

# iWay

## **Omni-Patient™ HealthViews User's Guide**

Version 2.5.1

Active Technologies, EDA, EDA/SQL, FIDEL, FOCUS, Information Builders, the Information Builders logo, iWay, iWay Software, Parlay, PC/FOCUS, RStat, Table Talk, Web390, WebFOCUS, WebFOCUS Active Technologies, and WebFOCUS Magnify are registered trademarks, and DataMigrator and Hyperstage are trademarks of Information Builders, Inc.

Adobe, the Adobe logo, Acrobat, Adobe Reader, Flash, Adobe Flash Builder, Flex, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2016, by Information Builders, Inc. and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

# Contents

<b>Preface.....</b>	<b>5</b>
Documentation Conventions.....	5
Related Publications.....	6
Customer Support.....	6
Help Us to Serve You Better.....	7
User Feedback.....	9
iWay Software Training and Professional Services.....	9
<b>1. Configuring Omni-Patient™ HealthViews.....</b>	<b>11</b>
Overview.....	12
Understanding the Architecture of Omni-Patient HealthViews.....	13
Prerequisites and Supported Platforms.....	13
Installing and Configuring the Hash64 Function.....	13
Installing the Hash64 Function.....	14
Creating the Stored Procedure.....	14
Executing the Scripts.....	14
Configuring the Settings in the Property File.....	15
Using the build_all.sh Shell Script.....	15
List of Available Scripts.....	16
Custom Views.....	27
Greenplum Database Considerations.....	28
Data Load Scripts.....	28
Configuring Greenplum Scripts.....	30
<b>Reader Comments.....</b>	<b>33</b>



# Preface

This documentation provides prerequisites and instructions to configure Omni-Patient™ HealthViews.

## How This Manual Is Organized

This manual includes the following chapters:

	Chapter/Appendix	Contents
1	Configuring Omni-Patient™ HealthViews	Provides an overview for Omni-Patient™ HealthViews.

## Documentation Conventions

The following table lists and describes the documentation conventions that are used in this manual.

Convention	Description
THIS TYPEFACE or this typeface	Denotes syntax that you must type exactly as shown.
<i>this typeface</i>	Represents a placeholder (or variable), a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option that you can click or select.
<u>underscore</u>	Indicates a default setting.
Key + Key	Indicates keys that you must press simultaneously.
{ }	Indicates two or three choices. Type one of them, not the braces.

Convention	Description
	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
...	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis (...).
. . .	Indicates that there are (or could be) intervening or additional commands.

## Related Publications

Visit our Technical Documentation Library at <http://documentation.informationbuilders.com>. You can also contact the Publications Order Department at (800) 969-4636.

## Customer Support

Do you have questions about this product?

Join the Focal Point community. Focal Point is our online developer center and more than a message board. It is an interactive network of more than 3,000 developers from almost every profession and industry, collaborating on solutions and sharing every tips and techniques. Access Focal Point at <http://forums.informationbuilders.com/eve/forums>.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our website, <http://www.informationbuilders.com>. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of [www.informationbuilders.com](http://www.informationbuilders.com) also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

Call Information Builders Customer Support Services (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 A.M. and 8:00 P.M. EST to address all your questions. Information Builders consultants can also give you general guidance regarding product capabilities. Be prepared to provide your six-digit site code (xxxx.xx) when you call.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

## Help Us to Serve You Better

To help our consultants answer your questions effectively, be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following table lists the environment information that our consultants require.

<b>Platform</b>	
<b>Operating System</b>	
<b>OS Version</b>	
<b>JVM Vendor</b>	
<b>JVM Version</b>	

The following table lists the deployment information that our consultants require.

<b>Adapter Deployment</b>	
<b>Container</b>	
<b>Version</b>	
<b>Enterprise Information System (EIS) - if any</b>	
<b>EIS Release Level</b>	
<b>EIS Service Pack</b>	
<b>EIS Platform</b>	

The following table lists iWay-related information needed by our consultants.

<b>iWay Adapter</b>	
<b>iWay Release Level</b>	
<b>iWay Patch</b>	

The following table lists additional questions to help us serve you better.

