SWDB

Unified Software database for DNF

https://github.com/edynox/libhif
https://github.com/edynox/dnf

Eduard Čuba 2017

HISTORY BACKEND

What is wrong?

yumdb – flat directory database
- inefficient
- slow
- no binding capability

history/history*.sqlite
- outdated design
- duplicity with yumdb

groups.json
- no binding capability

ACTUAL DESIGN

groups.json:
>   GROUPS:
>>   - GroupID
>>>   - full_list
>>>   - pkg_exclude
>>>   - name
>>>   - pkg_types
>>>   - ui_name
>>>   - grp_types

>   Meta:
>>   - version
>   ENVIRONMENTS:

>>   ...

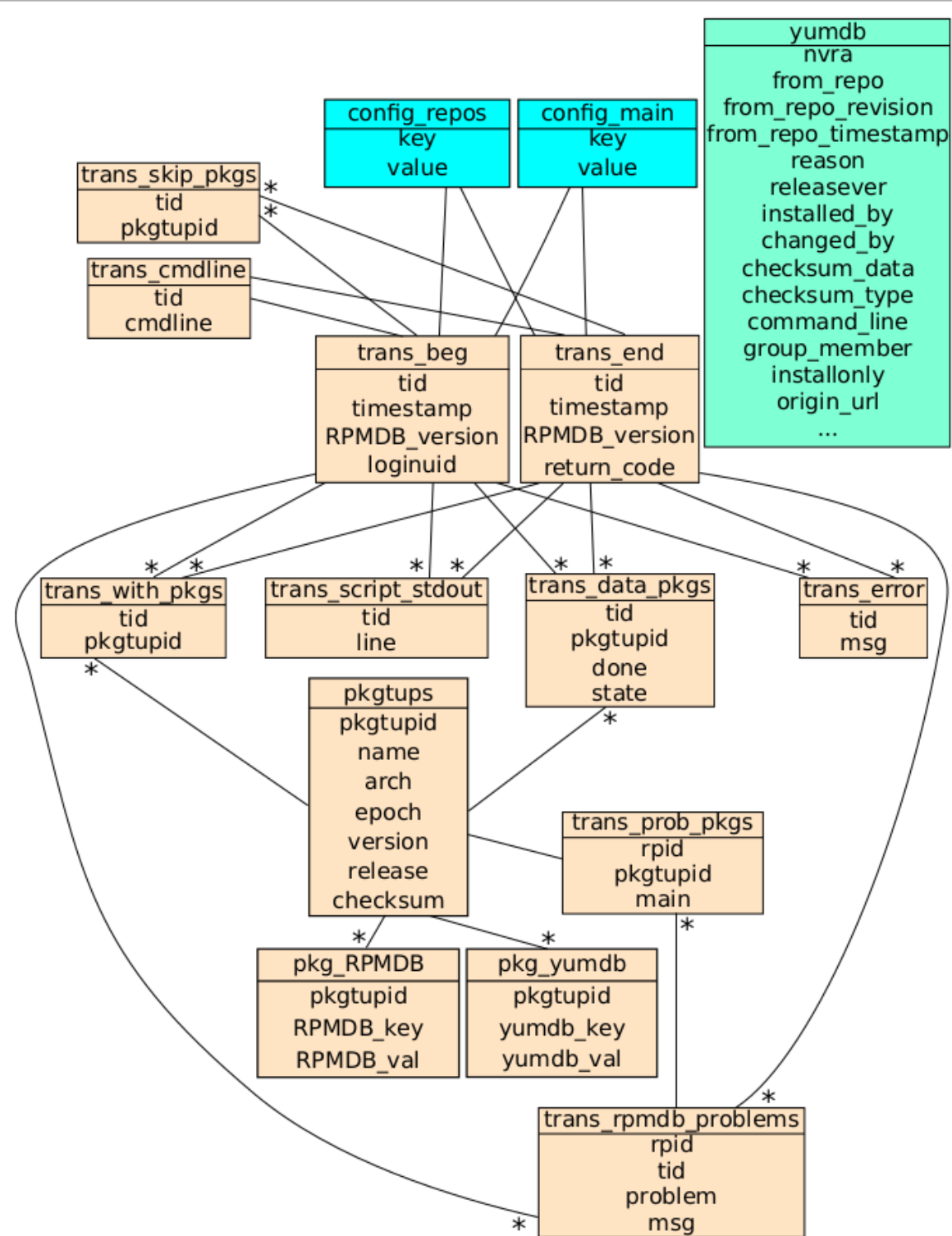**Source:** UNIFIED SOFTWARE DATABASE FOR RPM BASED SYSTEMS – Jan Šihlan 2014

