

図 2 INFORM システム構成

送受信する情報のうち、着発時刻・旅客案内情報など、更新頻度が低く、また更新されたことを別の情報から知ることができるものはデータベースとして公開し、SQL で下位システムから問い合わせる方式とした。また、在線列車番情報など更新頻度が高い情報や、運転整理情報など上位系から更新されたことを通知しなければならない情報は、能動的に統合サーバから送信する方式とした。

なお、下位アプリケーション側で必要だが上位の運行管理システムにない一部の情報については、統合サーバで生成している。

各機器の障害対策としては、相互にライフチェックを行い、上位系で障害が発生した際にはその旨を下位システムに通知する。障害復旧時にはダイヤ回復を行なう。

INFORM が提供する情報をまとめると次のようになる。

【DBとして提供する情報】

- ・ダイヤ情報
 - ⇒着発時刻や着発線情報
- ・旅客案内情報
 - ⇒旅客指令が入力した運行概況や振替輸送の情報

【能動的に送信する情報】

- ・列車番号窓情報
 - ⇒制御駅単位の列車在線情報
- ・CTC 情報
 - ⇒軌道回路及び信号現示情報
- ・現発情報
 - ⇒列車の制御駅からの発車情報
- ・運済み情報
 - ⇒列車の終着駅への到着情報
- ・運転整理情報
 - ⇒列車の運転休止等のダイヤ変更情報

3. 宝塚線 PRC からの情報取得

宝塚線情報取得インターフェース装置(以下「宝塚線 I/F 装置」)は JR 宝塚線 PRC の EDP 装置からダイヤ情報を取得し、常に前日/当日/翌日のダイヤを保持している。統合サーバの初回起動時と回復時にそのダイヤ情報を転送し、その後は毎日 23 時過ぎに翌々日のダイヤを取得して統合サーバに転送する。

列車の運転に伴う情報としては、宝塚線 PRC 本体から「列番窓情報」「CTC 情報」「現発情報」「運済み情報」の各情報が宝塚線 I/F 装置経由で統合サーバに送られる。これらは情報の変化の都度送られるイベント情報である。

また、宝塚線卓の輸送指令員が運転整理を行った際は「運転整理情報」としてイベント発生毎に送られると共に、宝塚線 I/F 装置内のダイヤ情報を書き換える。

4. JR 京都神戸線運行管理システムからの情報取得

京都神戸線情報取得インターフェース装置(以下「京都神戸線 I/F 装置」)は京都神戸線システムの指令管理中央装置からダイヤ情報を取得し、常に前日/当日/翌日のダイヤを保持している。統合サーバの初回起動時と回復時にそのダイヤ情報を転送し、その後は毎日 22 時過ぎに翌々日のダイヤを取得し、統合サーバに転送する。

列車の運転に伴う情報としては、「列番窓情報」「CTC 情報」を 4 秒周期、「実績情報」は 30 秒周期で京都神戸線 I/F 装置経由で送られる。

また、京都神戸線卓の輸送指令員が運転整理を行った際は「運転整理情報」としてイベント発生毎に送られると共に、京都神戸線 I/F 装置内のダイヤ情報を書き換える。

5. ダイヤ情報の統合

統合サーバは、両 I/F 装置からダイヤ情報を受け取ると、まずそれぞれのダイヤデータに対し、統合前の翻訳処理を行なう。

実際にはほぼすべての項目で変換作業を行っているが、中でも重要なのはデータベースから情報を検索する際のキー項目となる列車番号と施行日であるので、ここではこの二つの項目の変換について説明する。

当社の場合、列車番号は「回電第回 1000M 列車」のように表記する。(ここでは「回電」の部分「列車種別」、「回」の部分は「冠記号」、「1000」の部分を「数字部」、「M」の部分を末尾記号と呼ぶ) 両システム共に、各記号に独自の番号を割り振っており、その表現も BIN であったり BCD であったりするなどデータ型も異なる。このままでは両データを同一列車と認識できないため、両システムのデータ共に共通の番号、コードに変換している。