Request/Question	Error/Problem Details or Information
Did the problem arise through a service or event?	
Provide usage scenarios or summarize the application that produces the problem.	
When did the problem start?	
Can you reproduce this problem consistently?	
Describe the problem.	
Describe the steps to reproduce the problem.	
Specify the error messages.	
Any change in the application environment: software configuration, EIS/database configuration, application, and so forth?	
Under what circumstance does the problem <i>not</i> occur?	

The following is a list of error and problem files that might be applicable.

- ❑ Input documents (XML instance, XML schema, non-XML documents)
- ❑ Transformation files
- ❑ Error screen shots
- ❑ Error output files
- ❑ Trace files
- ❑ Service Manager package to reproduce problem
- ❑ Custom functions and agents in use



- ❑ Diagnostic Zip
- ❑ Transaction log

For information on tracing, see the *iWay Service Manager User's Guide*.

## User Feedback

In an effort to produce effective documentation, the Technical Content Management staff welcomes your opinions regarding this document. Please use the Reader Comments form at the end of this document to communicate your feedback to us or to suggest changes that will support improvements to our documentation. You can also contact us through our website, <http://documentation.informationbuilders.com/connections.asp>.

Thank you, in advance, for your comments.

## iWay Software Training and Professional Services

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our website, <http://education.informationbuilders.com>, or call (800) 969-INFO to speak to an Education Representative.

Interested in technical assistance for your implementation? Our Professional Services department provides expert design, systems architecture, implementation, and project management services for all your business integration projects. For information, visit our website, <http://www.informationbuilders.com/support>.



# 1 | Configuring Omni-Patient™ HealthViews

Omni-Patient™ HealthViews is a set of scripts that implements de-normalized tables and views.

Omni-Patient HealthViews allows organizations to leverage existing clinical data, generating valuable actionable insights, which lead to tangible business results. Leveraging Omni-Patient HealthViews to compile and relate content across the entire organizational spectrum, business users can be empowered to communicate, visualize, and analyze data effectively.

This section provides an overview for Omni-Patient HealthViews, key features, and describes the configuration steps that are required.

## Topics:

- ❑ Overview
- ❑ Understanding the Architecture of Omni-Patient HealthViews
- ❑ Prerequisites and Supported Platforms
- ❑ Installing and Configuring the Hash64 Function
- ❑ Creating the Stored Procedure
- ❑ Executing the Scripts
- ❑ Greenplum Database Considerations

## Overview

Omni-Patient HealthViews offers a commercial, off the shelf, clinical and operational data model with dynamic views to empower healthcare analytics and reporting.



Omni-Patient HealthViews provides a centralized data model, which is organized by Domain/Subject. In addition, client expansion and customization are supported.

Key features include:

- ❑ A centralized model to collect disparate healthcare data.
- ❑ Enables self-service healthcare business intelligence.
- ❑ A framework for developing clinical-centric analytic and data mining applications.
- ❑ Supports analysis for: Patient Movement, Demographics, Habits, Outcomes, Volumes, Infections, and more.
- ❑ A clinical health care COTS (commercial off the shelf) data model.

- ❑ The ability to be deployed as a standalone data model (RDBMS, MPP), or integrated with Omni-Patient.
- ❑ Support for multi-tenant data storage, allowing for custom content to be stored and integrated with the Omni-Patient HealthViews data model.
- ❑ Data model relationships clearly defined to expedite content creation (reports, analytics).
- ❑ An optimized data model ensures rapid answers to difficult questions.
- ❑ The capability of Omni-Patient HealthViews to host data that generates Healthcare Performance Analytics dashboards, and balance scorecards.

## Understanding the Architecture of Omni-Patient HealthViews

Omni-Patient HealthViews implements a star schema model that has been de-normalized for ease of use for reporting purposes. It is implemented as dynamic views over tables and provides the ability to view patient facts by date/time and also correlated clinical event data.

Key components include:

- ❑ **Healthcare data model (star schema).** Definition of common healthcare data and relationship for storage and reporting.
- ❑ **Dynamic database views of the data model.** A layer of abstraction of the data model to simplify business intelligence reporting.

## Prerequisites and Supported Platforms

Omni-Patient HealthViews is currently supported on **PostgreSQL** Version 9.3 and higher, **Greenplum**, and must be used only with **Omni-Patient Version 2.5.1**.

