

DBI-Link: 3.0



What is DBI-Link?

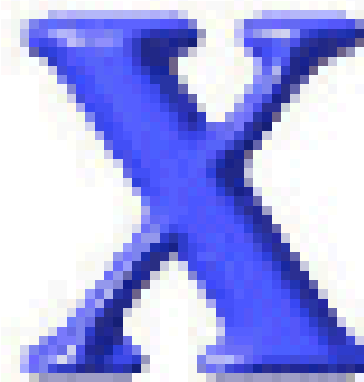
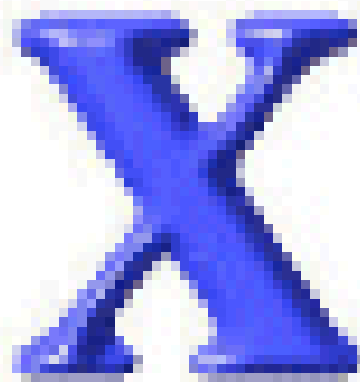
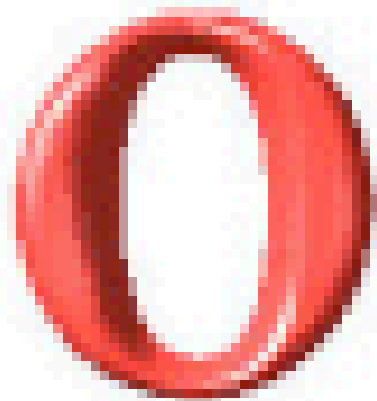
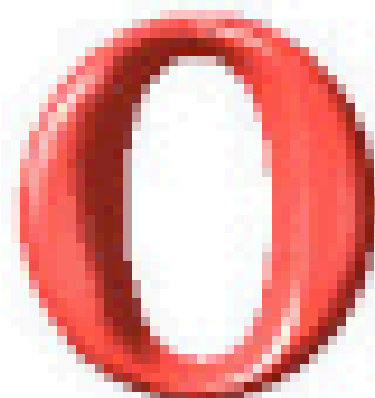
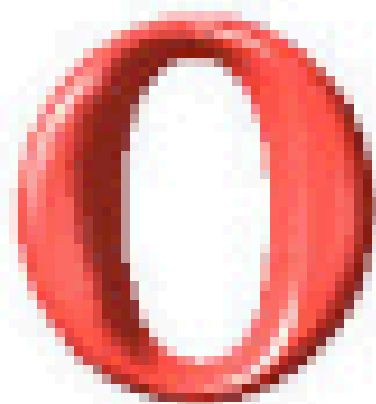
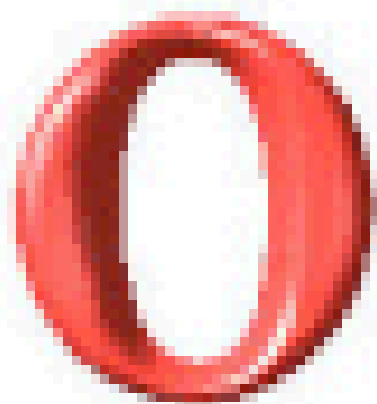
Gives you SQL access to external tabular data from PostgreSQL