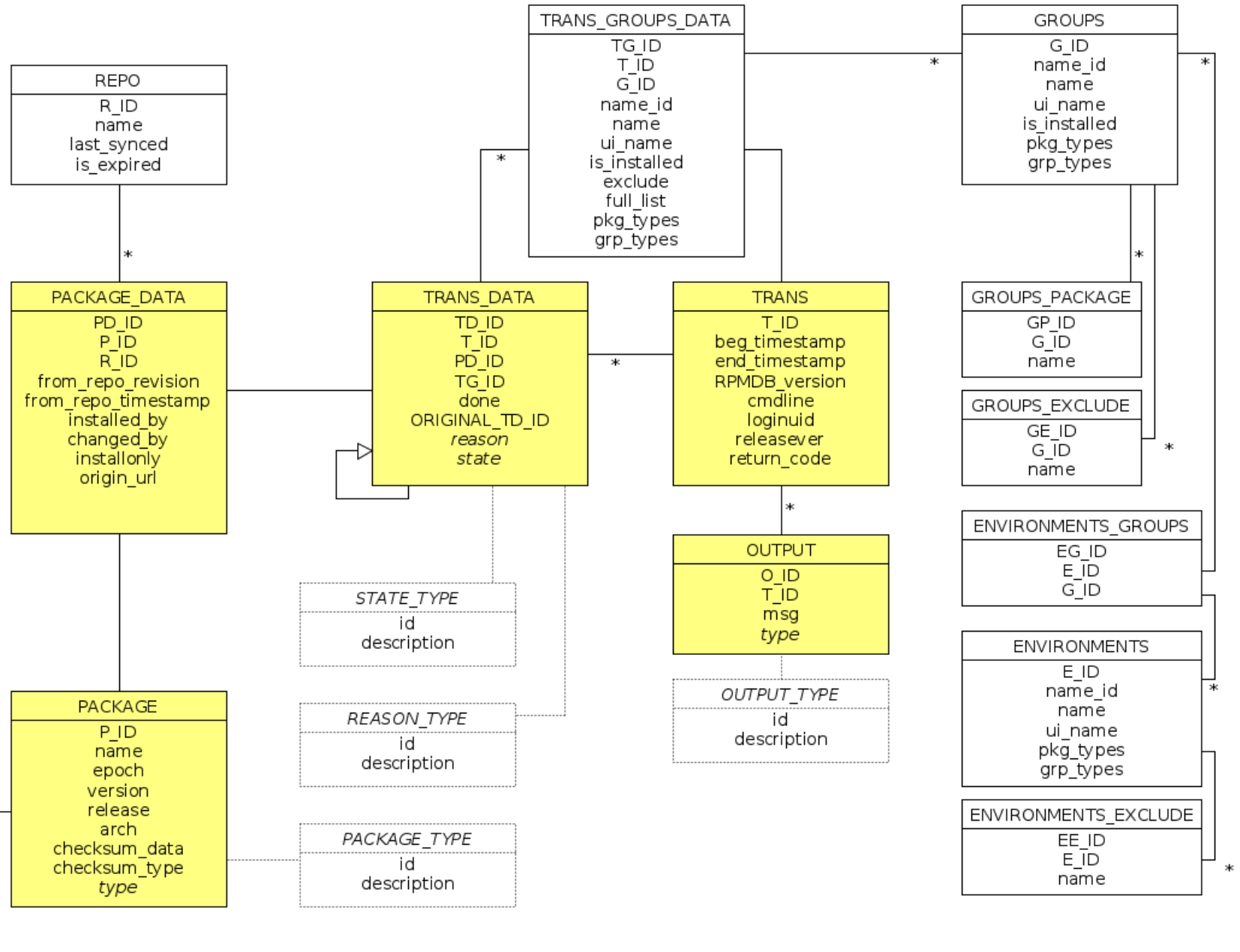Figure 3.3: database scheme in Yum and DNF nowadays

DEVELOPMENT

What to do?