## Installing and Configuring the Hash64 Function

### In this section:

Installing the Hash64 Function

Omni-Patient HealthViews relies on a Hash64 function to create unique keys for the tables in the Omni-Patient HealthViews schema. In addition, the Hash64 function is used to create integer values for fields that are character strings in Omni-Patient. Joining integer fields provides improved performance than joining character strings.

## Installing the Hash64 Function

The Hash64 function code is contained in the *ibi\_functions.so* file. A system administrator who has the required authorization, must place the *ibi\_functions.so* file in a known location (the default is */opt/pgsql*) and set permissions to 755 (readable to all, executable by all).

A shell script (*register.sh*) is used to install the Hash64 function. Execute this script by typing the following command:

```
./register.sh [database_name]
```

where:

```
[database_name]
```

Is the name of the database where you want to install the Hash64 function.

**Note:** Only a system administrator who has the required authorization must perform the steps described in this section.

## Creating the Stored Procedure

There are two stored procedures used by Omni-Patient HealthViews:

- ❑ *fn\_hvpartscollector*
- ❑ *fn\_hvtabstatcollector*

SQL scripts for creating both stored procedures are provided in the */stored\_procs* directory. They should be installed in the public schema.

## Executing the Scripts

### In this section:

- Configuring the Settings in the Property File
- Using the *build\_all.sh* Shell Script
- List of Available Scripts
- Custom Views

A shell script is provided to execute all of the Omni-Patient HealthViews scripts in the correct order. The shell script uses one of the following property files for configuration purposes, depending on whether PostgreSQL or Greenplum is the target database:

- ❑ */db\_config/postgres.properties*
- ❑ */db\_config/greenplum.properties*

## Configuring the Settings in the Property File

This section describes the contents and settings of the property file (*postgres.properties* and *greenplum.properties*).

❑ **database.type=**

Defines the database type. Currently, *GPDB* (Greenplum) or *POSTGRES* (PostgreSQL) are supported.

❑ **database.host=**

Defines the address of the database, which can be logical or an IP address.

❑ **database.name=**

Defines the name of the database for Omni-Patient HealthViews.

❑ **database.username=**

Defines the user name to log on to the database.

❑ **database.userpass=**

Password for the user of the database. This is optional and not recommended. Instead, use the *.pgpass* file in PostgreSQL to store passwords.

❑ **database.src\_schema=**

Name of the schema for Omni-Patient (typically *public* for PostgreSQL and *hub* for Greenplum).

❑ **database.targ\_schema=healthviews**

Defines the name of the schema for Omni-Patient HealthViews, which is usually set to *healthviews*.

## Using the `build_all.sh` Shell Script

The shell script to run all Omni-Patient HealthViews scripts is *build\_all.sh*. This shell script must be edited to point to the appropriate property file that is being used (*postgres.properties* and *greenplum.properties*).

Execute this shell script by typing:

```
./build_all.sh
```

In addition, the *build\_all.sh* shell script can be added to a *cron* job to run on a regularly scheduled interval (typically once per day).

## List of Available Scripts

The following table lists and describes all of the available scripts for Omni-Patient HealthViews.

Script	Description
000_hv_availability.sql	Creates <i>public.hv_availability</i> and populates it with the time of the start of the HealthViews build process.
001_date_dimension.sql	Defines all dates in the system.
002_time_dimension.sql	Defines all times in the system.
003_t_codeset_lookup.sql	Lookup table for codesets.
003_t_dim_codemap_lookup.sql	Defines all codemaps for the system.
003_v_dim_codemap_lookup.sql	View for all codemaps in the system.
005_t_patient_master.sql	All mastered patients in the system.
006_t_provider_master.sql	All mastered providers in the system.
007_t_facility_master.sql	All mastered facilities in the system.
008_t_organization_master.sql	All organizations in the system.
020_t_encounter.sql	All encounters in the system.
021_t_clinical_event2.sql	All clinical events in the system.
021_t_movement_event.sql	All movement events in the system.
021_t_observation_event.sql	All observation events in the system.
022_t_specimen.sql	All specimens in the system.
022_v_specimen.sql	View for all specimens in the system.
023_t_appointment.sql	All appointments in the system.
024_t_appointment_notification.sql	All appointment notifications in the system.
025_t_patient_wait_list.sql	All patient wait list records for the system.



