能性がある。

エスカレーターは、車椅子での移動不能として扱われるが、介助者の協力のもと、複数のステップを結合して、車椅子を運ぶ機能を持った設備の場合には、階数名に、その情報を記載する事ができるものとする。

入り口手前、降り口手前の場合は書式1で到達先の階数を階数名1に指定する。途中の場合は、書式2を利用し、階数名1には、到着先の階数名、階数2には、出発地点の階数名を記載する。

動く歩道では、建物の出口から遠ざかる方向を上り、出口に近く方向を下り方面と読み替えるものとする。

エスカレーターの外側から上り方向を見た場合:

符号	施設	車椅子への影響	位置
<b>\$W</b>	エスカレーター 上り方向	移動不能	入口手前の場合
<b>\$X</b>	エスカレーター 上り方向	移動不能	出口手前の場合

書式1の利用を推奨する。

エスカレーターの外側から上り方向の入口手前を見た場合:

**F:\$W**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

エスカレーターの外側から上り方向の出口手前を見た場合:

**F:\$X**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: エスカレーターの持つ属性

<通路名>: エスカレーターの名称

<階数の属性1>: 到着先の階数名の持つ属性

<階数名1>: 到着先の階数名

エスカレーターの上り方向の場合:

符号	施設	車椅子への影響	位置
*A	エスカレーター 上り方向	移動不能	途中-入口に近い場合
*B	エスカレーター 上り方向	移動不能	途中-中間にいる場合
*C	エスカレーター 上り方向	移動不能	途中-出口に近い場合

書式1の利用を推奨する。

上り方向の途中-入口近い場合:

**F:\*A**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

上り方向の途中-中間にいる場合:

**F:\*B**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

上り方向の途中-出口近い場合:

**F:\*C**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

<通路の属性>: エスカレーターの持つ属性

<通路名>: エスカレーターの名称

<階数の属性1>: 到着先の階数の持つ属性

<階数名1>: 到着先の階数名

<階数の属性2>: 出発地点の階数の持つ属性

<階数名2>: 出発地点の階数名

エスカレーターの外側から下り方向を見た場合:

符号	施設	車椅子への影響	位置
*D	エスカレーター 下り方向	移動不能	入口手前の場合
*E	エスカレーター 下り方向	移動不能	出口手前の場合

書式2の利用を推奨する。

下り方向の入口手前の場合:

**F:\*D**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

下り方向の出口手前の場合:

**F:\*E**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: エスカレーターの持つ属性

<通路名>: エスカレーターの名称

<階数の属性1>: 到着先の階数名の持つ属性

<階数名1>: 到着先の階数名

エスカレーターの下り方向の場合:

符号	施設	車椅子への影響	位置
*F	エスカレーター 下り方向	移動不能	途中-入口に近い場合
*G	エスカレーター 下り方向	移動不能	途中-中間にいる場合
*H	エスカレーター 下り方向	移動不能	途中-出口に近い場合

書式1の利用を推奨する。

下り方向の途中-入口近い場合:

**F:\*F**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

下り方向の途中-中間にいる場合:

**F:\*G**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

下り方向の途中-出口近い場合:

**F:\*H**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

<通路の属性>: エスカレーターの持つ属性

<通路名>: エスカレーターの名称

<階数の属性1>: 到着先の階数の持つ属性

<階数名1>: 到着先の階数名

<階数の属性2>: 出発地点の階数の持つ属性

<階数名2>: 出発地点の階数名

エレベーター(鉄道/バス/タクシー)の場合:

下記の符号が、階の種別として用いられる。

符号	施設	車椅子への影響	位置
*Y	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*J	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*K	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*L	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が決まっていない場合
*M	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*N	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*Z	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*P	エレベーターの扉用印刷物	自力移動可能	通常運行で到達可能な階が決まっていない場合
*Q	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*R	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最上階だけ決まっている場合
*S	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*T	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が決まっていない場合
*U	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まっているドアが閉まっている場合
*V	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まっているドアが開いている場合
*W	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが上昇している場合
*X	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが降下している場合



エレベーター用の階数の種別は、エレベーターの外側用と、内側用に大きく分けられる。

内側、外側は、QRコードが、エレベーターの外側に貼られているものなのか、内側に貼られているかを基本の向きを示すものである。

領域の指定や、サブ領域の指定により、向きの解釈を変更した場合には、内側と外側が入れ変える事もできるが、外側用を用意する事で、この指定を省略して、データの量を減す事を可能にしている。

また、エレベーターは、車椅子による自力移動が可能としているが、もし、車椅子での自力移動が不可能な場合には、階数名1にその補足情報を記載するものとする。

エレベーター(鉄道/バス/タクシー)の扉用印刷物:

符号	施設	車椅子への影響	位置
*Y	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*J	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な最上階だけ決まっている場合
*K	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*L	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が決まっていない場合
*M	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*N	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な最上階だけ決まっている場合
*Z	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*P	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が決まっていない場合

エレベーターの扉に印刷された、QRコードは、扉が開く事で、イメージセンサーから、消える事になる。従って、QRコードを検出できている間は、扉が閉まっている事を示す。

ただし、QRコードは、イメージセンサーとの間に人が入っただけで、検出できなくなるため、QRコードが検出できないことが必ずしも、扉が開いている事を示しているわけではない。

また、エレベーターの扉に印刷された、QRコードは、扉の開閉に合わせて、(一般的なものは、横方に、一部のものは横方向に)移動するため、位置の精度は、低いものとして扱う、また、エレベーター扉に印刷された案内板に掲載されるものについては、下記のように示す事とする。

エレベーターが通常運行で到達可能な階が決まっている場合:

書式1の利用を推奨する。

符号	施設	車椅子への影響	位置
*Y	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*M	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合

扉の外側用:

**F:\*Y**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

扉の内側用:

**F:\*M**<通路の属性><通路名>\$<階数の属性1><階数名1>\$<階数の属性2><階数名2>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最上階の持つ属性

<階数名1>: エレベーターが到達可能な最上階の名前

<階数の属性2>: エレベーターが到達可能な最低階の持つ属性

<階数名2>: エレベーターが到達可能な最低階の名前

エレベーターが通常運行で到達可能な階が上だけ決まっている場合:

書式2の利用を推奨する。

符号	施設	車椅子への影響	位置
*J	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な最上階だけ決まっている場合
*N	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な最上階だけ決まっている場合

扉の外側用:

**F:\*J**<通路の属性><通路名>\$<階数の属性1><階数名1>.

扉の内側用:

**F:\*N**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最上階の持つ属性

<階数名1>: エレベーターが到達可能な最上階の名前

エレベーターが通常運行で到達可能な階が下だけ決まっている場合:

書式2を利用を推奨する。

符号	書式	施設	車椅子への影響	位置
*K	1	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*Z	1	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合

扉の外側用:

**F:\*K**<通路の属性><通路名>\$<階数の属性1><階数名1>.

扉の内側用:

**F:\*Z**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最低階の持つ属性

<階数名1>: エレベーターが到達可能な最低階の名前

エレベーターが通常運行で到達可能な階が決まっていない場合:

書式3を利用する。

符号	施設	車椅子への影響	位置
*L	エレベーターの扉用印刷物 外側用	自力移動可能	通常運行で到達可能な階が決まっていない場合
*P	エレベーターの扉用印刷物 内側用	自力移動可能	通常運行で到達可能な階が決まっていない場合

扉の外側用:

**F:\*L<通路の属性><通路名>.**

**F:\*L.**

扉の内側用:

**F:\*P<通路の属性><通路名>.**

**F:\*P.**

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

エレベーターの固定印刷物:

符号	施設	車椅子への影響	位置
*Q	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最上階と最低階が決まっている場合
*R	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な最上階だけ決まっている場合
*S	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が最低階だけ決まっている場合
*T	エレベーターの固定印刷物	自力移動可能	通常運行で到達可能な階が決まっていない場合

エレベーターが通常運行で到達可能な階が決まっている場合:

書式1を利用する。

**F:\*Q**<通路の属性><通路名>**\$**<階数の属性1><階数名1>**\$**<階数の属性2><階数名2>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最上階の持つ属性

<階数名1>: エレベーターが到達可能な最上階の名前

<階数の属性2>: エレベーターが到達可能な最低階の持つ属性

<階数名2>: エレベーターが到達可能な最低階の名前

エレベーターが通常運行で到達可能な階が上だけ決まっている場合:  
書式2を利用する。

**F:\*R**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最上階の持つ属性

<階数名1>: エレベーターが到達可能な最上階の名前

エレベーターが通常運行で到達可能な階が下だけ決まっている場合:  
書式2を利用する。

**F:\*S**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが到達可能な最低階の持つ属性

<階数名1>: エレベーターが到達可能な最低階の名前

エレベーターが通常運行で到達可能な階が決まっていない場合:  
書式3を利用する。

**F:\*T**<通路の属性><通路名>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称



エレベーターの表示切り替えが可能な場合:

符号	書式	施設	車椅子への影響	位置
*U	1	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まってドアが閉まっている場合
*V	1	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーター止まってドアが開いている場合
*W	1	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが上昇している場合
*X	1	エレベーターで表示を切り替えが可能な場合	自力移動可能	エレベーターが降下している場合

エレベーター内の液晶ディスプレイなどの、動的に表示を切り替えが可能なものに掲載されるものについては、下記の様に示す事する。

エレベーターが止まってドアが閉まっている場合:

書式2を利用する。

**F:\*U<通路の属性><通路名>\$<階数の属性1><階数名1>.**

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターの現在の階の持つ属性

<階数名1>: エレベーターの現在の階の名前

エレベーターが止まってドアが開いている場合:

書式1を利用する。

**F:\*V**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターの現在の階の持つ属性

<階数名1>: エレベーターの現在の階の名前

エレベーターが上昇している場合:

書式2を利用する。

**F:\*W**<通路の属性><通路名>**\$**<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが次に止まる階の持つ属性

<階数名1>: エレベーターが次に止まる階の名前

エレベーターが下降している場合:

書式2を利用する。

**F:\*X**<通路の属性><通路名>\$<階数の属性1><階数名1>.

<通路の属性>: エレベーターの持つ属性

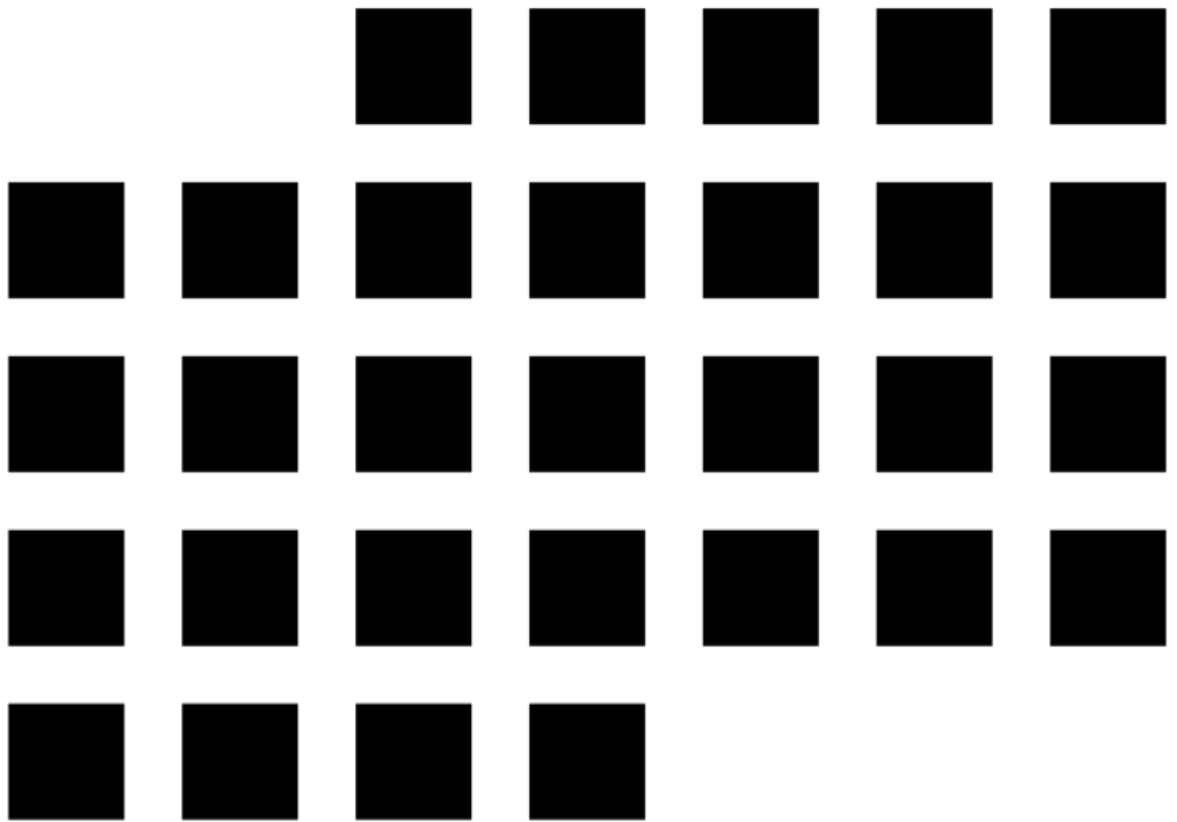
<通路名>: エレベーターの名称

<階数の属性1>: エレベーターが次に止まる階の持つ属性

<階数名1>: エレベーターが次に止まる階の名前

※地上でLPSを利用する場合にはこの項目を省略する事もできる。

※GPSを利用する事で、高さを求める事も可能になるが、高さとの階数の関係については、別途、建物毎にデータベースが必要となる。日本に於いては、地上面を1Fとするが、英国に於いては、Grand floor または、0で表す。また途中階(例:M2)を設ける場合や、宗教的な理由で縁起の悪い数値を省く場合もあり、建物内で、5A,5B,5Cなどに分かれる場合があるため、数値で管理できない場合が想定される。従って、階数は英数字の文字列として扱うものとする。



作成年月日時分秒:

プロパティ名	プロパティの名称	主な情報
Y	<マークの作成年月日のプロパティ>	作成年月および時分秒

年の単位は、2000年からの経過年数とする。

月と時は下記の変換表に従ってアルファベットとに置き換える。

日と分と秒はアラビア数値で指定する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

<マークの作成年月日のプロパティ>の書式:

**Y:**<年月日時分秒のパラメータ>

<年月日時分秒のパラメータ>の書式:

書式1

<年><月><日><時><分><秒>

書式2

<年><月><日><時><分>

書式3

<年><月><日><時>

書式4

<年><月><日>

書式5

<年><月>

書式6

<年>

<年>の書式:

西暦2000年からの差分をアラビア数字で表記する。

<月>の書式:

月の値は、下記のアルファベットで符号化する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

<日>の書式:

月の値は、アラビア数字1桁または、2桁で表記する。

<時>の書式:

時間の値は、下記のアルファベットで符号化する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

<分>の書式:

分の値は、アラビア数字2桁で表記する。2桁にならない場合には、十桁目を0埋めする。

<秒>の書式:

分の値は、アラビア数字2桁で表記する。2桁にならない場合には、十桁目を0埋めする。

例:

2021年5月2日の場合: 21F2

2099年12月31日の場合:99M31

2099年12月31日23時の場合:99M31X

2099年12月31日23時45分の場合:99M31X45

2099年12月31日23時0分12秒の場合:99M31X0012

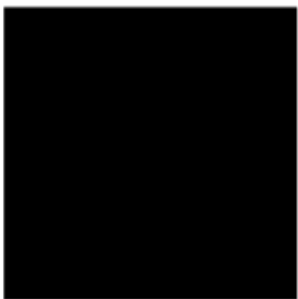
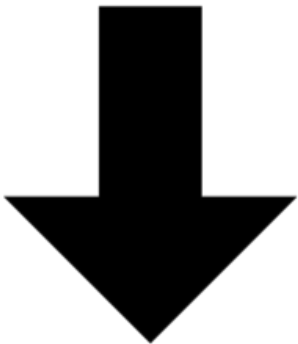
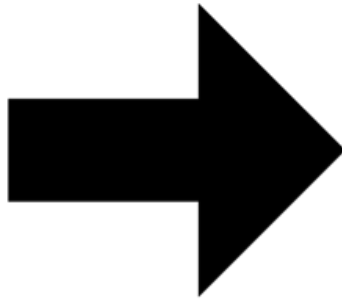
2099年12月31日23時59分59秒の場合:99M31X5959

2100年1月1日の場合: 100B01

※アルファベットの大文字のIとOはアラビア数字の1と0との判読が困難となるため、それぞれ大文字のYとZに置き換える。

※アラビア数字ではなく、アルファベットでコード化する理由は、より少ない文字数で多くの情報を入れる事を可能にする為である。





連結先のQRコードのある方向と距離:

プロパティ名	プロパティの名称	主な情報
L	<外部マークの情報のプロパティ>	連絡先のQRコードがある方向と距離

案内板上で複数のQRコードを利用して情報を提供する場合に、連結先のQRコードシンボルの方向を示す為の情報を入れる事ができる。

この項目は、連結先のQRコードが存在しない場合は省略できる。

連結先のQRコードは、同じ平面上に、同じ向きで並んでいるものとする。

書式1から、書式5まで必要に応じて選べるものとする。

書式1:

**L:**<QRコードのある方向 1><QRコードまでの距離 1>+<QRコードのバージョン 1><QRコードの横幅 1>\$<QRコードのある方向 2><QRコードまでの距離 2>+<QRコードのバージョン 2><QRコードの横幅 2>\$...<QRコードのある方向 n><QRコードまでの距離 n>+<QRコードのバージョン n><QRコードの横幅 n>.

