

# 狭域ブロードキャストによる 新行動予定通知システムの提案

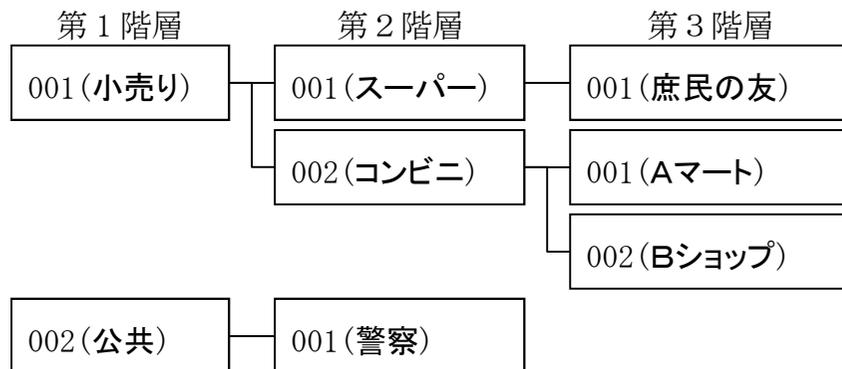
## 属性情報データ編

特願2010-189293 T.O.D.項目報知システム

# 送信データ

属性（店舗）を表すコードには意味を持たない連番コードは使用せず、分類コードを使った階層構造を用います。

階層は必ず大分類から始め、詳細な分類で終わるようにします。  
たとえば次のような体系とした場合、



上図の構成では、「Aマート」は「001-002-001」、「Bショップ」は「001-002-002」、「庶民の友」を表すコードは「001-001-001」となります。

「小売り」と「公共」のように、最大階層数は分類によって異なります。  
送信側は、必ず第1階層から目的の階層までのコードを並べます。

## 階層のメリット

このコード体系には利点がありますが、それを生かすために次の条件を定義します。

【属性コードの比較を行う場合、携帯端末側のコード長で先頭一致を見る】

【携帯端末側のコードが長い場合は不一致とする】

以下に、この条件の判定例を記します。

### 1、スケジュールが「Aマート」で限定されている場合

まず、予定(携帯端末側)に記録する属性コードをAマート「001-002-001」としておきます。

各店舗で下記の判定となり、Aマートだけを特定することが出来ます。

判定コード	001-002-001	一致判定
Aマート	001-002-001	○
Bショップ	001-002-002	×
庶民の友	001-001-001	×
警察	002-001	×

※赤い箇所は一致部分

## 階層のメリット

つぎに、概要編で出てきた「コンビニ」ならどこでも良いというケースです。

### 2、「コンビニ」で限定

予定に記録する属性コードはコンビニ「001-002」としておきます。

各店舗で下記の判定が行われ、コンビニだけを特定することが出来ます。

「携帯端末側のコード長で先頭一致」という条件はここに生きてきます。

判定コード	001-002	一致判定
Aマート	001-002-001	○
Bショップ	001-002-002	○
庶民の友	001-001-001	×

※赤い箇所は一致部分

※灰色箇所は比較されない部分

## 属性コード補足

これまで属性コードの各階層を便宜上'-'で区切って「001-002-001」等と表して来ましたが、実際には文字で表すのではなく各階層を1、2、4バイトなどに固定して数値(配列)として取り扱います。

また、単なる数値の並びはその末尾がどこであるかわからないため、本属性コードではコード毎にコード長を付加するようにします。

属性コードの追加・削除や名称の修正などにともない、携帯端末側の属性データも更新しなければなりません。

※ユーザに直接コードを入れさせるのではなく、リストから順次選択させるようなUIが望ましい

よって、携帯電話のようにインターネットに接続可能であれば所定のサーバから、電子辞書のように通信機能が無い端末では外部メモリなどから更新データが取得できるようにしておく必要があります。

そう言った意味では「サーバ不要」とは言えません。

# 属性情報データ編

-完-

初版 2010/10/01

特願2010-189293 T o D o 項目報知システム