<b>Script</b>	<b>Description</b>
026_t_patient_wait_list_preference.sql	All patient wait list preferences records for the system.
027_t_provider_availability_appt.sql	All provider availability appointment records for the system.
028_t_provider_availability_slot.sql	All provider availability slot records for the system.
029_t_provider_template.sql	All provider template records for the system.
030_t_account.sql	All accounts for the system.
030_t_provider_template_schedule.sql	All provider template schedule records for the system.
030_v_account.sql	View for all accounts in the system.
031_t_account_guarantor.sql	All account guarantor records for the system.
031_t_referral_source.sql	All referral source records for the system.
031_v_account_guarantor.sql	View for all account guarantor records.
031_v_referral_source.sql	View for all referral source records.
032_t_account_health_plan_coverage.sql	All account health plan records in the system.
032_v_account_health_plan_coverage.sql	View for all account health plan records.
033_t_account_transaction.sql	All account transactions in the system.
033_v_account_transaction.sql	View for all account transactions.
034_t_bill.sql	All bill records in the system.
034_v_bill.sql	View for all bill records.
035_t_bill_line_item.sql	All bill line item records in the system.
035_v_bill_line_item.sql	View for all bill line item records.

<b>Script</b>	<b>Description</b>
036_t_charge.sql	All charge records in the system.
036_v_charge.sql	View for all charge records.
037_t_claim.sql	All claim records in the system.
037_v_claim.sql	View for all claim records.
038_t_patient_health_plan.sql	All patient health plan records in the system.
038_v_patient_health_plan.sql	View for all health plan records.
040_t_transfusion_order_event.sql	All transfusion order event records in the system.
040_v_transfusion_order_event.sql	View for all transfusion order event records.
050_v_clinical_event.sql	All clinical event records in the system.
051_v_clinical_event_provider.sql	View for all clinical event records.
052_v_encounter.sql	All encounter records in the system.
059_all_dimensional_views.sql	Views for provider master records, facility location master records, organization master records, patient master records, codeset lookup records, time, dates, and date/times.
060_v_appointment.sql	View for all appointment records.
061_v_patient_wait_list.sql	View for all patient wait list records.
062_v_patient_wait_list_preference.sql	View for all patient wait list records.
063_v_provider_availability_slot.sql	View for all provider availability slot records.
064_v_provider_template.sql	View for all provider template records.
066_v_appointment_notification.sql	View for all appointment notification records.

<b>Script</b>	<b>Description</b>
067_v_provider_template_schedule.sql	View for all provider template schedule records.
068_v_provider_availability_appt.sql	View for all provider availability appointment records.
070_t_concept_tables.sql	All concept table records in the system.
071_v_concept_views.sql	View for all concept table records.
080_t_health_plan.sql	All health plan records in the system.
080_v_health_plan.sql	View for all health plan records.
090_t_consent.sql	All consent records in the system.
090_v_consent.sql	View for all consent records.
095_v_hpa_location.sql	View for hpa location records.
095_v_hpa_organization.sql	View for hpa organization records.
100_t_symptom.sql	All symptom records in the system.
100_v_symptom.sql	View for all symptom records.
110_t_allergy.sql	All allergy records in the system.
110_v_allergy.sql	View for all allergy records.
120_t_medication_dosage.sql	All medication dosage records in the system.
120_v_medication_dosage.sql	View for all medication dosage records.
130_t_chronic_condition.sql	All chronic condition records in the system.
130_v_chronic_condition.sql	View for all chronic condition records.
140_t_complication.sql	All complication records in the system.
140_v_complication.sql	View for all complication records.
150_t_substance_usage.sql	All substance usage records in the system.