施行日については、正式な施行日はその列車の始発駅の日付であるが、両線区とも、最初に当該線区での制御を開

始する日付を施行日としている。そのため、同一の列車であってもどちらかの線区内で日付をまたがる場合、そのままデータを結合すると 1 日違いの別の列車となってしまう。

そのため、先ほどの論理により統合した列車番号をキーに両線区の端駅である尼崎駅およびその宝塚線側隣接駅である塚口駅の着発時刻を比較し、統合することとした。具体的には、ある上り列車の場合、京都神戸線尼崎駅の時刻が 4 月 1 日 1 時 00 分であったとすると、JR 宝塚線の 3 月 31 日施行の列車のダイヤから塚口駅の発時刻を検索し、それが一定時間内であれば、終着駅側の線区のダイヤ施行日を 1 日繰り上げる操作を行なう。これにより、列車の施行日はいずれかの線区で運行を始める日に統一される。ただし、元の線区毎の施行日も、運転整理情報などの処理に必要なため保持している。

これらの読み替えをその他の着発線などの項目についても順次行なうことにより、それぞれのダイヤ情報の統合準備がなされる。そして、読み替え完了後に改めて始発・終着駅などの統合処理を行い、統合ダイヤデータベースとしてセットすることでダイヤ情報の統合が完了する。

6. 送信する情報の統合

<CTC 情報>

京都神戸線からは定周期で、宝塚線からはイベント発生毎に情報が送られてくるが、これを統合サーバで一旦取り込み、駅コードを統一駅コードにセットし直してから、定周期で送信する。

<列番窓情報>

京都神戸線はダイヤとの関連情報や上下運転方向区分を付加する。宝塚線は列車番号の統一フォーマット化と窓番号の統一窓番号コードへの変換と共に、遅延の秒から分への変換を行なう。その後、両線区ともメッセージフォーマットを合わせ、定周期で送信する。

<現発情報>

京都神戸線は列番窓情報のシフトを監視して遅延算出と共に生成する。宝塚線の現発情報は列車番号の統一フォーマット化と、遅延の秒から分への変換と施行日の付加を行なう。その後、両線区ともメッセージフォーマットを合わせて定周期で送信する。

<運済み情報>

京都神戸線は実績情報と終着駅情報の駅を比較し、一致した場合に運済みを生成する。宝塚線は列車番号の統一フォーマット化と施行日の付加を行なう。その後、両線区ともメッセージフォーマットを合わせ、定周期で送信する。

<運転整理>

運転整理情報については特に変換は行なわない。これらの情報は各線区施行日をキーとして、ダイヤ情報の更新に利用する。

7. アプリケーション

作成したシステムのみでは、データが正しく処理されているかの検証が困難であるため、業務指示や旅客案内を想定したアプリケーションをいくつか試作した。

統合運行状況サーバの下にアプリケーションサーバを構築し、送信されてくる各種情報や、それを元に統合ダイヤデータベースを参照した情報を処理し、画面を生成している。

(1) 動力者乗務員向け時刻表

将来、ダイヤ乱れの際に乗務員に対して運転線路や着発線の変更情報を自動的に送信することを想定し、統合ダイヤデータから時刻表情報を生成し表示した。

画面例は高槻発新三田行きの「電第 1241C 列車」である。両線区での駅間運転時分、着発時刻、着発線、運転線路を表示している。(画面 1)