書式2:

**L:**<QRコードのある方向 1><QRコードまでの距離 1>+<QRコードのバージョン 1>\$<QRコードのある方向 2><QRコードまでの距離 2>+<QRコードのバージョン 2>\$...<QRコードのある方向 n><QRコードまでの距離 n>+<QRコードのバージョン n>.

書式3:

**L:**<QRコードのある方向 1><QRコードまでの距離 1>+<QRコードの横幅 1>\$<QRコードのある方向 2><QRコードまでの距離 2>+<QRコードの横幅 2>\$...<QRコードのある方向 n><QRコードまでの距離 n>+<QRコードの横幅 n>.

書式4:

**L:**<QRコードのある方向 1>+<QRコードのバージョン 1>\$<QRコードのある方向 2>+<QRコードのバージョン 2>\$...<QRコードのある方向 n>+<QRコードのバージョン n>.

書式4:

**L:**<QRコードのある方向 1><QRコードまでの距離 1>**\$**<QRコードのある方向 2><QRコードまでの距離 2>**\$**...<QRコードのある方向 n><QRコードまでの距離 n>.

書式5:

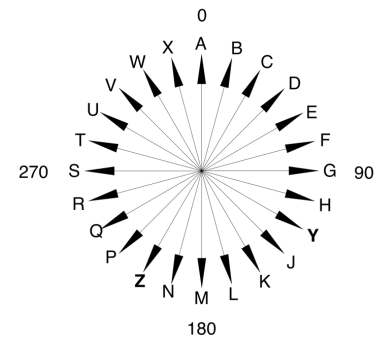
**L:**<QRコードのある方向 1>**\$**<QRコードのある方向 2>**\$**...<QRコードのある方向 n>.

<QRコードのある方向>:

角度のパラメータである。

下記の置換表と計算式によりアルファベットでコード化した  
もので指定する。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X



アルファベット1文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

例: A=0度、B=15度、D=45度、E=60度、G=90度、Y=120度、M=180度

アルファベット2文字で表記する場合:

角度=360/24\*<1文字目の対応表で示される数値>

+360/24/24\*<2文字目の対応表で示される数値>

例: AB=0.625度、GQ=97.5+2.5=100度

1文字のAに対し2文字のAAはより高い有効精度の場合に使用される。

アルファベットn文字で表記する場合:

$$y = \sum_{k=1}^n \frac{x_k}{24^k}$$

y = 角度

$x_k$  = アルファベットで表記された対応する数値

k = アルファベットの左から数えた文字数

$n =$  アルファベットの全体の文字数

角度とは $2\pi$ を分割する値に対する比率であり、度数法では360等分した際の割合で示すものである。LPSでは360等分ではなく、 $24^n$ 等分するという考えに基づいている。その為、 $n = 1$ の場合には24等分と精度が低くなるが、 $n = 2$ の場合には576等分となり、度数方で整数の範囲で表記する場合よりも精度が良くなる。

従って実用上、アルファベット1文字から2文字で十分な精度が得られる事が期待できるが、必要に応じて3文字以上で、表記しても良い。LPSを利用する携帯情報端末が、アルファベット3文字以上で書かれたデータを受け取った際の処理として、3桁以降を値として細かく計算するか、切り捨て処理を行うかは実装の目的に合わせて判断するものとする。

※符号化する際にアルファベットの大文字のIとOはアラビア数字の1と0との判読が困難となるため、それぞれ大文字のYとZに置き換える。

※アルファベットから数値に復号する際は、アルファベットの大文字のIは数値8、アルファベットの大文字のOは、数値14として扱うものとする。

※アラビア数字ではなく、アルファベットでコード化する理由は、より少ない文字数で多くの情報を入れる事を可能にし、精度と情報量のトレードオフを可能にする為である。

<QRコードまでの距離>:

必要に応じて、角度の情報に加えてQRコード間の距離も指定できる。  
長さのパラメータである。

単位はメートルを基本とし、先頭の文字はアラビア数字から始まるものとする。

小数点も利用できる。小数点を示す記号にはピリオドの代わりに、単位の1文字を使用する。

メートル単位の場合:

Mをピリオドの代わりに使用できる。デフォルトの単位とする  
為、少数点が必要な場合はmを省く事ができる。

センチメートル単位の場合:

Cをピリオドの代わりに使用できる。

キロメートル単位の場合:

Kをピリオドの代わりに使用できる。

例:

12.34mの場合: 12M34

12mの場合: 12

12cmの場合: 12C

0.34mの場合: 0M34

0.3mの場合: 0M3

連結対象が複数ある場合は、ドルマーク「\$」で区切る事により、複数のQRコードをリストとして登録する事ができる。

先頭に近いもの程、重要な情報となる。

リストに、A0がある場合は、自身に、他のQRコードに含まれていない重要な情報を含んでいる事を示す。

QRコードは大きな型番(バージョン)を利用する事で、小さな型番(バージョン)の内容を内包した上で、より沢山の情報を持たせる事もできる為、連結先のQRコードに自身の持つ情報が内包されている場合は、このQRコードの内容の全てを解析する必要がなくなる。このような場合には、「連結先のQRコードのある方向と距離」のパラメータは、データ構造の先頭の方に配置する事を推奨する。

- ※QRコード間の距離とは、マークの中心から中心までの距離を示す。
- ※ピリオドは、QRコードに登録されたデータを文字列として人が、目視で確認する作業が発生した場合に、見落としを減らす為に、代わりにアルファベット1文字を入れる事とする。
- ※QRコードの仕様自体で最大16個まで連結する機能を持っているが、LPS、GMSは、1つのQRコードだけが認識できる場合であっても単独で役立つため、QRコードの仕様とは別の連結機能を定義する。
- ※このプロパティは、主にLPSのデータが記録されているQRコードからGMSのデータが記録されているQRコードを参照する事を想定したものとなっている。GMSはデータ量が多い為、遠くからの認識が困難になるが、この仕組みを利用するとLPSをGMS用の特大のマーカとして利用できるようになる。



<QRコードのバージョン>:

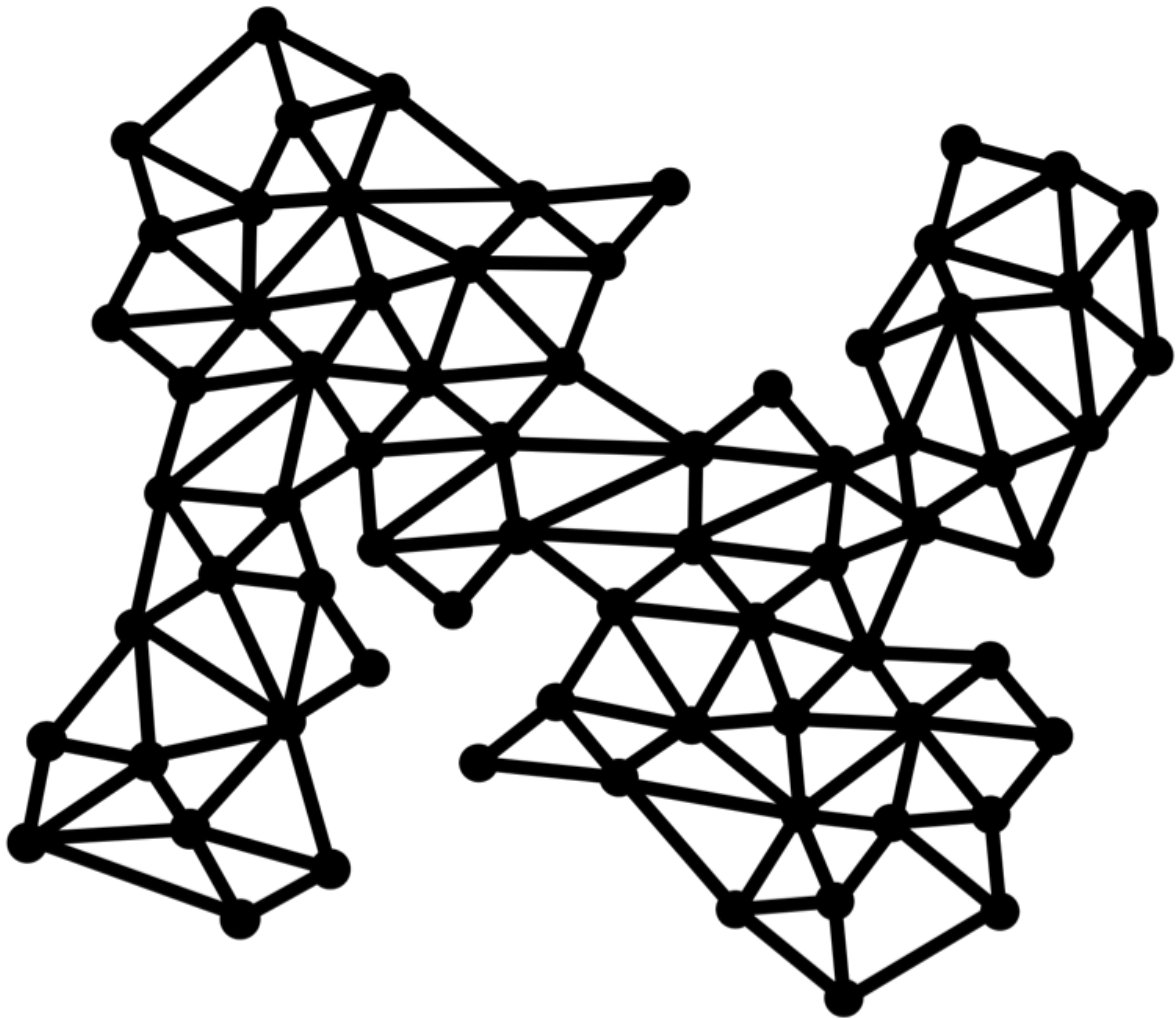
必要に応じて、QRコードのバージョンを指定できる。

バージョンはアルファベットを利用し下記の表に従い、24進数で符号化されるものとする。24進数で求められた値が $x$ とした場合は、バージョンは、 $x + 1$ とする。

数値	0	1	2	3	4	5	6	7	8	9	10	11
文字	A	B	C	D	E	F	G	H	Y	J	K	L
数値	12	13	14	15	16	17	18	19	20	21	22	23
文字	M	N	Z	P	Q	R	S	T	U	V	W	X

2021年7月現在の時点でQRコードのバージョンは、1から40まで定義されている。従って0を指定する事が無いため、オフセットとして1を加えている。

従ってアルファベット1文字で1から24までのバージョンを、2文字で、25から40までのバージョンを示すものとする。



プロパティ名	プロパティの名称	主な情報
N	<固有識別番号のプロパティ>	固有の識別番号

URLや、ucodeの番号を保存するためのプロパティである。

LPSおよびGMSでは、データの格納媒体として、QRコードの利用を前提としているが、QRコードに、入れる事ができるデータの量は限られている。これに対し、WifiやBluetooth、携帯電話の電波を利用すれば、QRコード単体では提供する事が難しい量のデータであっても、簡単に、提供することが可能となる。しかしながら、双方向性のあるネットワークにアクセスする事は、常に、ネットワークに接続された時点で、その携帯情報端末は位置情報が特定される事にも繋がる。また、電波による通信は、その通信インフラの状況に依存し、必ず接続できる保証は無い。また、ネットワークに接続する事自体がセキュリティに対し潜在的な脆弱性をもたらす事にも繋がる。

URLを直接埋め込む仕様にする事は、URLに悪意をWebサイトへのアドレスを埋め込む事も可能となる為、潜在的な脆弱性を持つ事に繋がる為、仕様からの排除を考慮したが、もし規格から排除した場合には、より大きな脆弱性を持つ無理な拡張を盛り込んだ、本規格の亜種が登場し、制御不能な自体になる恐れがあるため、予め仕様に盛り込む事で、リスクを下げる事を目的としてこのプロパティを定義した。

このプロパティは、オプションであり、使用する事を推奨しないが、利便性の為に、URLをLPS及びGMSに埋め込む必要が生じた場合には、このプロパティを利用して埋め込むものとする。

URLでホスト名をFQDNを利用して指定する場合:

書式 1

**N:A**<FQDN及びurl-path>.

FQDN及びurl-pathのURLで指定する。

書式 2

**N:B**<www.から始まるFQDN及びurl-path>.

www.から始まるFQDN及びurl-pathのURLで指定する。

書式 3

**N:C**<FQDN及びurl-path>.

FQDN及びurl-pathのURLで指定する。

プロパティMとプロパティD、プロパティPの情報を、<query-string>として送信する。

書式 4

**N:D**<www.から始まるFQDN及びurl-path>.

www.から始まるFQDN及びurl-pathのURLで指定する。

プロパティ名Mとプロパティ名D、プロパティ名Pの情報を、<query-string>として送信する。

パラメータ:

<FQDN及びurl-path>:

<文字列のパラメータ>であるが、下記に示す特別な規則が適用される。

<文字列のスタック>に保存されない。

<文字列のスタック定数>は利用できない。

その代わりとして、パーセントマーク(%)は、<文字列のパラメータ>としては他のアラビア数字の文字列と同じ扱いとなる。

漢字モードやバイナリモードを利用しUTF-8で記述された、URLでは直接利用する事ができない文字については、自動的にパーセントエンコーディングされる。

<www.から始まるFQDN及びurl-path> :

<FQDN及びurl-path>と基本的には同じであるが、ホスト名の一番先頭に、“www.”が追加される。このホスト名は、最も多様される為、4文字分データを削減できる様にパラメータを分けている。

URLでホスト名をIPアドレスで指定する場合:

#### 書式 1

**U:P**<IPアドレス>.

ホスト名としてIPアドレスで指定する。

#### 書式 2

**U:Q**<url-path>**\$**<IPアドレス>.

ホスト名としてIPアドレスで指定する。

#### 書式 3

**U:R**<IPアドレス>.

ホスト名として、IPアドレスで指定し、プロパティ名Mとプロパティ名D、プロパティ名Pの情報を、<query-string>として送信する。

#### 書式 4

**U:S**<url-path>**\$**<IPアドレス>.

ホスト名として、IPアドレスで指定し、プロパティMとプロパティD、プロパティPの情報を、<query-string>として送信する。

パラメータ:

<IPアドレス>:

<オクテットコード>のパラメータである。

LPSおよびGMSでは、<オクテットコード>のパラメータを扱う為のエンコードを専用に定めている。

<url-path>:

<文字列のパラメータ>であるが、下記に示す特別な規則が適用される。

下記のエスケープ文字が利用される。

表記したい文字	英語名	実際の記述
\$	Period	/\$

<文字列のスタック>に保存されない。

<文字列のスタック定数>は利用できない。

その代わりとして、パーセントマーク(%)は、<文字列のパラメータ>としては他のアラビア数字の文字列と同じ扱いとなる。

漢字モードやバイナリモードを利用しUTF-8で記述された、URLでは直接利用する事ができない文字については、自動的にパーセントエンコーディングされる。

ucodeを利用する場合:

#### 書式 1

**N:U**<ucode>.

ucodeを指定する。

#### 書式 2

**N:V**<ucode>.

ucodeを指定し、プロパティ名Mとプロパティ名D、プロパティ名Pの情報を、<query-string>として送信する。

パラメータ:

<ucode>:

<オクテットコード>のパラメータである。

LPSおよびGMSでは、<オクテットコード>のパラメータを扱う為のエンコードを専用に定めている。

ucodeは、トロントフォーラム傘下のユピキタスセンターが提供するあらゆるものに、固有の番号を割り振るサービスを提供するシステムである。

ucodeの詳細については、トロントフォーラムの資料を参照するものとする。



<固有の識別番号のプロパティ>内の指定と携帯情報端末の設定の関係:

<固有の識別番号のプロパティ>内で如何なる指定が行われた場合であっても、常に、携帯情報端末の設定するプライバシーの設定を厳守する事が要求される。

LPSおよびGMSに対応したアプリケーションは、LPSおよびGMSに保存されているデータを取り扱う一番初めに、プライバシーに関する設定を行う事ができるものとし、デフォルトでは、LPSおよびGMSの<固有の識別番号のプロパティ>を無視するものとする。

プライバシーに関する設定は、ユーザーに対する十分な説明と同意のも  
といつでも、ユーザーの意識で自由に切り替える事ができるものとする。  
プライバシーを保護する方向への切り替えは、簡単で良いが、プライバシーとなる位置情報が外部に送信される方向に対しては、ユーザーに詳しい説明を行う必要がある。

下記の2つの項目について、ユーザーにアクセスの許可を求め、  
許可が無い場合には、アクセスを許してはならない。

LPSおよびGMSに関する外部アクセスの許可

プロパティMとプロパティD、プロパティPの情報を送信する事の許可

双方向通信の技術を利用し、携帯情報端末が、外部ネットワークに対し、何らかのアクセスを行う接続が行われた場合には、潜在的に常に外

部のネットワークから、その携帯情報端末の位置情報が、識別される可能性がある事を意味する。

従って、まず初めに、LPSおよびGMSで、<固有の識別番号のプロパティ>により、外部へのアクセスを許可するかどうかを選べる必要がある。

<固有の識別番号のプロパティ>の中では、IPアドレス、FQDN、ucodeでドメインを指定できる仕様となっているので、ドメインの指定方法に合わせて、外部のアクセスの許可をオプションで更に細かく一部だけ、許可できる様にしても良い。また、一部だけ許可できる場合でも、一括で、禁止にできるようにする事が必要である。

アクセスが禁止されている場合には、プロパティMとプロパティD、プロパティPの情報を、<query-string>として送信する指定が、選択されている場合であっても、上記ののプロパティの情報は送信されないものとする。

携帯情報端末のユーザーは、地点と半径を決めて、複数の個所でプライバシーのポリシーを指定する事もできる。

また、ポリシーの中には、<領域のプロパティ>のパラメータ<領域の属性>に、従ってポリシーの切り替えを自動的に切り替えるという選択肢を設けても良いが、その場合であっても、手動で設定したアクセスの禁止の方が必ず優先しなくてはならないものとする。

ネットワークに対するアクセスが許可されている場合であっても、LPSおよびGMS以外の情報を提供された情報を、Webブラウザなどのアプリケーションと関係を許すかどうかは別にアクセスの許可が必要なものとする。

スキーム名:

https

プロトコル名:

HTTPS

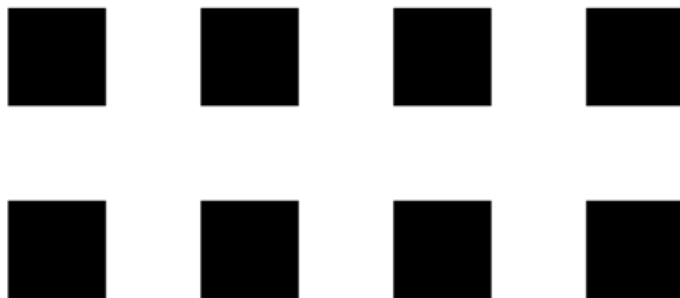
ファイル名:

“GUIDE.GMS”

全て大文字とする。

ファイル形式:

バイナリとする。



<オクテットコード>のパラメータ:

IPアドレスやunicodeなどのオクテットコードを表現したい場合にデータを記録できる様に<オクテットコード>のパラメータを定義する。

IPアドレスは2021年現在、IPv4では、32ビット長、4オクテッド長のアドレス。IPv6では、128ビット長、16オクテット長のものが利用されているが、<オクテットコード>のパラメータは、任意のオクテット長のデータを指定できる為、どちらの形式にも共通で利用できるものとなっている。

<オクテットコード>のパラメータの構造:

<オクテットコード>のエンコードを行う為のテーブルとして289個の項目で構成される、<High-density-table>と<Low-density-table>の2つのテーブルを定義している。

<High-density-table>は、前後のオクテットの値の差が、-17~0~+18の範囲に入った場合に、最も少ないデータで、オクテットの値を表現する事が可能なテーブルとなっている。つまり、オクテットに値が特定の値の範囲に集中している場合に、最もデータの量が少なくなる様に設計されたテーブルである。

<Low-density-table>は、<High-density-table>では効率的に表現できない、前後のオクテットの値の差が、分散して規則性が無いデータのパターンに合わせたテーブルとなっている。

どちらのテーブルも初期設定の状態では、前のオクテットの値を基準点の値として、この値に対する差分のから、次のオクテットの値を指定している。

### <オクテットエンコード>のテーブルの構成

Group	文字数	項目数	Index 一番小さな番号	Index 一番大きな番号	主な役割
A	1	36	0	35	基準点からのオフセット値による1オクテットの値の指定
B	2	246	36	255	基準点からのオフセット値による1オクテットの値の指定&テーブルの切り替え
C	2	32	256	287	基準点の制御、テーブルの制御、繰り返しパターンの指定
E	1	1	288	288	終了

この2つのテーブルは、共通の構造を持っており、オクテットの値に、よって相互に切り替わる。

この2つのテーブルは、0～288のインデックスで構成されているが、先頭の0～34までの35個の項目のインデックスは1文字(5.5ビット)で表現できる。この範囲のインデックスをグループAとして分類している。

1文字で、表現できる範囲から漏れた、残りは2文字(11ビット)で表現される。256-35項目で、残りの35～255までのインデックスの範囲をグループB、256～287をグループCと分類する。また、これとは別に、プロパティの終了を示すピリオド1文字をグループEに分類している。2つのテーブルとも、グループCとグループEの部分は共通のものとなっている。

グループAは、前の基準点からのオフセットの値で、次のオクテットの値を指定する事を想定したものとなっている。

グループBの項目にも、グループAと同様に、前の基準点からのオフセットの値で、次のオクテットの値を指定したものとなっているが、グループAとは、大

大きく異なる事がある。それは、グループAの場合の場合の倍のサイズが必要となる事である。つまり、グループAではなく、グループBに入った場合は、そのテーブルの想定する戦略が上手く機能しなかった可能性が高かった事を意味するの為、初期設定の戦略では、この後に、<Low-density-table>、<High-density-table>の2つのテーブル間での切り替えを行う事とする。

グループCには、テーブルを切り替える戦略を切り替える際の規則の変更、基準点の指定方法の変更、繰り返しパターンの指定、<オクテットコード>の終了を指定するための指示が入っている。このグループCの部分だけを切り出したテーブルを<Control-table>と呼ぶ。

グループEは、プロパティの終了を意味するピリオドが割り当てられている。このピリオドは、<オクテットコード>の一部であると同時に、プロパティの終了を示す為の文字の一部でもある。このグループE部分だけを切り出したテーブルを<End-table>と呼ぶ。

各テーブルの表記について:

特に、<Low-density-table>、<High-density-table>のテーブルサイズは、289個の項目から、構成されており、テーブルへの対応をQRコード上で指定する英数字と記号の文字で指定した場合には、仕様上の形状に近い文字を読み間違えが有る恐れがあるため、QRコードの英数字モード上の文字番号で、テーブル上の文字を指定する。<オクテットコード>では、QRコードの英数字モードをそのまま利用する。QRコードの英数字モードでは、<QRコード英数字コード表>に従い、アラビア数字のゼロ「0」からコロン「:」までの文字に、0から44までの番号割り振られている。ここではこの番号を<#C>としてテーブルのインデックスに使用する。但し、コロン「:」は利用しない。

まず、各テーブルで利用されているQRコードの英数字モードの文字コードのテーブルを下記に示す。

文字	<#C>	備考
0	0	アラビア数字
1	1	アラビア数字
2	2	アラビア数字
3	3	アラビア数字
4	4	アラビア数字
5	5	アラビア数字
6	6	アラビア数字
7	7	アラビア数字
8	8	アラビア数字
9	9	アラビア数字
A	10	アルファベット
B	11	アルファベット
C	12	アルファベット
D	13	アルファベット
E	14	アルファベット
F	15	アルファベット
G	16	アルファベット
H	17	アルファベット
I	18	アルファベット
J	19	アルファベット
K	20	アルファベット
L	21	アルファベット
M	22	アルファベット

文字	<#C>	備考
N	23	アルファベット
O	24	アルファベット
P	25	アルファベット
Q	26	アルファベット
R	27	アルファベット
S	28	アルファベット
T	29	アルファベット
U	30	アルファベット
V	31	アルファベット
W	32	アルファベット
X	33	アルファベット
Y	34	アルファベット
Z	35	アルファベット
Space	36	空白
\$	37	ドル
%	38	パーセント
*	39	アスタリスク
+	40	プラス
-	41	マイナス
.	42	ピリオド
/	43	スラッシュ
:	44	コロン

以降に、各テーブルの中身を示す。



<High-density-table>

<High-density-table> (1/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
0	0		A	-17	239	239
1	1		A	-16	240	240
2	2		A	-15	241	241
3	3		A	-14	242	242
4	4		A	-13	243	243
5	5		A	-12	244	244
6	6		A	-11	245	245
7	7		A	-10	246	246
8	8		A	-9	247	247
9	9		A	-8	248	248
10	10		A	-7	249	249
11	11		A	-6	250	250
12	12		A	-5	251	251
13	13		A	-4	252	252
14	14		A	-3	253	253
15	15		A	-2	254	254
16	16		A	-1	255	255
17	17		A	0	0	0
18	18		A	1	1	1
19	19		A	2	2	2
20	20		A	3	3	3
21	21		A	4	4	4
22	22		A	5	5	5
23	23		A	6	6	6
24	24		A	7	7	7
25	25		A	8	8	8
26	26		A	9	9	9
27	27		A	10	10	10
28	28		A	11	11	11
29	29		A	12	12	12

<High-density-table> (2/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
30	30		A	13	13	13
31	31		A	14	14	14
32	32		A	15	15	15
33	33		A	16	16	16
34	34		A	17	17	17
35	35		A	18	18	18
36	36	0	B	19	19	19
37	36	1	B	20	20	20
38	36	2	B	21	21	21
39	36	3	B	22	22	22
40	36	4	B	23	23	23
41	36	5	B	24	24	24
42	36	6	B	25	25	25
43	36	7	B	26	26	26
44	36	8	B	27	27	27
45	36	9	B	28	28	28
46	36	10	B	29	29	29
47	36	11	B	30	30	30
48	36	12	B	31	31	31
49	36	13	B	32	32	32
50	36	14	B	33	33	33
51	36	15	B	34	34	34
52	36	16	B	35	35	35
53	36	17	B	36	36	36
54	36	18	B	37	37	37
55	36	19	B	38	38	38
56	36	20	B	39	39	39
57	36	21	B	40	40	40
58	36	22	B	41	41	41
59	36	23	B	42	42	42

<High-density-table> (3/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
60	36	24	B	43	43	43
61	36	25	B	44	44	44
62	36	26	B	45	45	45
63	36	27	B	46	46	46
64	36	28	B	47	47	47
65	36	29	B	48	48	48
66	36	30	B	49	49	49
67	36	31	B	50	50	50
68	36	32	B	51	51	51
69	36	33	B	52	52	52
70	36	34	B	53	53	53
71	36	35	B	54	54	54
72	36	36	B	55	55	55
73	36	37	B	56	56	56
74	36	38	B	57	57	57
75	36	39	B	58	58	58
76	36	40	B	59	59	59
77	36	41	B	60	60	60
78	37	0	B	61	61	61
79	37	1	B	62	62	62
80	37	2	B	63	63	63
81	37	3	B	64	64	64
82	37	4	B	65	65	65
83	37	5	B	66	66	66
84	37	6	B	67	67	67
85	37	7	B	68	68	68
86	37	8	B	69	69	69
87	37	9	B	70	70	70
88	37	10	B	71	71	71
89	37	11	B	72	72	72

<High-density-table> (4/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
90	37	12	B	73	73	73
91	37	13	B	74	74	74
92	37	14	B	75	75	75
93	37	15	B	76	76	76
94	37	16	B	77	77	77
95	37	17	B	78	78	78
96	37	18	B	79	79	79
97	37	19	B	80	80	80
98	37	20	B	81	81	81
99	37	21	B	82	82	82
100	37	22	B	83	83	83
101	37	23	B	84	84	84
102	37	24	B	85	85	85
103	37	25	B	86	86	86
104	37	26	B	87	87	87
105	37	27	B	88	88	88
106	37	28	B	89	89	89
107	37	29	B	90	90	90
108	37	30	B	91	91	91
109	37	31	B	92	92	92
110	37	32	B	93	93	93
111	37	33	B	94	94	94
112	37	34	B	95	95	95
113	37	35	B	96	96	96
114	37	36	B	97	97	97
115	37	37	B	98	98	98
116	37	38	B	99	99	99
117	37	39	B	100	100	100
118	37	40	B	101	101	101
119	37	41	B	102	102	102

<High-density-table> (5/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
120	38	0	B	103	103	103
121	38	1	B	104	104	104
122	38	2	B	105	105	105
123	38	3	B	106	106	106
124	38	4	B	107	107	107
125	38	5	B	108	108	108
126	38	6	B	109	109	109
127	38	7	B	110	110	110
128	38	8	B	111	111	111
129	38	9	B	112	112	112
130	38	10	B	113	113	113
131	38	11	B	114	114	114
132	38	12	B	115	115	115
133	38	13	B	116	116	116
134	38	14	B	117	117	117
135	38	15	B	118	118	118
136	38	16	B	119	119	119
137	38	17	B	120	120	120
138	38	18	B	121	121	121
139	38	19	B	122	122	122
140	38	20	B	123	123	123
141	38	21	B	124	124	124
142	38	22	B	125	125	125
143	38	23	B	126	126	126
144	38	24	B	127	127	127
145	38	25	B	128	-128	128
146	38	26	B	129	-127	129
147	38	27	B	130	-126	130
148	38	28	B	131	-125	131
149	38	29	B	132	-124	132