<b>Script</b>	<b>Description</b>
150_v_substance_usage.sql	View for all substance usage records.
160_t_indication.sql	All indication records in the system.
160_v_indication.sql	View for all indication records.
170_t_procedure_complication.sql	All procedure complication records in the system.
170_v_procedure_complication.sql	View for all procedure complication records.
180_t_dim_date_infoapp.sql	Date records for infoapps.
180_v_dim_date_infoapp.sql	View for date records for infoapps.
190_t_encounter_health_plan_coverage.sql	All encounter health plan coverage records in the system.
190_v_encounter_health_plan_coverage.sql	View for all encounter health plan coverage records.
200_t_order_detail_reason.sql	All order detail records in the system.
200_v_order_detail_reason.sql	View for all order detail records.
210_t_pharmacy_suppl_code.sql	All pharmacy supply code records in the system.
210_v_pharmacy_suppl_code.sql	View for all pharmacy supply code records.
220_t_pharmacy_component_suppl_code.sql	All pharmacy component supply code records in the system.
220_v_pharmacy_component_suppl_code.sql	View for all pharmacy component supply code records.
230_t_pharmacy_administration_route.sql	All pharmacy administration route records in the system.
230_v_pharmacy_administration_route.sql	View for all pharmacy administration route records.
240_t_facility.sql	All facility records in the system.

<b>Script</b>	<b>Description</b>
240_v_facility.sql	View for all facility records.
250_t_dim_provider_contact_method.sql	All provider contact method records in the system.
250_v_dim_provider_contact_method.sql	View for all provider contact method records.
260_t_guarantor.sql	All guarantor records in the system.
260_v_guarantor.sql	View for all guarantor records.
270_t_guarantor_account.sql	All guarantor account records in the system.
270_v_guarantor_account.sql	View for all guarantor account records.
280_t_extended_attribute.sql	All extended attribute records in the system.
285_v_encounter_extended_attributes.sql	View for all extended attribute records.
290_t_payer_master.sql	All payer master records in the system.
290_v_payer_master.sql	View for all payer master records.
295_t_payer_name_master.sql	All payer name master records in the system.
295_v_payer_name_master.sql	View for all payer name master records.
296_t_worker_master.sql	All worker master records in the system.
296_v_worker_master.sql	View for all worker master records.
297_t_worker_job_master.sql	All worker job master records in the system.
297_v_worker_job_master.sql	View for all job master records.
298_t_organization.sql	All organization records in the system.
298_v_dim_organization_unit.sql	View for all organization unit records.

<b>Script</b>	<b>Description</b>
301_t_dim_patient_contact_method.sql	All patient contact method records in the system.
301_v_dim_patient_contact_method.sql	View for all patient contact method records.
302_t_patient_preferred_providers.sql	All patient preferred provider records in the system.
302_v_patient_preferred_providers.sql	View for all patient preferred provider records.
303_t_bed_count.sql	All bed count records in the system.
303_v_bed_count.sql	View for all bed count records.
304_t_closed_bed.sql	All closed bed records in the system.
304_v_closed_bed.sql	View for all closed bed records in the system.
305_t_costing.sql	All costing records in the system.
305_v_costing.sql	View for all costing records.
306_t_member.sql	All member records in the system.
306_v_member.sql	View for all member records.
308_t_survey.sql	All survey records in the system.
308_v_survey.sql	View for all survey records.
309_t_survey_question.sql	All survey question records in the system.
309_v_survey_question.sql	View for all survey question records.
310_t_survey_answer.sql	All survey answer records in the system.
310_v_survey_answer.sql	View for all survey answer records.
311_t_episode.sql	All episode records in the system.
311_v_episode.sql	View for all episode records.

<b>Script</b>	<b>Description</b>
312_t_mammogram_event.sql	All mammogram event records in the system.
312_v_mammogram_event.sql	View for all mammogram event records.
313_t_order_event.sql	All order event records in the system.
313_v_order_event.sql	View for all order event records.
314_t_patient_fall_event.sql	All patient fall event records in the system.
314_v_patient_fall_event.sql	View for all patient fall event records.
315_t_patient_count.sql	All patient count records in the system.
315_v_patient_count.sql	View for all patient count records.
316_t_surgery_case.sql	All surgery case records in the system.
316_v_surgery_case.sql	View for all surgery case records in the system.
318_t_service_order_event.sql	All service order event records in the system.
318_v_service_order_event.sql	View for all service record events.
319_t_event_note.sql	All event note records in the system.
319_v_event_note.sql	View for all event note records.
320_t_gl_transaction.sql	All general ledger transaction records in the system.
320_v_gl_transaction.sql	View for all general ledger transaction records.
321_t_appointment_note.sql	All appointment note records in the system.
321_v_appointment_note.sql	View for all appointment note records.
322_t_extended_attribute_master.sql	All extended attribute master records in the system.