画面 1 動力者乗務員向け時刻表

(2) 社員向け発車標画面

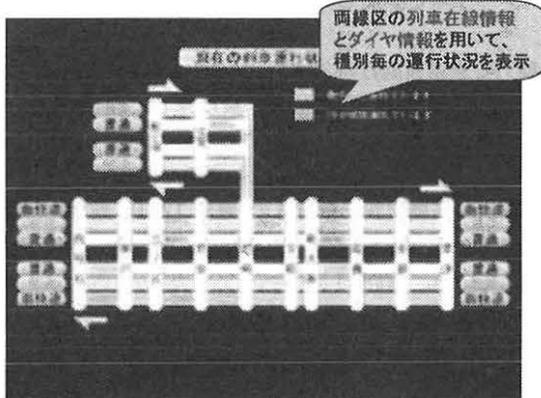
出改札窓口を設置し、案内に利用することを想定した画面である。画面例は尼崎駅で、京都神戸線、宝塚線それぞれの出発順序順に列車を表示し、その現在の在線位置を表示している。(画面 2)

画面 2 社員向け発車標

(3)旅客向け案内表示画面

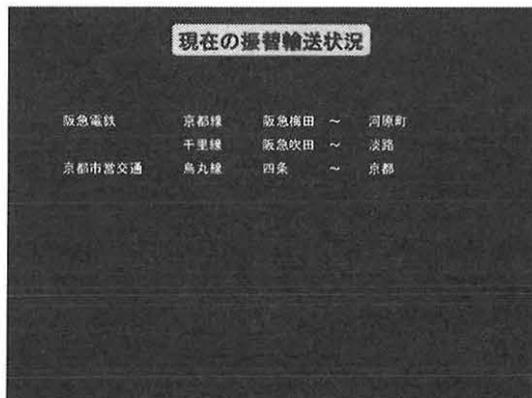
駅コンコースなどでお客様向けに運行概況をお知らせすることを想定した画面である。

両線区の列車番号情報を受けてダイヤを検索し、貨物列車、回送列車など、表示対象以外の列車を除外する。そして予め設定したロジックに従って各区間の列車の遅延の値が閾値を超える場合は色付きのシンボルを表示する。(画面3)



画面3 旅客用案内表示(運行概況)

また、旅客指令が入力した運転見合わせ情報に従い、運転見合わせ区間と振替輸送区間を表示したり、振替輸送情報をテキスト表示する機能がある。(画面4)



画面4 旅客用案内表示(振替輸送)

8. 残された課題

今回、可能な限り情報の統合について検討し実装した。しかしながらまだ解決する必要がある問題点がいくつか存在する。

(1)列車番号に関する課題

例えば、車両故障を起こして途中駅で収容する編成の代わりに、新たな編成を仕立てて同一列車番号、同一時刻で

運転する場合、各運行管理システム内部では同一の列車番号は許されない。そのため各システム内でのみ有効な新たな列車番号を設定するが、両線区でその列車番号の付与の仕方が異なる。これを INFORM で1本の列車として認識することが困難である。

(2)施行日に関する問題

今回2つの線区だけであったためロジックを工夫することで統合施行日を設定できたが、今後、多数の線区の情報統合することを想定した場合、今回の考え方を拡張するだけでは困難が予想される。同様な問題はダイヤ統合論理にもあり、より適切な方法を考える必要がある。

9. 今後の方向性

今回開発した INFORM は、取得/統合した情報を再び運行管理システムへ返すことはしていないが、運転通告前の進路構成抑止や保守作業の管理など、既存の運行管理システムの機能との整合性をどうするかという問題がある。また、前項で述べた課題についても INFORM 側の処理だけでは解決が困難な問題もあり、列車番号や施行日の考え方を整理し、どこでそのようなデータを持ち、処理するか良く考えなければならない。

これは今後の運行管理システムの設計や、当社で輸送計画を司る輸送計画システムのデータ構造と密接に関係する。そのため、今回得られた知見を元に、INFORM の開発と並行して異常時の業務支援や旅客案内のシステム構築の方向性についても再度検討を行なっていく予定である。

10. 終わりに

今回、異なる線区の情報統合して提供することで、新たな価値が発生することを示したが、まだ緒についたばかりである。今後さらに研究を深め、実用化への道筋をつけたいと考えている。

また、本開発にあたり、関係メーカーの皆様には並々ならぬご協力を頂いた。ここに深く謝意を表す。