<High-density-table> (6/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
150	38	30	B	133	-123	133
151	38	31	B	134	-122	134
152	38	32	B	135	-121	135
153	38	33	B	136	-120	136
154	38	34	B	137	-119	137
155	38	35	B	138	-118	138
156	38	36	B	139	-117	139
157	38	37	B	140	-116	140
158	38	38	B	141	-115	141
159	38	39	B	142	-114	142
160	38	40	B	143	-113	143
161	38	41	B	144	-112	144
162	39	0	B	145	-111	145
163	39	1	B	146	-110	146
164	39	2	B	147	-109	147
165	39	3	B	148	-108	148
166	39	4	B	149	-107	149
167	39	5	B	150	-106	150
168	39	6	B	151	-105	151
169	39	7	B	152	-104	152
170	39	8	B	153	-103	153
171	39	9	B	154	-102	154
172	39	10	B	155	-101	155
173	39	11	B	156	-100	156
174	39	12	B	157	-99	157
175	39	13	B	158	-98	158
176	39	14	B	159	-97	159
177	39	15	B	160	-96	160
178	39	16	B	161	-95	161
179	39	17	B	162	-94	162

<High-density-table> (7/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
180	39	18	B	163	-93	163
181	39	19	B	164	-92	164
182	39	20	B	165	-91	165
183	39	21	B	166	-90	166
184	39	22	B	167	-89	167
185	39	23	B	168	-88	168
186	39	24	B	169	-87	169
187	39	25	B	170	-86	170
188	39	26	B	171	-85	171
189	39	27	B	172	-84	172
190	39	28	B	173	-83	173
191	39	29	B	174	-82	174
192	39	30	B	175	-81	175
193	39	31	B	176	-80	176
194	39	32	B	177	-79	177
195	39	33	B	178	-78	178
196	39	34	B	179	-77	179
197	39	35	B	180	-76	180
198	39	36	B	181	-75	181
199	39	37	B	182	-74	182
200	39	38	B	183	-73	183
201	39	39	B	184	-72	184
202	39	40	B	185	-71	185
203	39	41	B	186	-70	186
204	40	0	B	187	-69	187
205	40	1	B	188	-68	188
206	40	2	B	189	-67	189
207	40	3	B	190	-66	190
208	40	4	B	191	-65	191
209	40	5	B	192	-64	192



<High-density-table> (8/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
210	40	6	B	193	-63	193
211	40	7	B	194	-62	194
212	40	8	B	195	-61	195
213	40	9	B	196	-60	196
214	40	10	B	197	-59	197
215	40	11	B	198	-58	198
216	40	12	B	199	-57	199
217	40	13	B	200	-56	200
218	40	14	B	201	-55	201
219	40	15	B	202	-54	202
220	40	16	B	203	-53	203
221	40	17	B	204	-52	204
222	40	18	B	205	-51	205
223	40	19	B	206	-50	206
224	40	20	B	207	-49	207
225	40	21	B	208	-48	208
226	40	22	B	209	-47	209
227	40	23	B	210	-46	210
228	40	24	B	211	-45	211
229	40	25	B	212	-44	212
230	40	26	B	213	-43	213
231	40	27	B	214	-42	214
232	40	28	B	215	-41	215
233	40	29	B	216	-40	216
234	40	30	B	217	-39	217
235	40	31	B	218	-38	218
236	40	32	B	219	-37	219
237	40	33	B	220	-36	220
238	40	34	B	221	-35	221
239	40	35	B	222	-34	222

<High-density-table> (9/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
240	40	36	B	223	-33	223
241	40	37	B	224	-32	224
242	40	38	B	225	-31	225
243	40	39	B	226	-30	226
244	40	40	B	227	-29	227
245	40	41	B	228	-28	228
246	41	0	B	229	-27	229
247	41	1	B	230	-26	230
248	41	2	B	231	-25	231
249	41	3	B	232	-24	232
250	41	4	B	233	-23	233
251	41	5	B	234	-22	234
252	41	6	B	235	-21	235
253	41	7	B	236	-20	236
254	41	8	B	237	-19	237
255	41	9	B	238	-18	238
256	41	10	C			
257	41	11	C			
258	41	12	C			
259	41	13	C			
260	41	14	C			
261	41	15	C			
262	41	16	C			
263	41	17	C			
264	41	18	C			
265	41	19	C			
266	41	20	C			
267	41	21	C			
268	41	22	C			
269	41	23	C			

<High-density-table> (10/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
270	41	24	C			
271	41	25	C			
272	41	26	C			
273	41	27	C			
274	41	28	C			
275	41	29	C			
276	41	30	C			
277	41	31	C			
278	41	32	C			
279	41	33	C			
280	41	34	C			
281	41	35	C			
282	41	36	C			
283	41	37	C			
284	41	38	C			
285	41	39	C			
286	41	40	C			
287	41	41	C			
288	42		E			

<Low-density-table>

<Low-density-table> (1/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
0	0		A	-128	128	128
1	1		A	-119	137	137
2	2		A	-113	143	143
3	3		A	-101	155	155
4	4		A	-97	159	159
5	5		A	-89	167	167
6	6		A	-73	183	183
7	7		A	-61	195	195
8	8		A	-53	203	203
9	9		A	-43	213	213
10	10		A	-31	225	225
11	11		A	-23	233	233
12	12		A	-13	243	243
13	13		A	-7	249	249
14	14		A	-3	253	253
15	15		A	-2	254	254
16	16		A	-1	255	255
17	17		A	0	0	0
18	18		A	1	1	1
19	19		A	2	2	2
20	20		A	3	3	3
21	21		A	5	5	5
22	22		A	7	7	7
23	23		A	11	11	11
24	24		A	13	13	13
25	25		A	23	23	23
26	26		A	31	31	31
27	27		A	43	43	43
28	28		A	53	53	53
29	29		A	61	61	61

<Low-density-table> (2/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
30	30		A	73	73	73
31	31		A	89	89	89
32	32		A	97	97	97
33	33		A	101	101	101
34	34		A	113	113	113
35	35		A	127	127	127
36	36	0	B	4	4	4
37	36	1	B	6	6	6
38	36	2	B	8	8	8
39	36	3	B	9	9	9
40	36	4	B	10	10	10
41	36	5	B	12	12	12
42	36	6	B	14	14	14
43	36	7	B	15	15	15
44	36	8	B	16	16	16
45	36	9	B	17	17	17
46	36	10	B	18	18	18
47	36	11	B	19	19	19
48	36	12	B	20	20	20
49	36	13	B	21	21	21
50	36	14	B	22	22	22
51	36	15	B	24	24	24
52	36	16	B	25	25	25
53	36	17	B	26	26	26
54	36	18	B	27	27	27
55	36	19	B	28	28	28
56	36	20	B	29	29	29
57	36	21	B	30	30	30
58	36	22	B	32	32	32
59	36	23	B	33	33	33

<Low-density-table> (3/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
60	36	24	B	34	34	34
61	36	25	B	35	35	35
62	36	26	B	36	36	36
63	36	27	B	37	37	37
64	36	28	B	38	38	38
65	36	29	B	39	39	39
66	36	30	B	40	40	40
67	36	31	B	41	41	41
68	36	32	B	42	42	42
69	36	33	B	44	44	44
70	36	34	B	45	45	45
71	36	35	B	46	46	46
72	36	36	B	47	47	47
73	36	37	B	48	48	48
74	36	38	B	49	49	49
75	36	39	B	50	50	50
76	36	40	B	51	51	51
77	36	41	B	52	52	52
78	37	0	B	54	54	54
79	37	1	B	55	55	55
80	37	2	B	56	56	56
81	37	3	B	57	57	57
82	37	4	B	58	58	58
83	37	5	B	59	59	59
84	37	6	B	60	60	60
85	37	7	B	62	62	62
86	37	8	B	63	63	63
87	37	9	B	64	64	64
88	37	10	B	65	65	65
89	37	11	B	66	66	66

<Low-density-table> (4/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
90	37	12	B	67	67	67
91	37	13	B	68	68	68
92	37	14	B	69	69	69
93	37	15	B	70	70	70
94	37	16	B	71	71	71
95	37	17	B	72	72	72
96	37	18	B	74	74	74
97	37	19	B	75	75	75
98	37	20	B	76	76	76
99	37	21	B	77	77	77
100	37	22	B	78	78	78
101	37	23	B	79	79	79
102	37	24	B	80	80	80
103	37	25	B	81	81	81
104	37	26	B	82	82	82
105	37	27	B	83	83	83
106	37	28	B	84	84	84
107	37	29	B	85	85	85
108	37	30	B	86	86	86
109	37	31	B	87	87	87
110	37	32	B	88	88	88
111	37	33	B	90	90	90
112	37	34	B	91	91	91
113	37	35	B	92	92	92
114	37	36	B	93	93	93
115	37	37	B	94	94	94
116	37	38	B	95	95	95
117	37	39	B	96	96	96
118	37	40	B	98	98	98
119	37	41	B	99	99	99



<Low-density-table> (5/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
120	38	0	B	100	100	100
121	38	1	B	102	102	102
122	38	2	B	103	103	103
123	38	3	B	104	104	104
124	38	4	B	105	105	105
125	38	5	B	106	106	106
126	38	6	B	107	107	107
127	38	7	B	108	108	108
128	38	8	B	109	109	109
129	38	9	B	110	110	110
130	38	10	B	111	111	111
131	38	11	B	112	112	112
132	38	12	B	114	114	114
133	38	13	B	115	115	115
134	38	14	B	116	116	116
135	38	15	B	117	117	117
136	38	16	B	118	118	118
137	38	17	B	119	119	119
138	38	18	B	120	120	120
139	38	19	B	121	121	121
140	38	20	B	122	122	122
141	38	21	B	123	123	123
142	38	22	B	124	124	124
143	38	23	B	125	125	125
144	38	24	B	126	126	126
145	38	25	B	129	-127	129
146	38	26	B	130	-126	130
147	38	27	B	131	-125	131
148	38	28	B	132	-124	132
149	38	29	B	133	-123	133

<Low-density-table> (6/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
150	38	30	B	134	-122	134
151	38	31	B	135	-121	135
152	38	32	B	136	-120	136
153	38	33	B	138	-118	138
154	38	34	B	139	-117	139
155	38	35	B	140	-116	140
156	38	36	B	141	-115	141
157	38	37	B	142	-114	142
158	38	38	B	144	-112	144
159	38	39	B	145	-111	145
160	38	40	B	146	-110	146
161	38	41	B	147	-109	147
162	39	0	B	148	-108	148
163	39	1	B	149	-107	149
164	39	2	B	150	-106	150
165	39	3	B	151	-105	151
166	39	4	B	152	-104	152
167	39	5	B	153	-103	153
168	39	6	B	154	-102	154
169	39	7	B	156	-100	156
170	39	8	B	157	-99	157
171	39	9	B	158	-98	158
172	39	10	B	160	-96	160
173	39	11	B	161	-95	161
174	39	12	B	162	-94	162
175	39	13	B	163	-93	163
176	39	14	B	164	-92	164
177	39	15	B	165	-91	165
178	39	16	B	166	-90	166
179	39	17	B	168	-88	168

<Low-density-table> (7/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
180	39	18	B	169	-87	169
181	39	19	B	170	-86	170
182	39	20	B	171	-85	171
183	39	21	B	172	-84	172
184	39	22	B	173	-83	173
185	39	23	B	174	-82	174
186	39	24	B	175	-81	175
187	39	25	B	176	-80	176
188	39	26	B	177	-79	177
189	39	27	B	178	-78	178
190	39	28	B	179	-77	179
191	39	29	B	180	-76	180
192	39	30	B	181	-75	181
193	39	31	B	182	-74	182
194	39	32	B	184	-72	184
195	39	33	B	185	-71	185
196	39	34	B	186	-70	186
197	39	35	B	187	-69	187
198	39	36	B	188	-68	188
199	39	37	B	189	-67	189
200	39	38	B	190	-66	190
201	39	39	B	191	-65	191
202	39	40	B	192	-64	192
203	39	41	B	193	-63	193
204	40	0	B	194	-62	194
205	40	1	B	196	-60	196
206	40	2	B	197	-59	197
207	40	3	B	198	-58	198
208	40	4	B	199	-57	199
209	40	5	B	200	-56	200

<Low-density-table> (8/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
210	40	6	B	201	-55	201
211	40	7	B	202	-54	202
212	40	8	B	204	-52	204
213	40	9	B	205	-51	205
214	40	10	B	206	-50	206
215	40	11	B	207	-49	207
216	40	12	B	208	-48	208
217	40	13	B	209	-47	209
218	40	14	B	210	-46	210
219	40	15	B	211	-45	211
220	40	16	B	212	-44	212
221	40	17	B	214	-42	214
222	40	18	B	215	-41	215
223	40	19	B	216	-40	216
224	40	20	B	217	-39	217
225	40	21	B	218	-38	218
226	40	22	B	219	-37	219
227	40	23	B	220	-36	220
228	40	24	B	221	-35	221
229	40	25	B	222	-34	222
230	40	26	B	223	-33	223
231	40	27	B	224	-32	224
232	40	28	B	226	-30	226
233	40	29	B	227	-29	227
234	40	30	B	228	-28	228
235	40	31	B	229	-27	229
236	40	32	B	230	-26	230
237	40	33	B	231	-25	231
238	40	34	B	232	-24	232
239	40	35	B	234	-22	234

<Low-density-table> (9/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
240	40	36	B	235	-21	235
241	40	37	B	236	-20	236
242	40	38	B	237	-19	237
243	40	39	B	238	-18	238
244	40	40	B	239	-17	239
245	40	41	B	240	-16	240
246	41	0	B	241	-15	241
247	41	1	B	242	-14	242
248	41	2	B	244	-12	244
249	41	3	B	245	-11	245
250	41	4	B	246	-10	246
251	41	5	B	247	-9	247
252	41	6	B	248	-8	248
253	41	7	B	250	-6	250
254	41	8	B	251	-5	251
255	41	9	B	252	-4	252
256	41	10	C			
257	41	11	C			
258	41	12	C			
259	41	13	C			
260	41	14	C			
261	41	15	C			
262	41	16	C			
263	41	17	C			
264	41	18	C			
265	41	19	C			
266	41	20	C			
267	41	21	C			
268	41	22	C			
269	41	23	C			

<Low-density-table> (10/10)

Index	<#C>[0]	<#C>[1]	Group	オフセット値A	オフセット値B	符号無し
270	41	24	C			
271	41	25	C			
272	41	26	C			
273	41	27	C			
274	41	28	C			
275	41	29	C			
276	41	30	C			
277	41	31	C			
278	41	32	C			
279	41	33	C			
280	41	34	C			
281	41	35	C			
282	41	36	C			
283	41	37	C			
284	41	38	C			
285	41	39	C			
286	41	40	C			
287	41	41	C			
288	42		E			

<Control-table>

<Control-table> (1/2)

Index	<#C> [0]	<#C> [1]	Group	#	指示内容	
256	41	10	C	0	基準点を直前のオクテットの値に更新	初期 設定
257	41	11	C	1	基準点をGroup AとBのテーブルを参照する度に現在の値から+36する	
258	41	12	C	2	基準点をGroup AとBのテーブルを参照する度に現在の値から+72する	
259	41	13	C	3	基準点をGroup AとBのテーブルを参照する度に現在の値から+108する	
260	41	14	C	4	基準点をGroup AとBのテーブルを参照する度に現在の値から+144する	
261	41	15	C	5	基準点をGroup AとBのテーブルを参照する度に現在の値から+180する	
262	41	16	C	6	基準点をGroup AとBのテーブルを参照する度に現在の値から+216する	
263	41	17	C	7	基準点を直前のオクテットの値に固定	
264	41	18	C	8	基準点を64に固定	
265	41	19	C	9	基準点を128に固定	
266	41	20	C	10	基準点を172に固定	
267	41	21	C	11	基準点を238に固定	
268	41	22	C	12	Group Bのテーブルの参照する度にテーブルを切り替え	初期 設定
269	41	23	C	13	Group Bのテーブルを2度連続して参照した後にテーブルを切り替え	
270	41	24	C	14	Group AとBのテーブルを参照する度にテーブルを切り替え	
271	41	25	C	15	現在のテーブルに固定	
272	41	26	C	16	基準点の更新規則と、テーブルの切替規則を初期設定に戻す	