<b>Script</b>	<b>Description</b>
322_v_extended_attribute_master.sql	View for all extended attribute master records.
323_t_provider_assignment.sql	All provider assignment records in the system.
323_v_provider_assignment.sql	View for all provider assignment records.
324_t_surgery_case_cart.sql	All surgery case cart records in the system.
324_v_surgery_case_cart.sql	View for all surgery case cart records.
325_t_surgery_movement.sql	All surgery movement records in the system.
325_v_surgery_movement.sql	View for all surgery movement records.
326_t_surgery_procedure.sql	All surgery procedure records in the system.
326_v_surgery_procedure.sql	View for all surgery procedure records.
327_t_additive_code.sql	All additive code records in the system.
327_v_additive_code.sql	View for all additive code records.
328_t_underlying_cause.sql	All underlying cause records in the system.
328_v_underlying_cause.sql	View for all underlying cause records.
329_t_daily_census.sql	All daily census records in the system.
329_v_daily_census.sql	View for all daily census records.
330_t_daily_emergency.sql	All daily emergency records in the system.
330_v_daily_emergency.sql	View for all daily emergency records.
331_t_payer_claim.sql	All payer claim records in the system.
331_v_payer_claim.sql	View for all payer claim records.
332_t_benefit_plan.sql	All benefit plan records in the system.



<b>Script</b>	<b>Description</b>
332_v_benefit_plan.sql	View for all benefit plan records.
333_t_care_plan.sql	All care plan records in the system.
333_v_care_plan.sql	View for all care plan records.
334_t_payer_client.sql	All payer client records in the system.
334_v_payer_client.sql	View for all payer client records.
335_t_patient_satisfaction_survey.sql	All patient satisfaction survey records in the system.
335_v_patient_satisfaction_survey.sql	View for all patient satisfaction survey records.
336_t_item_cost_component.sql	All item cost component records in the system.
336_v_item_cost_component.sql	View for all item cost component records.
337_t_cart_item.sql	All cart item records in the system.
337_v_cart_item.sql	View for all cart item records.
338_t_clinical_body_site.sql	All clinical body site records in the system.
338_v_clinical_body_site.sql	View for all clinical body site records.
339_t_patient_fall_risk_assessment_event.sql	All patient fall risk assessment event records in the system.
339_v_patient_fall_risk_assessment_event.sql	View for all patient fall risk assessment event records.
340_t_ed_visit.sql	All ed visit records in the system.
340_v_ed_visit.sql	View for all ed visit records.
341_t_hospital_billing_account.sql	All hospital billing account records in the system.
341_v_hospital_billing_account.sql	View for all hospital billing account records.

Script	Description
342_t_costing_other_procedure_code.sql	All costing other procedure code records in the system.
342_v_costing_other_procedure_code.sql	View for all costing other procedure code records.
343_t_reference_range.sql	All reference range records in the system.
343_v_reference_range.sql	View for all reference range records.
347_t_ahrq_qi_tool_input.sql	All ahrq qi tool input records in the system.
347_v_ahrq_qi_tool_input.sql	View for all ahrq qi tool input records.
348_t_ahrq_qi_tool_output.sql	All ahrq qi tool output records in the system.
348_v_ahrq_qi_tool_output.sql	View for all ahrq qi tool output records.
349_t_pharmacy_dispense_event.sql	All pharmacy dispense event records in the system.
349_v_pharmacy_dispense_event.sql	View for all pharmacy dispense event records.
350_t_vaccination_administration_event.sql	All vaccination administration event records in the system.
350_v_vaccination_administration_event.sql	View for all vaccination administration event records.
351_t_medication_administer_event.sql	All medication administer event records in the system.
351_v_medication_administer_event.sql	View for all medication administration event records in the system.
352_t_family_history.sql	All family history records in the system.
352_v_family_history.sql	View for all family history records.
353_t_provider_practice.sql	All provider practice records in the system.
353_v_provider_practice.sql	View for all provider practice records.