— [PostgreSQL

— [Perl and DBI

— [License: BSD





PID	COMMAND	%CPU	TIME	#TH	#PRTS	#MREGS	RPRVT	RSHRD	RSIZE	VSIZE
2021	top	10.6%	0:03.50	1	16	25	312K	400K	2.18M	27.1M
2004	bash	0.0%	0:00.02	1	12	20	160K	856K	656K	18.2M
1996	Preview	0.0%	0:00.68	2	60	124	1.55M	6.31M	5.86M	103M
1992	Keynote	0.0%	1:17.72	3	104	349	23.3M	18.5M	34.0M	161M
1971	lookupd	0.0%	0:00.19	2	37	65	320K	720K	1.04M	28.5M
1958	firefox-bi	0.0%	2:15.09	11	294	471	29.3M	32.6M	49.9M	342M
1889	Terminal	2.6%	0:27.93	3	68	179	2.64M	11.7M-	19.5M-	111M-
1868	iChatAgent	0.0%	0:00.35	3	62	49	0K	752K	4.85M	69.2M
1865	iSightAudi	0.0%	0:00.01	1	22	20	0K	288K	996K	26.8M
483	AppleSpell	0.0%	0:00.28	1	25	37	184K	908K	3.60M	36.3M
454	Microsoft	0.0%	0:07.02	1	70	116	728K	2.29M	5.66M	94.2M
452	iTunesHelp	0.0%	0:00.13	1	49	74	200K	1.79M	3.93M	85.4M
451	iCalAlarms	0.0%	0:00.33	1	60	83	472K	1.34M	6.55M	89.0M
447	Finder	0.0%	0:19.17	1	108	172	3.29M	12.7M	19.0M	120M
445	SystemUISe	0.0%	0:08.28	2	200	240	1.96M	6.46M	13.1M	106M
443	Dock	0.0%	0:02.98	2	103	142	948K	7.34M	11.1M	103M
439	pbs	0.0%	0:00.34	2	32	42	316K	976K	3.46M	44.0M
435	httpd	0.0%	0:00.01	1	9	83	0K	312K	268K	28.3M
430	master	0.0%	0:00.06	1	13	22	52K	312K	268K	26.8M
379	ntpd	0.0%	0:01.39	1	10	19	96K	388K	592K	17.9M
373	ucontrolld	0.0%	0:00.02	1	22	25	0K	492K	2.10M	27.3M
361	xinetd	0.0%	0:00.01	1	12	20	0K	292K	424K	26.8M
349	DirectoryS	0.0%	0:00.69	2	65	107	256K	1.27M	3.24M	31.5M
348	httpd	0.0%	0:00.82	2	38	84	16K	312K	1.16M	28.3M
335	automount	0.0%	0:00.04	2	29	27	8K	612K	568K	28.3M
308	automount	0.0%	0:01.77	3	71	40	228K	680K	3.33M	29.1M
307	cupsd	0.0%	0:01.52	1	11	28	284K	760K	1.73M	28.3M
280	rpc.lockd	0.0%	0:00.00	1	9	17	0K	288K	284K	17.7M
265	nfsiod	0.0%	0:00.01	5	29	24	0K	288K	272K	19.6M

BANGALORE CITY TRAFFIC POLICE

ROAD ACCIDENT CASES PARTICULARS

YEAR	FATAL	KILLED	NON FATAL	INJURED	TOTAL
2003	843	883	9,662	7,980	10,505
2004	875	903	8,226	6,921	9101
2005 ^{UPTO} AUG	544	570	4,507	4,026	5,051

COURTESY & MAINTAINED BY :