<Control-table> (2/2)

Index	<#C> [0]	<#C> [1]	Group	#	指示内容
273	41	27	C	17	直前の2オクテット分のパターンをもう一度繰り返す
274	41	28	C	18	直前の3オクテット分のパターンをもう一度繰り返す
275	41	29	C	19	直前の4オクテット分のパターンをもう一度繰り返す
276	41	30	C	20	直前の5オクテット分のパターンをもう一度繰り返す
277	41	31	C	21	直前の6オクテット分のパターンをもう一度繰り返す
278	41	32	C	22	直前の7オクテット分のパターンをもう一度繰り返す
279	41	33	C	23	直前の8オクテット分のパターンをもう一度繰り返す
280	41	34	C	24	直前の1オクテットの値を2回繰り返す
281	41	35	C	25	直前の1オクテットの値を3回繰り返す
282	41	36	C	26	直前の1オクテットの値を4回繰り返す
283	41	37	C	27	直前の1オクテットの値を5回繰り返す
284	41	38	C	28	直前の1オクテットの値を6回繰り返す
285	41	39	C	29	直前の1オクテットの値を7回繰り返す
286	41	40	C	30	2オクテット前の1オクテットの値を2回繰り返す
287	41	41	C	31	<オクテットコード>の終了

<End-table>

<End-table>

Index	<#C> [0]	<#C> [1]	Group	#	指示内容	
288	42		E	0	<オクテットコード>と<プロパティ>の終了	

<オクテットコード>のデコードは次の規則に従うものとする。

<オクテットコード>上のデータの順番:

常にMSB側が先頭側、LSB側が後ろ側となる。

<オクテットコード>の終了:

<オクテットコード>は、グループCの#31、または、グループE#0で終了する。

グループCの#31は、<オクテットコード>のパラメータが、プロパティの最後に無い場合と、<オクテットコード>が示すデータが2の累乗の値ではない値を示す。

グループCの#31は、将来的に規格を拡張した場合に備えた予約であり、2021年時点での規格では実際には利用することは無い。従って、現時点ではグループE#0だけが利用される。

グループE#0、ピリオド(.)は、<オクテットコード>のパラメータが、プロパティの最後にある場合にだけ利用される。

この場合には、<オクテットコード>が示すデータは、常に2の累乗の分のオクテット数のデータである事を示す。

もし、今までにデコードされたオクテット数が2の累乗に達していない場合、MSB側が2の累乗の個数のデータに達するまでゼロで埋められる。

まだ1オクテットのデータもデコードする前に、ピリオド(.)に到達した場合には、1オクテット分だけゼロという値を返す。

2の累乗に調整するためのMSBのゼロ埋め

デコードされた オクテット 数	得られる オクテット 数	デコードされた オクテットの値	最終的に得られる オクテットの値
0	1		00
1	1	XX	XX
2	2	XX XX	XX XX
3	4	XX XX XX	00 XX XX XX
4	4	XX XX XX XX	XX XX XX XX
5	8	XX XX XX XX XX	00 00 00 XX XX XX XX XX
6	8	XX XX XX XX XX XX	00 00 XX XX XX XX XX XX
7	8	XX XX XX XX XX XX XX	00 XX XX XX XX XX XX XX
8	8	XX XX XX XX XX XX XX XX	XX XX XX XX XX XX XX XX
9	16	XX XX XX XX XX XX XX XX YY	00 00 00 00 00 00 XX XX XX XX XX XX XX YY
10	16	XX XX XX XX XX XX XX XX YY YY	00 00 00 00 00 XX XX XX XX XX XX XX YY YY
11	16	XX XX XX XX XX XX XX XX YY YY YY	00 00 00 00 00 XX XX XX XX XX XX YY YY YY
12	16	XX XX XX XX XX XX XX XX YY YY YY YY	00 00 00 00 XX XX XX XX XX XX YY YY YY YY
13	16	XX XX XX XX XX XX XX XX YY YY YY YY YY	00 00 00 XX XX XX XX XX XX XX YY YY YY YY
14	16	XX XX XX XX XX XX XX XX YY YY YY YY YY YY	00 00 XX XX XX XX XX XX XX YY YY YY YY YY
15	16	XX XX XX XX XX XX XX XX YY YY YY YY YY YY YY	00 XX XX XX XX XX XX XX YY YY YY YY YY YY
16	16	XX XX XX XX XX XX XX XX YY YY YY YY YY YY YY	XX XX XX XX XX XX XX YY YY YY YY YY YY YY

表の例として、16オクテットまでしか記載していないが、17オクテットから先も同じ規則が適用される。例えば、デコードされた、オクテット数が17オクテットまでの値でピリオド(.)に到達した場合であれば、32オクテットの値になる様に15オクテット分MSB側がゼロ埋めされる事になる。

プロパティが要求するオクテット数に対して、<オクテットコード>がもたらす、オクテット数が少ない場合は次の様に処理される。

LSB側を求めるデータのオクテットデータのパターンに合わせる。

不足しているMSB側はゼロで埋められる。

IPアドレスのパラメータの様に、要求するオクテットデータの個数が異なる規格が同じパラメータを共有している場合には、次の規則が適用される。

0個から、4個までならば、IPv4

5個から、16個までならば、IPv6

また、2021年の時点で、IPアドレスもuicodeも128ビット長、つまり16オクテットのデータとなる。従って、16オクテットの値を超える場合には、異常なデータと見做した処理が必要となるが、異常なデータを受け取った場合であっても、携帯情報端末は、ユーザーの許可無く無闇に、その情報を無線などの通信で送ってはならない。これは、わざと異常なデータを格納する事で、携帯情報端末持つ人間が、そのQRコードを端末に読み込ませたかどうかの情報を収集される事を防ぐためである。

繰り返しパターンの指定:

グループCの#17~#30には、同じ値が連続して現れた場合に、データ量を減らす為、繰り返しパターンを指定する為の項目である。

デコードされたオクテットのデータの個数を超える値を参照した場合に、矛盾が生じる事が無い様に、オクテットコードのデコードの先頭には、8オクテット分だけゼロの値がダミーとして入っている事とする。

このダミーのオクテットの値は、オクテットの繰り返しパターンの指定を行う場合にだけ意味を持つ。これにより、例えば、<オクテットコード>の先頭で、Cの#23、「直前の8オクテットのパターンをもう一度繰り返す」を選んだ場合は、8オクテット分、ゼロの値のデータが並ぶ事になる。

繰り返しパターンは、LSB側のオクテットのデータから見た、MSBまでのオクテットの個数となる。繰り返された値はLSB側に追加される。

Cの#17の、「直前の2オクテットのパターンをもう一度繰り返す」場合の例:

直前のオクテットのパターン:

**VV WW XX YY ZZ**

Cの#17が指定された後のパターン:

**VV WW XX YY ZZ YY ZZ**

Cの#18の、「直前の3オクテットのパターンをもう一度繰り返す」場合の例:

直前のオクテットのパターン:

**VV WW XX YY ZZ**

Cの#17が指定された後のパターン:

**VV WW XX YY ZZ XX YY ZZ**

Cの#24～29は、直前の1オクテットの値を複数回繰り返すパターンを指定するものである。繰り返し回数は2～7まで用意している。

Cの#24の「直前の1オクテットの値を2回繰り返す」場合の例:

直前のオクテットのパターン:

VV WW XX YY **ZZ**

Cの#24が指定された後のパターン:

VV WW XX YY **ZZ ZZ ZZ**

Cの#25の「直前の1オクテットの値を3回繰り返す」場合の例:

直前のオクテットのパターン:

VV WW XX YY **ZZ**

Cの#25が指定された後のパターン:

VV WW XX YY **ZZ ZZ ZZ ZZ**

Cの#30は、直前の2オクテット前の1オクテットの値を2回繰り返すパターンを指定するものである。

直前のオクテットのパターン:

VV WW XX **YY ZZ**

Cの#30が指定された後のパターン:

VV WW XX **YY ZZ YY YY**



基準点からの差分値による指定:

基準点からの差分による指定の基本的な考え方は、円上に、0～255の値を等間隔で刻み、基準点設けて、次のオクテットの値を、この基準点に対して回す量で指定する考え方である。回す量は、<High-density-table>または、<Low-density-table>に含まれるグループAとグループBの値で指定する事となる。

<High-density-table>は、細かく回す量を指定する場合に効率適なテーブルである。または、<Low-density-table>は、粗く回す量を指定する場合に効率適なテーブルである。

次の1オクテットの値は、基準点の値から、テーブルから得られた値を足した後に、0～255に収まる様に $\&= 0xUFF;$ でビット毎の論理積を取った値が適用される。

<オクテットコード>の始まりでは、<High-density-table>を利用し、基準点は、初期値をゼロとし、以降は、直前のオクテットの値となる。

また、データの種類の合合わせ、基準点の更新規則や、テーブルの切り替え規則をグループCの<Control-table>から変更できる仕様となっている。

グループAとグループBで表現に必要なデータ数が倍の差があるために、1オクテットのデータを表現する場合、グループAの範囲に値が入った場合の方が、グループBの範囲に入った場合よりも、少ないデータの量で表現する事ができる。しかし、グループAの項目数は35に限られている為、そのまま、単純に値を割り振った場合には、 $35/256=0.140625$ の割合で、小さなグループAの項目が選択される事となる。そこで、グループAの項目は、前の値を基準点とし、差分の値を指定する事にした。こ

の規則により、<High-density-table>が適用される場合には、前の値から、-17~0~+18の値の場合に、グループAの項目が適用されるものとする。

もし、この戦略が外れた場合には、グループBの項目が適用されるが、その後、適用されるテーブルは、<High-density-table>から、<Low-density-table>に切り替わる。

<Low-density-table>側のテーブルが利用される場合にも、同じ規則が適用され、グループBの項目が適用された後は、その後使用されるテーブルは、<Low-density-table>から<High-density-table>側のテーブルに切り替わる。

繰り返しになるが、データの種類の合わせ、基準点の更新規則や、テーブルの切り替え規則をグループCの<Control-table>から変更できる仕様となっている。

オクテットのデータをバイナリモードを利用せずに、<オクテットコード>を採用した理由:

<オクテットコード>を定義した理由は、QRコードで、バイナリモードに切り替えた場合、また、英数字モードに戻す必要もあるために、オーバーヘッドが非常に大きくなる。また、バイナリモードでオクテットデータを入れた場合は、UTF-8のエンコード専用で使用できなくなる為、構文解析エンジンがより複雑になるという課題も生じる。総合的に判断した結果、バイナリモードはUTF-8専用にした方が良いと判断した為である。

バイナリモードに切り替えてから、英数字モードに戻すのに必要となるオーバーヘッドは、QRコードのバージョンに依存する。

バージョン1～9までが25ビット、バージョン10～26までが、35ビット、バージョン27～40までが37ビットである。

4オクテットのパターンのデータの場合は、バイナリに切り替えた場合には、バイナリデータそのものは、32ビットであるが、オーバーヘッドが生じるため、バージョン1～9まで57ビット、バージョン10～26までが、67ビット、バージョン27～40までが69ビットとなる。

これに対し、<オクテットコード>が、プロパティの最後にある場合には、プロパティの終了を示すピリオド(.)が<オクテットコード>の終了を示す文字を兼ねる為、バイナリモードに切り替えて再度、英数字モードに切り替える場合と異なり、ヘッダーやフッターによるオーバーヘッドは一切無いが、1オクテット当たり最悪値では、11ビットとなる為QRコードのバージョンに限らず常に、44ビットが最悪値となる。

従って4オクテットまでのデータであれば、<オクテットコード>の方が必ず小さなデータ量に抑える事ができる。

16オクテットの場合、バイナリよりも小さくなるかは、データパターンに依存する。

バージョン1～9まで153ビット、バージョン10～26までが、163ビット、バージョン27～40までが165ビットとなる。

<オクテットコード>が、プロパティの最後にあり、<High-density-table>または<Low-density-table>のグループBだけを利用して、エンコードされた場合には、最悪値、 $(0*5.5)+(16*11)=176$ ビットなり、バイナリモードを利用した場合よりも効率が落ちるが、ここで、仮に、16オクテット中の5オクテット分だけ、<High-density-table>または<Low-density-table>に含まれるグループAの項目を利用してエンコードできたと仮定した場合には、 $(5*5.5)+(11*11)=27.5+121=148.5$ ビットとなり、バイナリモードに切り替えた場合よりも効率が高くなる。また、グループCでは、多数の規則を用意しているため、16オクテットのデータであれば、データの一部に規則性が生まれ、組み合わせが膨大な数となる為、数学的検証は行っていないが<オクテットコード>の方が、効率が上が高くなる確率が高くなると考えられる。

以降に、グループAとCの参照回数と、データ量をまとめた表を示す。

バイナリモードのオーバーヘッドを考慮した4オクテットに対するビット数

オクテット\QRコードのバージョン	1~9	10~26	27~40
4	57	67	69

<オクテットエンコード>4オクテットに対するビット数

グループAを参照した回数\ グループCを参照した回数	0	1	2	3
0	44	55	66	77
1	38.5	49.5	60.5	71.5
2	33	44	55	66
3	27.5	38.5	49.5	60.5
4	22	33	44	55

この表のグループCを参照した回数は、テーブルの切替規則または、基準点の切替規則を変更する為に、グループCの#0~#16のどれかを参照した回数の事である。グループCの#17以降を参照した回数は含まれないものとする。

4オクテットのデータをエンコードする場合は、グループCの#0~#16は、1回までに抑え無いと、逆にデータ量が増え効率が下がる事となる。また、グループCを参照する回数が0回の場合に、グループAが参照できる回数が、1回以下の場合には、グループCを1回参照して規則を切替える事で、グループAを参照できる回数が、3回以上の場合は、効率が上がり、必要なデータの量がより少なくなる。

何方の場合であっても、<オクテットコード>の仕組みを利用した方が、バイナリモードと英数字モードを切替えるよりも、4オクテットまでのデータであれば常に必要なデータ量がより少なくなる。

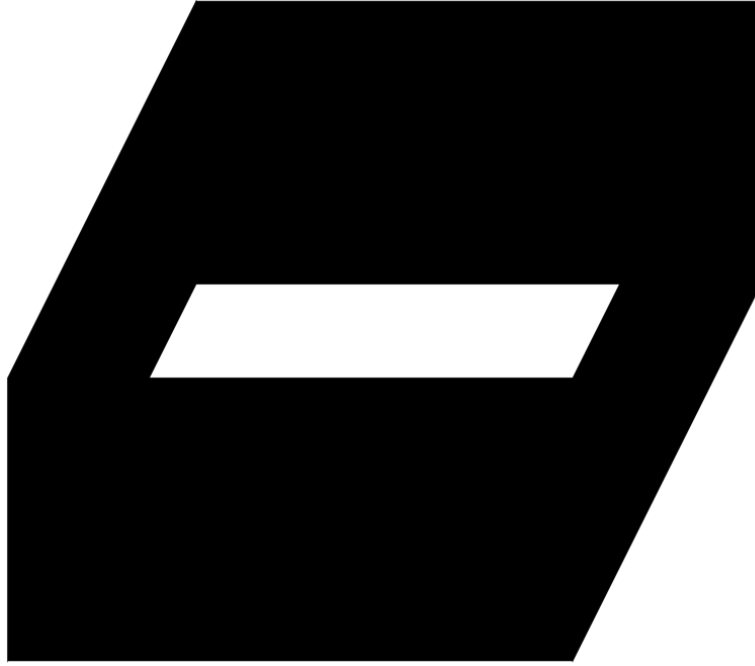
バイナリモードのオーバーヘッドを考慮した16オクテットに対するビット数

オクテット\QRコードのバージョン	1~9	10~26	27~40
16	153	163	165

<オクテットエンコード>16オクテットに対するビット数

グループAを参照した回数 \グループCを参照した回数	0	1	2	3	4	5	6	7
0	176	187	198	209	220	231	242	253
1	170.5	181.5	192.5	203.5	214.5	225.5	236.5	247.5
2	165	176	187	198	209	220	231	242
3	159.5	170.5	181.5	192.5	203.5	214.5	225.5	236.5
4	154	165	176	187	198	209	220	231
5	148.5	159.5	170.5	181.5	192.5	203.5	214.5	225.5
6	143	154	165	176	187	198	209	220
7	137.5	148.5	159.5	170.5	181.5	192.5	203.5	214.5
8	132	143	154	165	176	187	198	209
9	126.5	137.5	148.5	159.5	170.5	181.5	192.5	203.5
10	121	132	143	154	165	176	187	198
11	115.5	126.5	137.5	148.5	159.5	170.5	181.5	192.5
12	110	121	132	143	154	165	176	187
13	104.5	115.5	126.5	137.5	148.5	159.5	170.5	181.5
14	99	110	121	132	143	154	165	176
15	93.5	104.5	115.5	126.5	137.5	148.5	159.5	170.5
16	88	99	110	121	132	143	154	165

16オクテットの場合は、データがグループAを参照できる回数により、バイナリモードと英数字モードを相互に切替た方がデータの量が少なくなる場合もあり得るが、グループCでは多数の規則の変更ができる為、16オクテット分のデータであれば、どれもが上手く機能しない確率は、かなり低くなり、多くの場合で同等または、より少ないデータに抑える事ができると考えられる。



LPS,GMS共通のデータの書式:

<識別子>:<プロパティ 1><プロパティ 2>...<プロパティ n>.