<b>Script</b>	<b>Description</b>
354_t_provider_practice_location.sql	All provider practice location records in the system.
354_v_provider_practice_location.sql	View for all provider practice location records.
355_t_provider_speciality_m.sql	All mastered provider specialty records in the system.
355_v_provider_speciality_m.sql	View for all mastered provider specialty records.
356_t_provider_practice_speciality_m.sql	All mastered provider practice specialty records in the system.
356_v_provider_practice_speciality_m.sql	View for all mastered provider practice specialty records.
903_v_hv_release.sql	HealthViews release information.
904_t_hvtabstats.sql	Record counts and deltas for all tables in HealthViews as well as record counts for all partitions of tables with multiple partitions
999_drop_temp_tables.sql	Script to clean up any temporary tables.
999_t_hv_availability.sql	Records stop time for the HealthViews build time and creates the <i>healthviews.t_hv_availability</i> table.
999_v_hv_availability.sql	View for seeing the most recent start and end times for the HealthViews build process as well as the duration.

## Custom Views

In addition to the standard views provided by Omni-Patient HealthViews, custom views can also be created and integrated into the Omni-Patient HealthViews build process. For more information about these customizations, contact your Information Builders Account Manager.

## Greenplum Database Considerations

### In this section:

Data Load Scripts  
Configuring Greenplum Scripts

The *build\_all.sh* script is intended to be used to build HealthViews when the Omni-Patient data resides on the same database. However, many customers choose to use Greenplum as their data warehouse database for HealthViews and PostgreSQL for the more transactional-oriented Omni-Patient. In this type of deployment, additional scripts are required to extract the data from the PostgreSQL database and insert it into Greenplum. A set of utilities is provided for this task.

### Data Load Scripts

#### In this section:

EXTRACT\_OMNI\_DATA.BASH  
LOAD\_OMNI\_STG\_TABLES.BASH  
TRANSFORM\_ALL\_TABLES.BASH  
EXTRACT\_LOAD.BASH  
GP\_DB\_INIT.BASH  
GP\_GEN\_SCHEMA\_ONLY.BASH  
BACKUP\_DB.BASH  
TRUNCATE\_STG.BASH

Separate scripts have been created to extract the data from Omni-Patient, insert the data into a staging schema called *stg* in Greenplum and add a hash value to the data once it is in a Greenplum database. In addition, scripts have been created to create the schemas necessary for the above steps as well as truncate unused data once Omni-Patient has been loaded. All of the data load scripts are described in this section.

#### EXTRACT\_OMNI\_DATA.BASH

This script uses the `psql copy` function to extract data from the Omni-Patient database and write it to flat files in a known location, one file per Omni-Patient table to be replicated.

## **LOAD\_OMNI\_STG\_TABLES.BASH**

This script loads data from the flat files into a staging schema using the GPLOAD utility of Greenplum.

## **TRANSFORM\_ALL\_TABLES.BASH**

This script transforms the data in the staging schema by adding a hashed value to the ID field and distributing the data by that field. It calls a sub-script, TRANSFORM\_DATA\_TO\_HUB.BASH to accomplish this.

## **EXTRACT\_LOAD.BASH**

This is a master script to coordinate the execution of all of the above scripts in the proper order. It is typically configured to run as a cron job. It can also be modified to run other scripts as required.

## **GP\_DB\_INIT.BASH**

This script reads the schema from Omni-Patient and then creates the staging and hub schemas in Greenplum. If those schemas already exist it will drop them and recreate them, all data in those schemas will be lost. It is typically not run on a regular basis.

## **GP\_GEN\_SCHEMA\_ONLY.BASH**

This is similar to the GP\_DB\_INIT.BASH script except it does not actually create the schema. Instead, it creates a DDL file to for both the *stg* and *hub* schemas. This can be useful to run if new tables are added to Omni-Patient.

## **BACKUP\_DB.BASH**

This script creates a backup copy of the current HealthViews database by creating a new database using the current production database as a template. It is typically run before loading any new data into the staging or hub schemas.