- New database design
- Transformation tool
- database API
- DNF refactor

SWDB

NEW DATABASE DESIGN

SWDB

IDs:
PD_ID = package data ID
P_ID = package ID
TD_ID = transaction data ID
R_ID = repo ID
G_ID = group ID
T_ID = transaction ID
GP_ID = group package ID
GE_ID = group package ID
GP_ID = group package ID
EG_ID = enviroment group ID
E_ID = enviroment ID
EE_ID = env. exclude ID

REPO
R_ID
name
last_synced
is_expired

PACKAGE_DATA
PD_ID
P_ID
R_ID
from_repo_revision
from_repo_timestamp
installed_by
changed_by
installonly
origin_url

TRANS_GROUPS_DATA
TG_ID
T_ID
G_ID
name_id
name
ui_name
is_installed
exclude
full_list
pkg_types
grp_types

GROUPS
G_ID
name_id
name
ui_name
is_installed
pkg_types
grp_types

TRANS_DATA
TD_ID
T_ID
PD_ID
TG_ID
done
ORIGINAL_TD_ID
reason
state

TRANS
T_ID
beg_timestamp
end_timestamp
RPMDB_version
cmdline
loginuid
releasever
return_code

GROUPS_PACKAGE
GP_ID
G_ID
name

GROUPS_EXCLUDE
GE_ID
G_ID
name

OUTPUT
O_ID
T_ID
msg
type

ENVIRONMENTS_GROUPS
EG_ID
E_ID
G_ID

STATE_TYPE
id
description

REASON_TYPE
id
description

OUTPUT_TYPE
id
description

ENVIRONMENTS
E_ID
name_id
name
ui_name
pkg_types
grp_types

RPM_DATA
RPM_ID
P_ID
buildtime
buildhost
license
packager
size
sourcerpm
url
vendor
committer
committime

PACKAGE
P_ID
name
epoch
version
release
arch
checksum_data
checksum_type
type

PACKAGE_TYPE
id
description

ENVIRONMENTS_EXCLUDE
EE_ID
E_ID
name

SWDB TRANSFORMER

dnf/yum/swdb_transformer.py

Automatically triggered

```
[edynox@ares dnf]$ sudo ./dnf.sh history
Database transformation running...
Transforming packages
Transforming transactions
Transforming yumdb
Transforming groups
Database integrity 100%
Last metadata expiration check: 2:52:41 ago on Sun Sep 25 12:43:54 2016 CEST.
ID    | Command line        | Date and time   | Action(s)    | Altered
```

Transforms:
- yumdb
- history*.sqlite
- groups.json

Into:
➜ swdb.sqlite

SWDB

DNF-SWDB

LIBDNF MODULE WRITTEN IN C

Providing required functionality for SWDB
- rewritten DNF modules for:
    - groups
    - history
    - yumdb

Passing data to DNF
- GObject Introspection

libdnf/dnf-swdb*

SWDB

SWDB INTERFACE
DNF EXTENSION FOR SWDB

Interface for using hif-swdb from DNF
- Wraps up swdb functions into Python Class
- Replaces some classes from yum/history.py
- Better maintainability

```python
419    class SwdbInterface(object):
420        def __init__(self, db_path, root='/', releasever=""):
421            self.swdb = Hif.Swdb.new(db_path, releasever)
422            self.releasever = releasever
423            self.addon_data = _addondata(db_path, root)
424            if not self.swdb.exist():
425                self.swdb.create_db()
426                swdb_transformer.run(output_file=self.swdb.get_path())
```

SWDB

TODO
WHAT ELSE?

- Finish unit and CI tests
- Fix some bugs (history info command)
- Unify package objects (future)
- Integration with PackageKit? (future)
- Talk about your questions and ideas :)

SWDB

SOMETHING USEFUL

CODE:

https://github.com/edynox/libhif
https://github.com/edynox/dnf

ME:

cuba.edo@gmail.com
xcubae00@stud.fit.vutbr.cz