識別子は、「識別子名」のアルファベットで始める名前で始まり、  
コロン「:」で<プロパティ 1>と区切られる。

<プロパティ n>の後ろにピリオド「.」を付ける事で、識別子が示す  
データ構造の終わりを示す。

プロパティ:

プロパティは、下記の項目から構成される。

<プロパティ名>:<データ>.

プロパティ名:

プロパティ名は、アルファベットから始まりコロン「:」により、<  
データ>と区切られる。

データ:

データは、コロン「:」の後から始まり、ピリオド「.」の前で終わる。  
必要に応じて、データ中に、別の複数個のプロパティを入れ子構造と  
して埋め込む事ができる。

※1つの識別子の中に含まれるコロン「:」とピリオド「.」の個数は、エス  
ケープ文字が利用された場合を除き一致する事になる。



複数の識別子が含まれている場合の書式:

<識別子1>:<プロパティ1><プロパティ2>...<プロパティn>.<識別子2>:<プロパティ1><プロパティ2>...<プロパティn>...<識別子n>:<プロパティ1><プロパティ2>...<プロパティn>.

ピリオド(.)により、識別子で指定したデータ構造が終わった後に、別の識別子続ける事ができる。

LPSやGMSが管理を行わないデータを入れる場合の書式:

<識別子1>:<プロパティ1><プロパティ2>...<プロパティn>..<管理外のデータ>

ピリオド(.)により、識別子で指定したデータ構造が終わった後に、もう一度ピリオド(.)が指定されている場合には、LPSおよびGMSのデータは、この後に続いて現れない事が保証される。この後に続くデータについては、本規格の<管理外のデータ>とする。<管理外のデータ>の用途については、自由な用途に使う事ができるものとする。

エスケープ文字の記述方法:

データの中で、文字列の一部として「/」、「:」、「.」を利用する場合には、エスケープ文字として記述する必要がある。

「/」は、「//」と記述する。

「:」は、「/:」と記述する。

「.」は、「/.」と記述する。

英数字モードで扱えない文字をエスケープ文字で表記する方法:

英数字モードから、他のモードに切り替える事なく、英数字範囲外のUS-ASCIIに含まれる制御文字以外の文字とローマ字の長音文字を扱う方法を定義する。この規定は、あくまでデータの量を減らす為の手段であり、他の漢字やバイナリモードの使用を禁止するものではない。

英文字の大文字、小文字の切り替え「/-」:

後続の英字に対し、大文字として解釈されるのか、小文字として解釈するのかを切り替える。

切り替えの有効範囲は限定的なものとし、各プロパティの内部に、留まるものとする。また、英数字モードから他のモードに切り替えた場合は、自動的に大文字に戻るものとする。

初期値は大文字であり、「/-」の記述がある度に、大文字から、小文字、小文字から、大文字に切り替わる。

英数字モードでは扱えないその他の記号:

「!」は、「/B」と記述する。

「?」は、「/C」と記述する。

「”」は、「/2」と記述する。

「#」は、「/3」と記述する。

「&」は、「/6」と記述する。

「'」は、「/7」と記述する。

「(」は、「/8」と記述する。

「)」は、「/9」と記述する。

「,」は、「/4」と記述する。

「;」は、「/F」と記述する。

「<」は、「/G」と記述する。

「=」は、「/H」と記述する。

「>」は、「/5」と記述する。

「?»は、「/J」と記述する。

「@」は、「/K」と記述する。

「[」は、「/L」と記述する。

「\」は、「/M」と記述する。

「]」は、「/N」と記述する。

「^」は、「/Z」と記述する。

「\_」は、「/P」と記述する。

「`」は、「/Q」と記述する。

「{」は、「/V」と記述する。

「}」は、「/W」と記述する。

「|」は、「/X」と記述する。

「~」は、「/Y」と記述する。

ローマ字の長音記号:

「Â」は、「/A」と記述する。

「Ê」は、「/E」と記述する。

「Î」は、「/I」と記述する。

「Ô」は、「/O」と記述する。

「Û」は、「/U」と記述する。

※LPSやGMSで共通に使用されるデータの書式で使用される文字は、QRコードの英数字符号化(Alphanumeric)の範囲で利用できる文字に限定する事により、少ないデータサイズ(2文字で11ビット)に収まるように設計されているため、アルファベットの小文字は、表現出来ない。また、利用できる記号は以下に示す9つの記号に限られている。

- (1) 空白「 」
- (2) ドル「\$」
- (3) パーセント「%」
- (4) アスタリスク「\*」
- (5) プラス「+」
- (6) マイナス「-」
- (7) ピリオド「.」
- (8) スラッシュ「/」
- (9) コロン「:」

※その他の文字を利用する為には、他のモードに切り替える事が必要となるが、QRコードのモード切り替えを含んだ、エンコード方式は、同じモードの範囲の文字が連続した場合に、一番効率的に機能する設計となっており、頻繁なモード

切り替えた場合には、モードインジケータと、文字数インジケータが挿入された分だけ、ビット数が増え、合計ビット数が増加する。

定義済みの識別子:

Local Positioning Systemの識別子は、「LPS」とする。

Guide Mark Systemの識別子は、「GMS」とする。

Private Guide Mark System の識別子は、「PGS」とする。

User Guide Mark System の識別子は、「UGS」とする。

Service Guide Mark System の識別子は、「SGS」とする。

Food Allergies Digital Marking System の識別子は、「FAS」とする。

Digital Certificate System の識別子は、「DCS」とする。

規格の改定が発生した場合には、上記の識別子名の後ろに、アラビア数字でバージョンを指定するものとする。

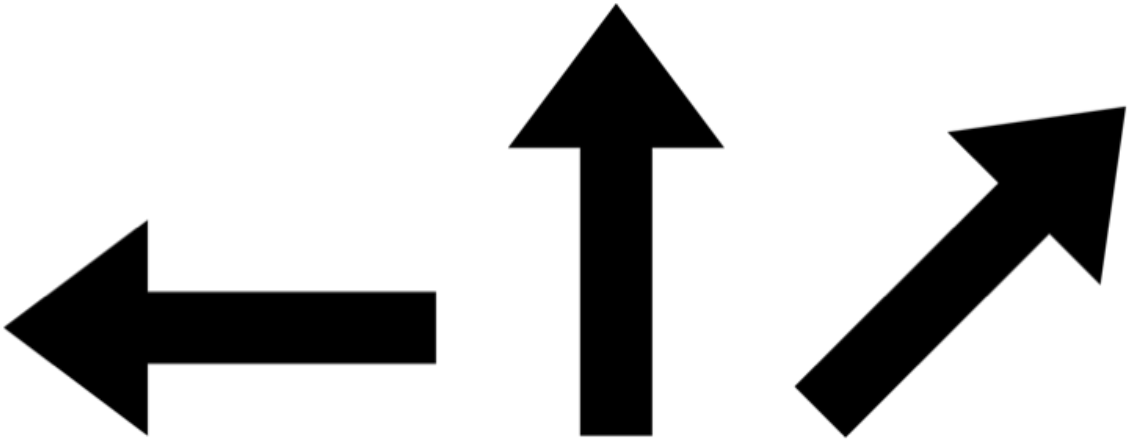
0番は、一番最初のバージョンを示し、全て識別子でバージョン番号が省かれている場合には、0番を指定した事となる。

QRコードの中で、前の識別子にアラビア数字でバージョンが指定されていて、後続の識別子のバージョンが省略されている場合には、後続の識別子には、前の識別子で指定されたバージョンが適用されるものとする。

※ (不等号)より小「<」、(不等号)より大「>」を本文中では、項目を示す為の山括弧代わりのメタ文字として使用している。実際には、この文字をQRコードのデータとして直接入れるわけではない。

※ソフトウェアが登録に無い<識別子>や<プロパティ名>を検出した場合は、中身は解釈できなくても、書式のデータ構造に従って読み進める事とする。





## 9. LPSの案内板に対する適用

公共のスペース、公共の交通機関の案内板は、多言語で記載される事が増えているが、表記する言語の数が増えるほど、誰もが不便な思いをする案内板となってしまいう矛盾を抱えている。

案内板の役割は、情報を必要とする人に対して、適切な情報を提供することである。対応する言語の数を増やす為に、一つの案内板に、多数の言語の表記を入れる事は、逆に多数の人には、適切では無い情報が多く含まれる事になる。

案内板の役割は、必要な時に、必要な場所で、必要な情報を提供する事である。

### 9.1 日本国内における案内板の多言語対応の現状

現在、日本国内で利用されている多言語の案内板は、多くの場合、2つに分けられる。

(1) 日本語と英語の併記

(2) 日本語と英語、韓国語、中文(繁体字)、中文(簡体字)の併記

このうち、(1)は、大抵の場合、日本人にも、読み易く、海外の方にも配慮されたバランスの良いものとなっている。

これに対して特に(2)は、次の述べる複数の問題点を持っている。

## 9.2 多言語対応案内板の課題

各言語の文字が小さくなり、どこに自分が読める言語で書かれているのかを見つけるために時間を要する。つまり、案内板に記載される言語の数が加くなるほど、人が認識に必要な時間が増える。

また、案内板の大きさには物理的な制限があり、対応させる言語の数を増やすほど、各言語の文字が小さくなり遠くから読めなくなる。

また、日本で行われている多言語化は、対応している言語が不足し過ぎている。

G8(フランス、アメリカ、イギリス、ドイツ、日本、イタリア、カナダ、ロシア)どころか、G7(フランス、アメリカ、イギリス、ドイツ、日本、イタリア、カナダ)の参加国が使用している言語すら網羅できていない。

更に、ネイティブスピーカーが多い言語で母語話者1億人以上の言語に限定した場合であっても、11言語に達する。

Google社が提供する自動翻訳サービスに限定した場合でも2021年6月現在、108言語に対応している。また、Wikipediaで「言語」の項目を調べて見ると230項目検索結果出てくる為、Wikipediaだけで使用されている言語に絞った場合には、230の言語が利用されていると考えられる。しかし、単純に、同じ案内板の面積に、230通りの言語で書き込んだ場合、1言語あたりの面積は、1つの言語を書き込む場合に比べて、230分の1のサイズになってしまう。

そのため、230の言語の中から自分が読める言語で書かれている部分を見つけて出す作業も発生する。従って、仮に230通りの言語を平等に扱う案内標識を作成した場合は、誰もが読む事に苦勞する非実用的な芸術作品となると考えられるのでこの様な方法は避けるべきである。

2021年現在は新型コロナウイルスの蔓延により、海外からの旅行客需要が極端に減少してしまっているが、今後、効果的なワクチンの接種と、コロナ

ウイルスは変異により、最終的に弱毒化する方向に進む事になる為、何年後になるかは、不明であるが、再び海外旅行が活発に行われる時代が訪れるはずである。

将来的に、様々な海外の方が日本に訪れる状態になった際に、現在のたったの5種類の文字だけに対応する事は、今後の旅客動向の推移次第では不十分となる事が考えられる。また、日本の多言語案内の現状が、言語が壁となり、大きな旅客需要を見逃していることも考えられるので、今後新たに、来日者が増えた国に合わせて、更に多数の言語を案内板に加える方法を検討してみる事とする。

### 9.3 案内板の面積を分割する方法

案内板に標記する言語の種類を限定し、優先順位を付けて、優先順位に応じて文字の大きさを変える方法。

各言語の文字の大きさを同一にする方式や、面積を同一にする方式よりも、現実的な手段ではあるが、対応させる言語の数を増やす程、可読性が減り、遠くからは読めなくなる。

また、読める言語を探すまでに時間が掛かるので、2桁や3桁の言語を盛り込んだ場合は、間違いなく案内板としての役割を果たせなくなる。

本質的に、人は、同時に表示される項目の数が多くなる程、認識に掛かる時間が増えてしまう為、多くの言語の標記する事は適切では無い。

この方法が無理なく機能するのは、2言語までと言える。

従って多言語のガイドラインで、日本語と英語の併記を指示している事は妥当であると考えられる。

### 9.4 案内板を時分割で表示を切り替える方法

LEDディスプレイや液晶ディスプレイ、電動膜などを利用し、時分割で、言語を切り替えながら表示するという手段を利用する方法が考えられる。この場合は、表示する言語の数が増える程、その案内板を見たタイミングで必要とする情報が入手できる機会が下がるという問題を持っている。

1言語につき5秒間表示し、次の言語に切り替えるというルールで230通りの言語を切り替えながら、表示した場合を仮定した場合、1周するまでに、1,150秒掛かる事になる。分数に直すと約19分となり、この方式も対応する言語の数が多くなれば成る程、実用性が皆無となる。

### 9.5 案内板の時分割と面積分割の併用

時分割の中で、更に画面を分解する方式もあるが、これは、それぞれの方式の問題点を合わせた方式であると言う事もできる。

## 9.6 特定の言語に依存しない案内

ピクトグラムやロゴ、路線駅番号を活用する方法は、特定の言語に縛られる事なく、情報を伝える事ができる優れた手段である。しかし、この方法は、地名や建物の名前等、ピクトグラムなどが無い、固有の名称を伝える事が目的の場合は利用できない手法でもある。また、解像度の低い電光掲示板では、ピクトグラムを表示できない場合もある。

## 9.7 多くの人に役立つ多言語案内板

本来の意味で、多くの人に役立つ案内板にする為には、一つの案内板に、多くの言語を盛り込むという方法を見直すべきである。

多くの地域の人から分かり易い案内板を実現する為には、案内板に記載する言語は以下の文字と言語に限定するべきである。

- (1) ピクトグラム: 表記したい者に対応するピクトグラムがある場合のみ表記する
- (2) 現地の公用語: 日本であれば日本語
- (3) 英語: 世界で最も多くの国や地域で使用されている言語であるため

英語には無い固有名詞などは、ラテン文字化し、(日本では、ローマ字表記と一般的に呼ぶ)英語と書きませ表記とする。ラテン文字化した固有名詞は、英単語と区別するため、全て大文字表記とし、英単語は、先頭の1文字を除き、原則小文字で表記する。