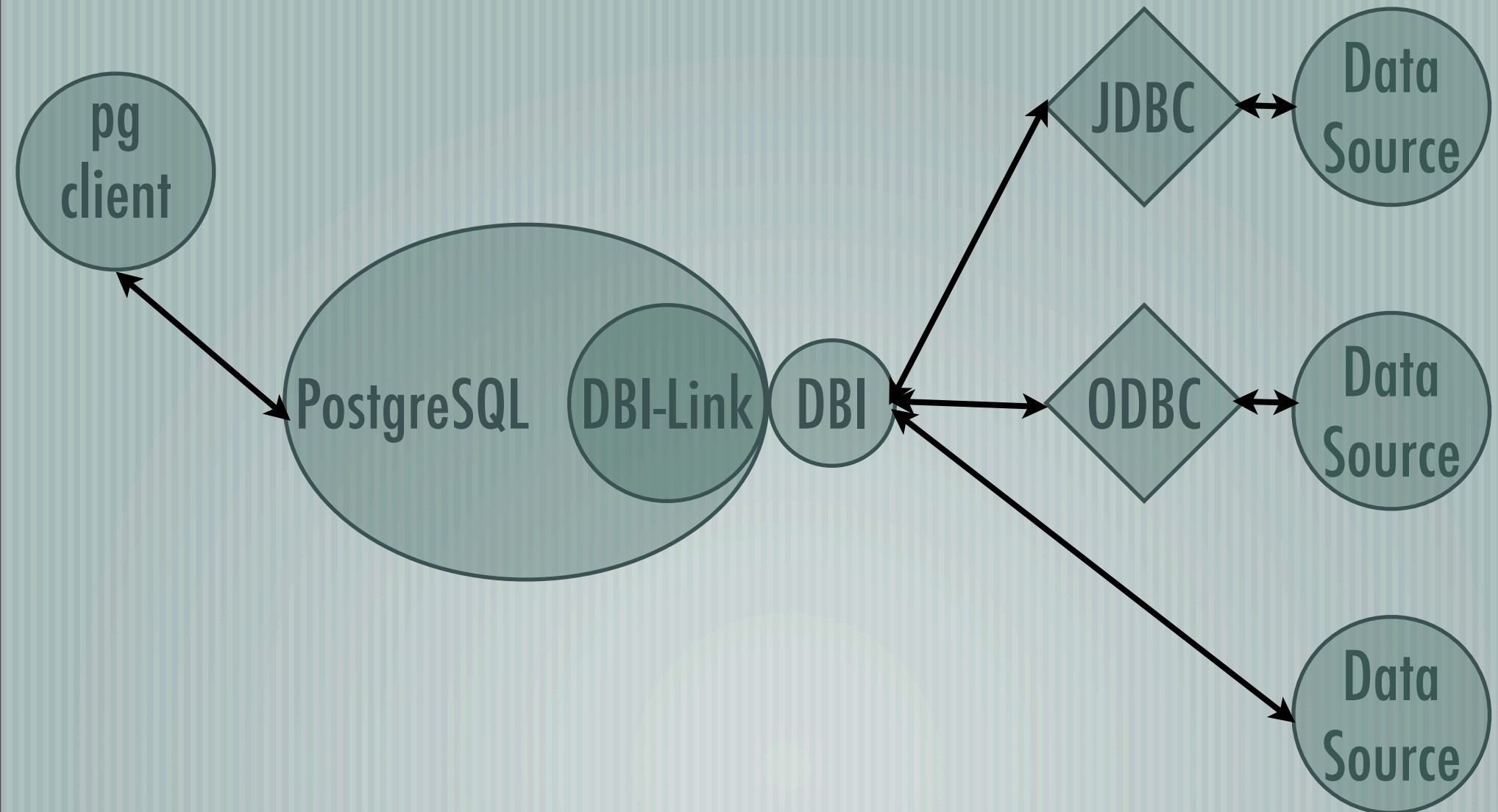
Vijayalakshmi Silks & Sarees

M.G. ROAD BANGALORE, PH:25587937
25587395

Computer-based Tabular Data DBI Can Use

PostgreSQL	Oracle
DB2	Excel
MySQL	CSV
NMAP	Google
ps	etc.

The Big Picture



Current Uses

- [MySQL

- [Informix

- [MS-SQL Server (spawned project)

- [Oracle Conversion: multi-TB

Software You Need

— [DBI-Link <http://www.pgfoundry.org/projects/dbi-link/>

— [PostgreSQL 8.3 or better

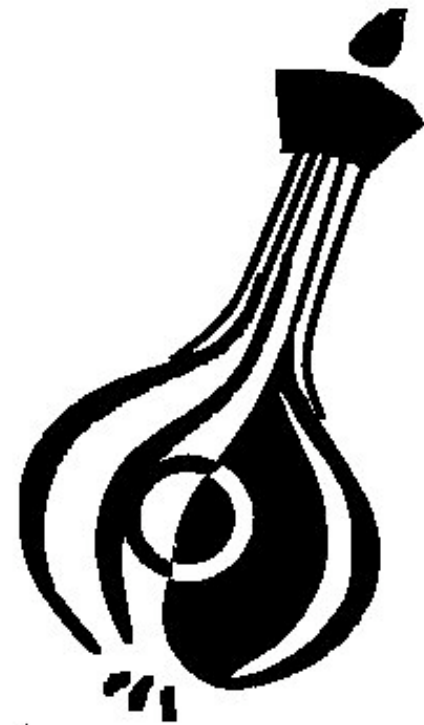
— PL/PerlU

— JSON::XS



Software You Need

- Perl 5.8.8
 - JSON::XS.pm from CPAN
 - DBI.pm
 - DBD module for each type of external data source



Setting Up DBI-Link

- [Load PL/PerlU into PostgreSQL
- [Run the DBI-Link creation SQL script in PostgreSQL
- [For each remote data source:
 - Make the accessor functions