## **TRUNCATE\_STG.BASH**

This script can be used to truncate the *stg* schema after HealthViews has been populated in order to save space and reduce backup time.

## Configuring Greenplum Scripts

### In this section:

Greenplum YAML Files

Configuring the various scripts for populating Omni-Patient HealthViews in Greenplum is performed through the *replicator.properties* file. This properties file contains the following settings:

- ❑ **postgres.hostname=**  
Name of the host running the PostgreSQL Omni-Patient database.
- ❑ **postgres.port\_number=**  
Port for that host.
- ❑ **postgres.database\_name=**  
Name of the database for Omni-Patient.
- ❑ **postgres.schema\_name=**  
Name of the schema for Omni-Patient (normally *Public*).
- ❑ **postgres.username=**  
PostgreSQL database admin user name.
- ❑ **postgres.password=**  
PostgreSQL database admin password (optional if configured in the *.pgpass* file of the user).
- ❑ **greenplum.hostname=**  
Name of the host running the HealthViews database.
- ❑ **greenplum.port\_number=**  
Port for that host.
- ❑ **greenplum.database\_name=**  
Name of the database to be used (normally *omnipatient\_bi*).
- ❑ **greenplum.staging.schema\_name=**  
Name of the staging schema (normally *stg*).
- ❑ **greenplum.staging.ddl\_file=**

Location of the ddl file for the staging schema.

- ❑ **greenplum.hub.schema\_name=**  
Name of the hub schema (normally *hub*).
- ❑ **greenplum.hub.ddl\_file=**  
Location of the ddl file for the staging schema.
- ❑ **greenplum.username=**  
Greenplum database admin name.
- ❑ **greenplum.yaml\_dir=**  
Location of the YAML files needed for Gpload.
- ❑ **greenplum.yaml\_template=**  
Template to be used by Gpload.
- ❑ **replicator.lzdir=**  
Root directory for the Omni-Patient to HealthViews replication processes.

## Greenplum YAML Files

YAML files are used by the Greenplum GPLOAD utility to define the data format for the flat files that are being loaded into Greenplum. The YAML file contains a description of the database server to be accessed and information on the formatting of the files to be loaded. In addition, the YAML file must be customized to the specific environment. The YAML file is structured as follows:

```
VERSION: YAML version to be used. Should be left as 1.0.0.1
DATABASE: Database to be used. Should be L_GP_DATABASE which allows
replicator.properties to specify the database.
USER: Name of the user to load the files.
HOST: Name of the host with the database
PORT: Port number on the host.
GPLOAD:
  INPUT: - SOURCE: LOCAL_HOSTNAME:
    - Name of the host with the flat files to be loaded.
      FILE:
        - Name of the file to be loaded. Should be L_LOADPIPE to match
the value passed by the loading script.
      PORT: Port number to read the file in on. Should be
8081
    - FORMAT: format of the file. Should be csv.
    - DELIMITER: Delimiter to mark fields. Should match the delimiter used
in the extract_omni_data.bash script.
    - ESCAPE: Escape character to use. Should match the escape value in
the extract_omni_data.bash script.
    - NULL_AS: Leave as "
    - QUOTE: Quote character. Should be ""
    - HEADER: Whether or not a header is used in the flat file. Should be
set to FALSE.
    - ERROR_LIMIT: Number of errors to ignore before rejecting the flat
file. Typically set to 1000.
    - ERROR_TABLE: Table to write errors to.
OUTPUT: - TABLE: The table to write the files to. Should be set to
L_SCHEMA.L_TABLE.
  - MODE: The mode to use. Should be set to insert.
```



## Reader Comments

In an ongoing effort to produce effective documentation, the Technical Content Management staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please share your suggestions for improving this publication and alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

**Mail:** Technical Content Management  
Information Builders, Inc.  
Two Penn Plaza  
New York, NY 10121-2898

**Fax:** (212) 967-0460

**Email:** [books\\_info@ibi.com](mailto:books_info@ibi.com)

**Website:** <http://documentation.informationbuilders.com/connections.asp>

Name: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_

Telephone: \_\_\_\_\_ Date: \_\_\_\_\_

Email: \_\_\_\_\_

Comments:

# **Reader Comments**