上記の以外の言語を無理やり案内板に詰め込むと、現地の人にとっても、海外の人にとっても分かり辛く、不便な案内板となる。

また、英語と現地の公用語で利用される文字の字形が同一か、一部重なっている場合は、この2つの言語で書かれた文字列は文字の大きさを極端に変える

(現地公用語方を大きくする)文字列同士の距離を大きくとるなどの工夫が必要になる。

案内板は、まず、現地人にとって分かり易く、便利なものであることが求められる。その次に、海外の旅行客にもわかり易く、便利なものである事が求められる。

## 9.8 IT技術を利用した動的表示言語の切り替え

案内板がLEDや液晶を利用し、動的に書き換えが可能な仕組みであり、案内板を見ている人が読む事が可能な言語を知る術があれば、表示言語を切り替える方法も考えられる。この方式は、その案内板の表示を同時に見る人が読める言語の数が鍵となる。

同時に見る人が読める言語の数が一致している場合は、有効に機能するが、読む事ができる言語が異なる場合は、時分割や、領域分割が必要となるので、やはり、必要な情報を見つけるのに掛かる時間が増るか、遠くからは見えないという問題は残るが、単純に、来訪者の言語を予め網羅する前述の方式よりも表示する言語を絞る事ができる為、状況次第では上手く機能する方法と言える。

例えば、100人が同時に観ている状況下で、100人全員が、イギリス人の場合は、英語だけを表示すれば良い。また、100人全員が、日本人であれば日本語だけ表示すれば良い。逆に、100人全員が別の国出身で、読める言語が全員異なる場合には、この方式も破綻する事となる。

逆に、案内板を見る人数が1人であれば、1人の母国語だけを表示する事ができるので、この様な状況では、最も効果的に機能する案内板として機能する可能性もある。

## 9.9 画像認識型翻訳サービスの利用

案内板の情報を画像として情報携帯端末からイメージセンサーにより収集し、端末内部、または、ネット上の翻訳サービスを利用して、情報携帯端末を持つ者の母国語に翻訳し、結果を文字情報や、音声情報として返す方式である。

この方法は、画像情報を文字情報に変換し、文字情報を母国語に変換という二つの過程を必要とするため、必然的に変換ミスによる誤訳が発生する可能



性を含んでいるが、案内板に標記する言語は1つで済む為、案内板の内容を遠くからも確認できるというメリットと、翻訳サービスが提供可能な言語の組み合わせだけ、対応できる言語がそのまま増えるというメリットを持つ。また、その逆に、情報携帯端を持たない者は、案内板に直接書かれた限られた言語を読めないと対応ができない事にも繋がる。

#### 9.10 位置情報と方位情報に基づく情報携帯端末のよる案内の提示

LPSを応用すれば、屋内であっても情報携帯端末を持つ者の位置と方位の特定が容易にできるようになる。

1つの案内板の限られた、空間に2桁や3桁の言語を直接盛り込んだ場合は、実用性が無くなってしまいが、位置情報を盛り込んだQRコードを案内板または、その周囲に記載し、情報端末からQRコードに対し、登録された情報を、提供する方法であれば、実用的なレベルで、各言語に合わせた適切な情報を提供する事ができるようになると考えられる。また、この方式で、提供される言語の対応に、情報携帯端末所有者が読解できる言語が含まれていない場合は、翻訳サービスを経由でアクセスさせれば、翻訳サービスが提供する言語の数だけ、対応する言語を、増やす事ができる様になる。

拡張現実の技術と組み合わせて、情報端末に母国語の情報を登録することで、LPSに対応する案内板は、どの国の方でも、全て母国語で記載されているように見ることが可能になる。

また、この仕組みを音声読み上げのサービスやアプリと組み合わせれば、視覚に障害を持つ方にも、適切な情報を提供することができる様になるため、言語の壁による障害だけでなく、視覚の障害に対する壁も取り除くことができるようになる。

また、このような仕組みを構築するには、LPSよりも、トロンフォーラム傘下のユピキタスIDセンターが提供するucodeの仕組みを直接活用の方が向いているといえる。

また、大都市の案内板には、既に十分な、空きスペースが残っていない。案内板にLPSの情報が入ったQRコードを直接入れる事は、あくまで概念であり、実際の適用としては、新規に案内板を作る機会がある場合を除き、案内板自体ではなく、周囲の柱や、地面(床)、天井などの空いているスペースに適用すべきである。案内板には、現時点で、既にLPS用のQRコードを入れる余白が無い事の方が大半である。その一方で、柱や、地面(床)には、まだ、QRコードなどを印刷する余地が残されている。

#### 6.11 場所に合わせた注意事項

LPS やGMSのデータが入ったQRコードは、利用する場所に合わせた配慮が必要とされる。

屋外や、屋内でも窓などの開口部がある場合:

日中は、陽射しの影響を考慮する必要がある。逆に夜間は、車道と隣接している場合には、ヘッドライトの影響を考慮する必要がある。

これらの光源は、季節や時間帯によっても変化するため、この変化の影響も考慮が必要となる。

特に、壁面に配置された場合、QRコードの印刷物自身や、その上に、光沢の強い素材、ガラス板等が被さっている場合は、照り返しの影響を受けて、読み取りができなくなる事が考えられる。

天井に設置する場合:

窓などの開口部がある場合には、日中は、陽射しの影響を考慮する必要がある。また、天井に照明がある場合には、照明との距離も考慮すべきである。光量が強い光源の側にQRコードを配置した場合には、

イメージセンサー及び、補正処理の仕組みのよっては、読み取れない事が考えられる。

また、天井は、高さが高いほど、設置に掛かるコストが高くなると考えられる。

地面に設置する場合:

LPS やGMSのデータが入ったQRコードを、地面に対して直接印刷する場合や、印刷物を直接地面に貼る場合は、天井や壁に印刷した場合に比べて、摩耗による破損や、表面にゴミが被さる事が増えるため、マークの一部が破損や、ゴミで隠される事により、QRコードの認識率が低下させるように、印刷する面積を大きくし、誤り訂正能力の大きなレベルを選択するなどの対応が必要になると考えられる(物理的にQRコードを構成する、モジュールのサイズを大きくすれば、小さなゴミが重なった場合であっても、モジュールの大きさがゴミよりもはるかに大きければ、ゴミの影響を画像処理により排除する事が可能となる。また、逆にゴミの大きさが、モジュールのサイズよりも大きなサイズとなった場合には、1つのゴミにより、複数のモジュールが隠れて読み取り不能となる。

また、QRコードの上を通過する人や車などがスリップし難い素材である事が求められる。また、厚みが大きなもの場合は、躓く事による事故の原因ならない厚さに止めるべきである。

摩耗の発生を前提とした配慮が必要となる。

定期的を確認し、機能を失う前に張り替え作業を行う方法と、その設備のライフサイクル全体を考慮した、十分な、耐摩耗性を持たせる方法が考えられる。

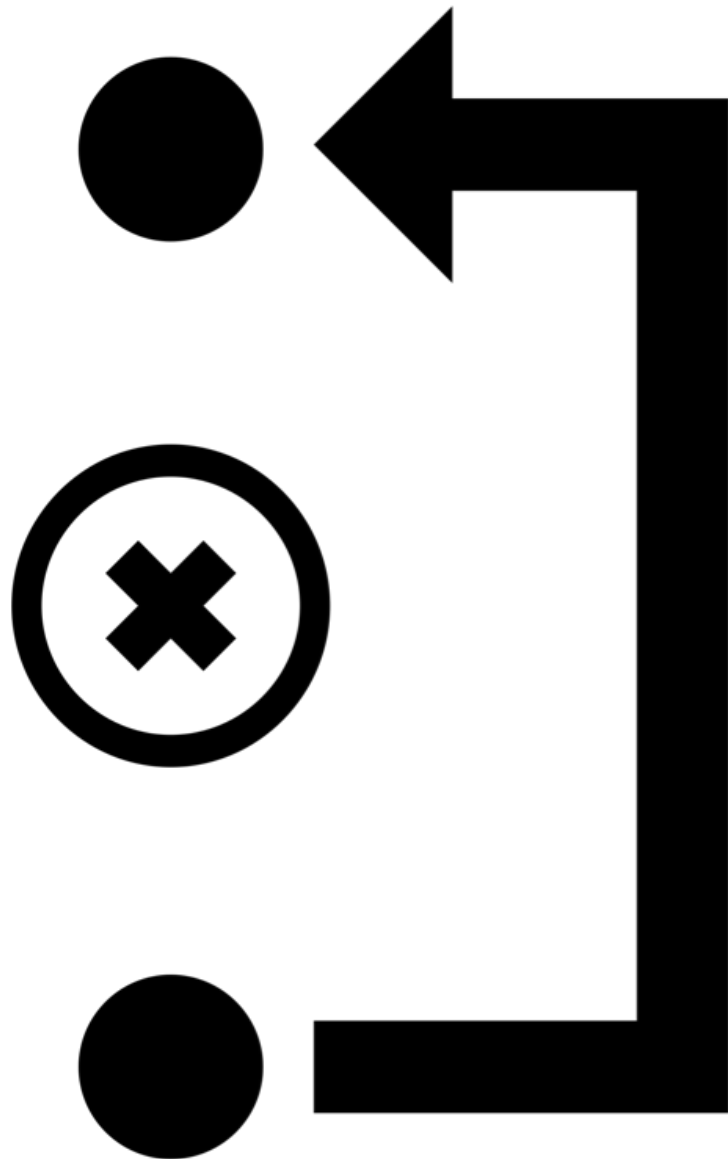
## 視覚障害者誘導ブロック (点字ブロック):

QRコードを直接印刷する場合や、印刷されたシールを貼り付ける場合には、点字ブロックと被らない様に配慮が求められる。視覚障害者誘導ブロックは、形状だけが意味を持つものではない。色により、弱視者を助ける役割も持っている。従って、QRコードが視覚障害者誘導ブロックの上に被った場合、弱視者に不利益を及ぼす事態が発生すると考えられるため、必ず、視覚障害者誘導ブロックの上には被らない様に印刷すべきである。また、仮に、障害者誘導ブロックの上に、QRコードを入れた場合は、QRコードのサイズが小さくなり、ある程度離れた距離からでも、検出したいのに、自分で細かくQRコードの向きに合わせて、イメージセンサーを近付ける作業が必要になる事が考えられる。また、凹部分は、構造上ゴミが溜まり易く、清掃が困難なため、読み取りミスが発生する原因となる。QRコード自信に誤り検出訂正の機能が含まれているので、訂正能力を越えた場合は読み取りエラーとして扱われる事になると考えられる。

そして、視覚障害者誘導ブロックは主要な用途として、白杖により、存在を確認される為、頻繁に白杖と接触する事になり、表面に印刷する手法で作成された物の場合は、削れる事による破損が生じる事が考えられる。

従って、視覚障害者誘導ブロックとその周囲から、適切に距離をとって設置した方が、メンテナンスのコストも下げる事ができると考えられる。

# GMS



## G. 電波到達圏外での運用をする為の案内マークシステムGMS (Guide Map System)

GPSが受信できない屋内、地下街、トンネルのような場所では、携帯電話の電波を受信できる保証は無いまた、仮に携帯電話の電波を受信できる場所であっても、緊急事態が発生した場合や、人が密集する場所では、携帯電話の電波による通信が成立しない状況が発生すると考えられる。

また、飛行機の機内などでは、携帯電話回線を利用した通信行為その物に規制がある場合も考えられる。この様なスタンドアローン状況下では、データベースとの連携を前提とした、シリアル番号ベースのシステムを運用する場合、スタンドアローンの状況下に置かれる前に、予め、その直近の地域で、受け取る可能性が大きい、シリアル番号に対応する情報を、データベースのサブセットをキャッシュとしてダウンロードして置く方法が考えられる。

しかし、この方法は、通信が途絶した状況で予めキャッシュした範囲を超えた際に、有効に機能しなくなる。震災が発生した場合には、被災地域全域で、通信が途絶し、突然スタンドアローンの状態に置かれる事も考えられる。データベースのキャッシュが無い状況下でも、有効に機能させる方法として、QRコード自身に、必要最低限の案内を行う情報を埋め込む方法が考えられる。

QRコードを利用して道案内を実現する為のシステムを以降、本文では、案内マップシステムGMS (Guide Map System)と呼ぶ事とする。

GMSは、LPSのフォーマットをベースとし、様々な情報を提供する事を目的としている。

## G.1 案内マップシステムGMS(Guide Map System)

GMSの基本的な考え方は、マークアップである。GMSは、案内板などの限られた面積に印刷し、離れた位置から、情報を取得する使い方を想定しているため、できる限り、データサイズを小さくする事が求められる。

現在地から、目的地に移動した際に、道案内に必要とされている情報は、その地域や地球上の全て事細かな正確な情報を全て盛り込んだ平面図では無い。現在の地点の周囲と、目的地へ至るまでの方向や道のりの情報である。

案内板の情報や地図を画像データとして用意した場合には、視覚障害者の壁となる。1つのQRコードで提供できるデータ容量の上限バイナリで、2,953バイトの問題も発生する（ファイルサイズを小さくできるGIF形で400x200ピクセルを3Kバイトとなり、QRコードが提供できる情報量を画像データだけで超える事が考えられるので、ピクセルデータの画像情報として案内板の情報を保存する方法は、不適切である）。この問題を解決する方法として、フロアマップの画像を提供するのではなく、案内板の情報を、マークアップし、ベクター情報と文字列の組み合わせで、情報を提供する方法が考えられる。

※QRコードの仕様上では、最大16分割(連結)が可能となっている。型番(バージョン)40、誤り訂正レベルLの指定のQRコードを、最大16個連結する事ができるのであれば、 $2,953 \times 16$ バイト=47,248バイト、約46kバイトまでの情報を提供する事ができる。しかし、印刷が必要なQRコードの数が増えるほど、同じ面積に印刷した際に、同じ性能の情報端末から、認識できる距離が短くなってしまう。

## G.1.1 GMSとLPSの関係

GMSとLPSはそれぞれが独立した規格として単体でも利用できる。

GMSとLPSを連係して利用する事もできる。

### G1.1.1 GMSとLPSを連携して利用する場合

GMS地球上のある特定の位置と結び付けて運用したい場合は、LPSと合わせて利用すべきである。GMSのデータの書式は、LPSと共通の書式となっているため、LPSとGMSを1つのQRコードに入れる事もできる。また、連結先のQRコードのある方向と距離の情報を、入れる事で、LPSでQRコードを分ける事もできる。そしてLPSが入ったQRコードは、LPS単独のデータだけを保存し、GMSが入ったQRコードを連結先とし、その一方で、GMSの入ったQRコードには、LPSとGMSの両方のデータを入れておき、LPSとGMSの両方のデータが入ったQRコードから、他のQRコードに対する連結は行わないといった、一方通行のQRコードの連結も許すものとする。

LPSと連係して利用する場合は、位置情報と関連付けを行う事ができる様になる為、GMSの情報は、LPSの公開領域属性の情報に従って扱われるべきものと考えられる。

LPSのデータが入ったQRコードが複数ある場合には、MGRSに従っている為、1.5mを水平方向に超えた位置にあるものに対しては、同一のデータを持つものは存在してはならない。

垂直方向の同一フロアの壁に同じサイズのものが縦一直線に並べられている場合や、1mよりも近い距離に配置されている場合にはMGRSグリッドの位置によっては、水平方向でも同一のデータのものを含む事ができる。



### G1.1.2 GMS単体で利用する場合

GMSは公共交通機関の乗り物の内部で単体で利用する場合の用途も考慮し、  
相対座標指定としている。

LPSと関係しない場合、GMSの情報はデフォルトで、公開領域属性が不明と  
して扱われるものとする。

GMSが単体で利用される場合には、位置への紐付けが相対的にしか行われな  
いため、全く同一の内容が印刷されたQRコードが、世界中の全く異なる座標  
の位置に複数存在する事があり得る。