Installing DBI-Link On Your Machine

```
yum install postgresql-dbi-link
```


Installing DBI-Link Into a Database

```
psql -f dbi_link.sql outreach
```

Setting Up Access to an External Data Source

```
SELECT make_accessor_functions(  
    'dbi:Sybase:dbname=help;host=db.mssql.com;port=4100',  
    'newbie',                               /* User */  
    'help_me',                             /* Password */  
    '{  
        "AutoCommit" : 1,  
        "RaiseError" : 0,  
    }',                                     /* Attributes (JSON) */  
    'needs_help_schema',                   /* External Schema */  
    NULL,                                 /* External Catalog */  
    'needs_help',                         /* Local Schema */  
);
```

JOIN Internal and External Data

```
SELECT h.column_here, t.column_there
FROM
    table_here h
LEFT JOIN
    needs_help.table_there t
ON
    (h.here_id = t.there_id);
```

Import External Data

```
INSERT INTO schedule (  
    venue,  
    room,  
    start_time  
)  
SELECT venue, room, start_time  
FROM a_spreadsheet.schedule;
```

Reach Out and Fix Something

```
UPDATE needs_help.log_table  
SET my_timestamp = NULL  
WHERE my_timestamp = '0000-00-00 00:00:00';
```

Using PostgreSQL Features Externally

```
DELETE FROM needs_help.schedule
WHERE
    start_time < CURRENT_TIMESTAMP
    - INTERVAL '1 day';
```


Some Terms

— [**RULE**

— [**VIEW**

— [**Set-Returning Function (SRF)**

RULEs (1/3)

- [RULEs rewrite SQL statements into other SQL statements

VIEWS (2/3)

- [Like a TABLE

- [A RULE called `_RETURN`

SRFs (3/3)

- [Can take parameters
- [Return rows
- [These SRFs return SETOF RECORD

The DBI-Link Implementation

- [Stores connection information in the DBI-Link schema
- [CREATES VIEWS for each external table
 - CREATES RULEs & TRIGGER functions
 - SELECT, INSERT, UPDATE and DELETE of the VIEWS
- [Functions use cached database handles to:
 - read from external data sources
 - write to external data sources

What Goes On Underneath

- [call `make_accessor_functions()`
 - INSERTs database connection info into `dbi_link.dbi_connection`
 - CREATEs writeable VIEWS

Set-Returning Functions

- [A Set-Returning Function (SRF) takes parameters and returns rows as though they were from a table.
- [These SRFs return a set of composite TYPEs.
- [Each TABLE and VIEW in the external data source now has a corresponding SRF

Tools DBI-Link Uses (1/2)

```
remote_select(  
  data_source_id INTEGER, /* Created by make_accessor_functions() */  
  remote_query TEXT      /* Any query that returns rows */  
)
```

```
SELECT country_name  
FROM remote_select(  
  1,  
  'SELECT country_name FROM country'  
)  
AS (country_name TEXT);
```

Tools DBI-Link Uses (2/2)

```
remote_execute(  
  data_source_id INTEGER, /* Created by make_accessor_functions() */  
  remote_query TEXT      /* Any query that does NOT return rows */  
)
```

```
SELECT remote_execute(  
  1,  
  $$DELETE FROM country WHERE country_name = 'Empire of Grand Colombia'$$  
)
```

CREATE VIEW for Each External TABLE

- [Each TABLE in the external data source has a set of columns
- [DBI-Link uses DBI methods to guess what data types those columns are.
- [It then CREATEs a VIEW with each of the names and the corresponding type.

VIEWS use the SRFs

—— [SELECT on a VIEW ->

- Rule transforms it into a SELECT on an external TABLE
- Sends it out via DBI
- Returns the rows (if any)

INSERT, UPDATE and DELETE

— [INSERT, UPDATE or DELETE on a VIEW ->

- Rule transforms it into an INSERT on a Shadow TABLE
- Trigger on shadow TABLE operates on each inserted row
- Sends the row transformation out via DBI.

Shadow Tables

- Shadow TABLEs have $2n+1$ columns in them, where n is the number of columns in the external TABLE or VIEW
- 1 for the action to be taken (INSERT, UPDATE or DELETE)
- n for the OLD values of a row (possibly all NULL)
- n for the NEW values (possibly all NULL)

DBI-Link 3.0 Features

- ["Neutral" Data Type Mapping by default

- [Data Types Per DBD

- [User-settable Type Mapping

The Future

- [More Data Types Per DBD

- [Internationalization

- [Automatic Predicate Pushing (PostgreSQL 8.4)

- [Your Features

Obrigado!

Copyright © David Fetter 2008

All Rights Reserved

<http://fetter.org/>