※列車内、飛行機機内、船内、バスの車内では、絶対座標が利用できない。  
相対座標であれば適用できる。

## G.2 マークアップする対象と種別

最低でも以下に示すものは、マークアップできる必要がある。

- 現在地
- 入口/出口
- 改札口
- 非常口
- 緊急時の指示
- 通常時の指示
- 鍵
- ドアノブの位置
- 操作板の位置
- ボタン
- トイレ:男性用/・女性用・共用・多目的...
- バスルーム
- 階段: 上・下
- エスカレーター: 上・下
- エレベータ
- 駅・乗り場
- 歩道
- 横断歩道
- 動く歩道
- 信号機
- 踏切
- 点字ブロックの位置
- 重要なもの (ランドマーク)
- 周囲の障害物、危険物
- 地点や施設の名称

この内、出入り口、エスカレーター、動く歩道、改札口は、方向性を持つ場合があり、場合によっては、方向性が切り替わる場合がある。

### G.3 固有名詞を持つ場合の指定名称

現地の公用語（日本語）、アメリカ英語、発音（固有名詞の場合はローマ字表記）

※現地の公用語がアメリカ英語の場合は、下記の情報を登録するものとする。

アメリカ英語、発音（固有名詞の場合はローマ字表記）

※現地の公用語が、複数に渡る場合には、その地域の定める順序で、複数の公用語を記録することができるものとする。

※公用語がUS-ASCIIやShift-JISでは表現できない言語の場合は、QRコードで指定された文字列として記録することができない。このような場合は、QRコード全体としては、バイナリのモードを利用し、バイナリデータの中身はUnicode（UTF-8）として取扱う方法が考えられる。

#### G.4 基点

QRコードからの原点位置のオフセットを指定できる。

床面から約90度に近い角度に配置されているものは、QRコード正面位置から60cm離れた位置を基点とする事を推奨値とするが、必要に応じて任意に変更できるものとする。

地面に印刷されている場合は、地面の直上を基点とする。

天井に印刷されたものは、直下を基点とする。

※GMS用のQRコードは接写ではなく、一定距離、離れた位置から、取得される事を前提としている。特に駅のプラットホームなどでは、線路を挟んだ壁面上に、大きく印刷されている事も考えられるので、基点を変える事ができる方が良い。

※QRコードが高い位置の壁や案内板に印刷されている場合は、QRコードの真下ではなく、より離れた地点からで無いとマークがイメージセンサーの視野内に入らなくなる。また、地面や天井に直接印刷されている場合は、オフセットは0の方が望ましい場合もある。

## G.5 バリア情報

GMSの基点から周囲に対し、バリア情報を提供する。

下記の様な情報を提供する事を目的とする。

表現したい項目:

工事中、清掃中、関係者以外立ち入り禁止、危険箇所、壁、柱、椅子、ゴミ箱、階段、踏切、線路

場所の指定方法:

基点からの角度と距離の組み合わせにより表現する。

高精度領域:

領域サイズの指定: 基点を中心として、情報を提供したい有効半径の距離を指定する。この有効半径を超えたものは、精度が低く曖昧である。

その為、この距離よりも大きな値の距離を座標として与えられている場合は、そこから先が連続しているか、途切れているかは不明とする。

高精度領域を複数指定する事ができる。

これは、中継地点、目的地を、細かく指示する事を場合を想定したものである。

高精度領域の指定は、全てGMSの基点を基に指定する。

指定対象の選定:

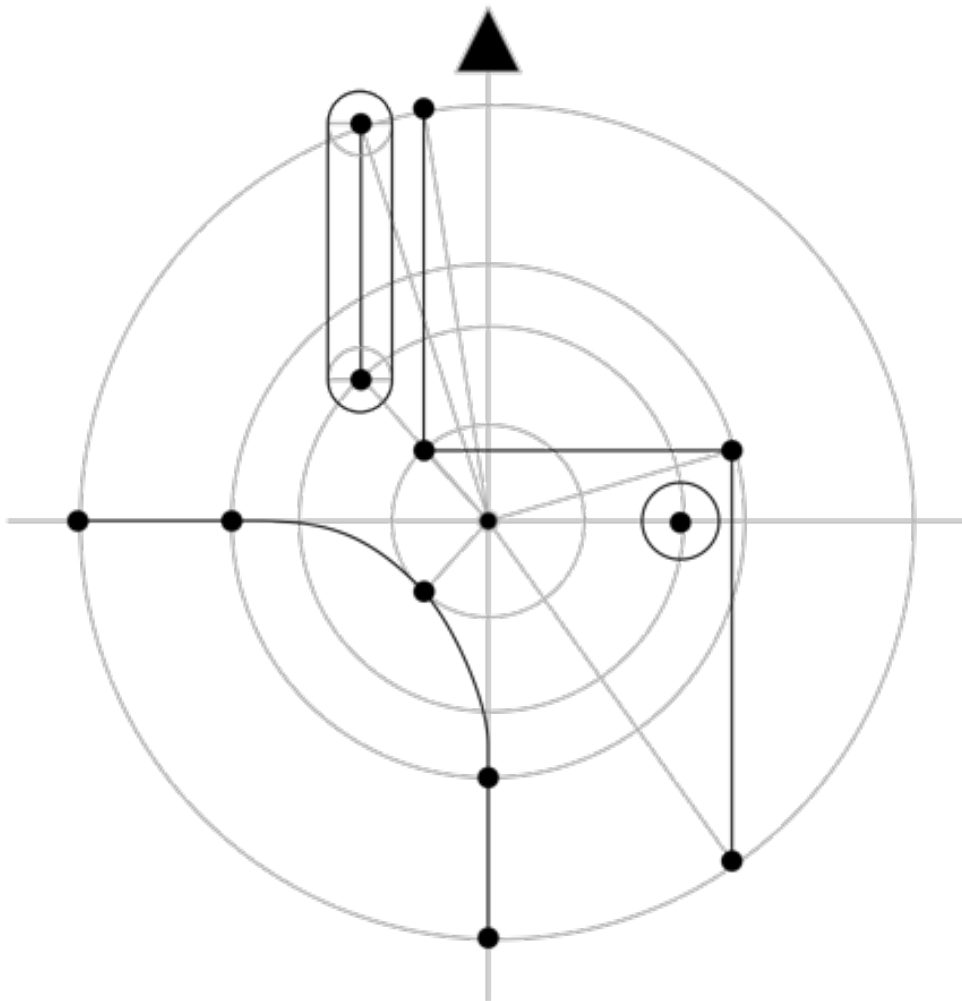
基点の周囲に対し、危険性の高いものを優先して登録するものとする。

記述する順番:

危険性の高いもの、進行方向の正面に近いもの、距離が近いものを先に持つて行くものとする。

※バリアの座標情報を作成する手段として、図面を利用する方法の他に、LiDARを利用する方法が考えられる。

※バリア情報の座標指定の基本的な考えかたは、レーダーの画面である。



#### GMSが持つ制限:

GMSは、銀の弾丸では無い。あくまでQRコードを利用して少ないスペースで、多くの人に役立つ情報を提供する為の仕組みである。少なくとも以下に挙げる制限を持っている。

#### 場所による制限:

GMSは、印刷されたQRコードと、それを読み取る機能を持ったイメージセンサーを組み合わせて使用する事を前提としているため、カメラを利用する事が制限される場所では、運用する事ができない。これは、光学的な条件だけでなく、カメラを利用する事事態が、社会通念やし法律的な理由等で禁止される場面がある事も考慮するべきである。

#### 機材による制限:

GMSを人が利用する場合には、QRコードに保存されているデータをイメージセンサーで読み取り、携帯型の情報処理端末で処理し、映像、音、振動等により情報を伝達されて初めて役立つものとなる。従って、GMSの情報を読み取り、提示する為の装置を持っていない人や、装置を持っていてもGMSの情報が含まれているQRコードに対し、機器が機能する状態に置かれていない場合には役に立たない。

#### 人混みによる制限:

GMSは、GMSの情報が含まれているQRコードをイメージセンサーで、読み取る事ができないと機能しないため、QRコードとイメージセンサーの間に、人が立つ事により、イメージセンサーの視界が塞がれるだけで機能しなくなる。

従って、点字案内板、点字ブロックや、音声案内等も合わせて整備されるべきである。

## US-ASCIIとQRコード英数字符号化の比較:

US-ASCIIで定義されている制御コードは全て削除されている。

英字は大文字のみ、小文字は削除されている。

記号は9種類のみとし、残りは削除されている。

また、当然の事ながら、各文字に対応する番号も異なっている。

数字、アルファベット大文字、記号の順に改変されている。

下記に、英数字モードで直接利用できる文字は太文字、利用できない文字は通常の字体で文字の一覧を制御文字除いて表示する。

**SP**

!?"#

**\$%**

&'()

**\*+**

,

**-. / 0 1 2 3 4 5 6 7 8 9 :**

; < = > ? @

**ABCDEFGHIJKLMNOPQRSTUVWXYZ**

[ \ ] ^ \_ `

abcdefghijklmnopqrstuvwxy

{ } | ~



QRコードの漢字コードの取り扱い:

GMSのエンコード方式に関する規定を見直す必要有り:

GMSの情報が含まれるQRコードの場合:

英数字だけが含まれている場合:

US-ASCIIを利用して符号化する。

英数字に加えてShift\_JISの範囲までの文字が含まれている場合:

Shift\_JISを利用して符号化する。

Shift\_JISの範囲でカバーできない文字が含まれている場合:

QRコードの規格的にはバイナリデータとして扱い、

UTF-8を利用して符号化する。

BOMは付けないものとする。

規格:

QRコード:

ISO (ISO/IEC 18004)または JIS (JIS-X-0510)を参照

## LPS及びGMS向けのエディタを作る場合に考慮すべき項目

本仕様では、QRコードの英数字モードを利用して、できる限り多くの情報を効率的に少ないビット数で表現できる様に、エスケープ文字の概念や、辞書圧縮を実現するために、スタック定数という概念を導入してしている。

これらは、できる限り少ないビット数で表現できれば、同じ面積でもより遠くから、正確に認識可能なQRコードに収める事ができるメリットがある為である。

しかしながら、テキストエディタにより、直接データを作成する場合には、本仕様を理解し、データ構造を常に意識する事が必要となる為、作業者の負担が大きくなる。そこで、本仕様向けのエディタを作る際の指針を提示するものとする。

まず、LPSやGMSの利用場所をシミュレートするモデルと一体するべきである。

### LPSやGMSを適用する場所の3Dデータ

施設の立体形状や図面などのデータを取り扱える必要がある。

DXF形式のインポートとエクスポートへの対応を必須とするべきである。

また、下記の情報を扱える必要がある。

QRコードを貼るまたは、印刷する場所の条件の入力

QRコードのサイズ

誤り訂正のレベル

高さ

取り付け角度

向き等

印刷面の素材

## 光源の情報

### QRコードの利用者の条件

携帯情報端末の光学系及び処理能力の諸元のデータ

各年齢層の平均身長と、携帯情報端末を掲げた場合の高さや角度

車椅子利用者が携帯情報端末を掲げた場合の高さや角度

自動配送ロボット

ドローン

### QRコードが利用される地域の特性:

施設の種類

国、現地の公用語

車椅子とスロープに対する基準

### LPSの場合は下記の情報を入力できる事が求められる

MGRSを含めた、LPSの本仕様で定めた各種プロパティとパラメータのその値

### GMSの場合は下記の情報を入力できる事が求められる

GMSの本仕様で定めた各種プロパティとパラメータとその値

LPS及びGMS向けのエディタは、本仕様で定めたプロパティ及びパラメータのその値を生のテキストで表示できる他に、その演算結果表示できる事が求められる。

例えば、エスケープ文字の処理が適用される前と後の結果の両方で入力と表示を行える事が求められる。

定数の宣言、スタック定数の値の参照は、エディタ側で、ビット数が最小になる様に自動的に適用させる機能が求められる。

また、データから求められた総ビット数と、入力された誤り訂正のレベル、QRコードを表示する面積のサイズとシミュレータの情報から、実際にLPSやGMSを運用した場合に、QRコードの設置箇所に対してどの位置から、どの範囲内の人や、自動配送ロボット、ドローンが認識できるのかを、シミュレータ上で確認できる機能が必要とされる。

また、作成された、QRコードのデータは、テキストファイルで出力する機能、Scalable Vector Graphics形式で出力する機能、QRコードの取り付け位置角度、実際に入れるQRコードのマークも含めて、DXF形式で出力する機能も求められる。

また、既に、LPSやGMSが設置されている場所に於いては、そのQRコードを、写真から読み込む機能も求められる。

LPSの場合は、推奨されるシンボルマークのデータもオプションで出力することができるものとする。

本文の表記上の注意事項:

不等号の小なり(<)と大なり(>)記号でパラメータを囲っているが、これは、あくまで、本文中で、見易くするためだけに入れているのであり、実際のデータとして、不等号の記号を入れるわけではない。

本規格を利用する際の注意事項:

本規格自体は何も保証しない。

規格を利用するものは自己責任により、安全性や特許などを確認する事が求められる。

参考文献:

QRコード

<https://ja.m.wikipedia.org/wiki/QRコード>

QRコードとは

<https://www.qrcode.com/about/index.html>

完全理解！MGRS

<https://www.wingfield.gr.jp/archives/6833>

NTTドコモ 「サービス機能 作ろうiモードコンテンツ」

<https://www.nttdocomo.co.jp/service/developer/make/content/barcode/function/application/common/index.html>

他言語化案内表示ガイドライン

<https://www.tb.mlit.go.jp/hokkaido/bunyabetsu/kankou/tiiki/tagengokahyoji/tagenngoka.pdf>

ネイティブスピーカーの多い言語

<https://ja.m.wikipedia.org/wiki/ネイティブスピーカーの数が多言語の一覧>

ISO 7010

[https://ja.m.wikipedia.org/wiki/ISO\\_7010](https://ja.m.wikipedia.org/wiki/ISO_7010)

DOT Pictograms

[https://en.m.wikipedia.org/wiki/DOT\\_pictograms](https://en.m.wikipedia.org/wiki/DOT_pictograms)

UNICODE のその他絵文字

<http://www.unicode.org/charts/PDF/U1F300.pdf>

UNICODE

<http://www.unicode.org/charts/PDF/U1F680.pdf>

<http://unicode.org/emoji/charts/full-emoji-list.html>

バリアフリー・ユニバーサルデザイン

交通消費者行政/公共交通事故被害者支援

[https://www.mlit.go.jp/sogoseisaku/barrierfree/sosei\\_barrierfree\\_tk\\_000145.html](https://www.mlit.go.jp/sogoseisaku/barrierfree/sosei_barrierfree_tk_000145.html)

公共交通機関の旅客施設に関する 移動等円滑化整備ガイドライン バリアフリー整備  
ガイドライン 旅客施設編

<https://www.mlit.go.jp/sogoseisaku/barrierfree/content/001396978.pdf>

<https://www.mlit.go.jp/barrierfree/public-transport-bf/guideline/guidelinesisetu.pdf>

JIS X 0280

<https://www.asahi-net.or.jp/~ax2s-kmtn/ref/jisx0208.html>

サロゲートペア

<https://www.tohoho-web.com/ex/charset.html>

日本語のローマ字表記の推奨形式

<https://park.itc.u-tokyo.ac.jp/eigo/UT-Komaba-Nihongo-no-romaji-hyoki-v1.pdf>

代用表記

<https://green.adam.ne.jp/roomazi/daiyoohyooki.html>

ローマ字で使う横棒(長音記号マクロン)を加えた英字を表示する方法



[https://www.nishishi.com/blog/2017/05/add\\_macron\\_roma.html](https://www.nishishi.com/blog/2017/05/add_macron_roma.html)

外来語の表記

[https://www.bunka.go.jp/kokugo\\_nihongo/sisaku/joho/joho/kijun/naikaku/gairai/index.html](https://www.bunka.go.jp/kokugo_nihongo/sisaku/joho/joho/kijun/naikaku/gairai/index.html)

きっぷあれこれ

<https://www.jreast.co.jp/kippu/>

家電品を安全に使うマークとその意味

<https://www.aeha.or.jp/safety/mark/page2.html>

図記号データベース

[http://www.ishiimark.com/pict\\_caution00.html](http://www.ishiimark.com/pict_caution00.html)