starzel.de

GROWING PAINS

PosKeyErrors and other malaises

Philip Bauer - starzel.de - Plone Conference 2020

SYMPTOM 1: HUGE DATABASE

SYMPTOM 1: HUGE DATABASE

- ***Cause 1: Revisions / Versions***
- ***Remedies***
  - Remove all versions and pack

```
hs = api.portal.get_tool('portal_historiesstorage')
zvcr = hs.zvc_repo
zvcr._histories.clear()
storage = hs._shadowStorage
storage._storage.clear()
```

  - Manage/limit revisions. Use collective.revisionmanager
  - Disable versioning of files
  - Enable manual versioning instead of automatic

SYMPTOM 1: HUGE DATABASE

- ***Cause 2: No packing***
- ***Remedies***
  - Pack!
  - Use the script `zeopack` which is part of `plone.recipe.zeoserver`
  - Add a cronjob for it

SYMPTOM 1: HUGE DATABASE

- ***Cause 3: Unused Content***
- ***Remedies***
  - Delete it
  - Find it first.
  - You could use `statistics.py` from `collective.migrationhelpers`

SYMPTOM 1: HUGE DATABASE

- ***Cause 4: SearchableText is huge***
- ***Remedies***
  - Use solr or elasticsearch and remove `SearchableText` index
  - Don't index files

SYMPTOM 1: HUGE DATABASE

- ***Cause 5: Large blobs***
- ***Remedies***
  - Limit upload size
  - Get stats and remove/replace too large items

SYMPTOM 1: HUGE DATABASE

- ***Cause 6: Aborted uploads***
- ***Remedies***
  - Check `IAnnotations(portal).get('file_upload_map')`

SYMPTOM 2: SLOW SITE

SYMPTOM 2: SLOW SITE

- ***Cause 1: Unneeded full renders of content***
- ***Remedies***
  - Use Python in page templates
    - ~~`tal:define="foo context/foo"`~~
    - `tal:define="foo python:context.foo"`

SYMPTOM 2: SLOW SITE

- ***Cause 2: Wake up many objects***
- ***Remedies***
  - Always use brains and metadata
  - Listing 3000 brains: 0.2 seconds
  - Listing 3000 objects: 2 seconds
  - Same for Volto when you use search-endpoint with `fullobjects`

SYMPTOM 2: SLOW SITE

- ***Cause 3: No caching***
- ***Remedies***
  - Switch on built-in caching
  - Add varnish
  - Manage zeocache
  - Use `memoize` in your code

SYMPTOM 2: SLOW SITE

- ***Cause 4: Hardware***
- ***Remedies***
  - Don't be cheap
  - Buy enough ram to keep the DB in memory

SYMPTOM 2: SLOW SITE

- ***Cause 5: Slow code***
- ***Remedies***
  - Learn and use profiling
    - https://pypi.org/project/py-spy
    - `sudo py-spy top --pid 57425`
  - Don't call methods multiple times from templates

SYMPTOM 2: SLOW SITE

- ***Cause 6: Slow data sources***
- ***Remedies***
  - Decouple (e.g. using redis/celery)
  - Async
  - Lazyload

SYMPTOM 3: CONFLICT ERRORS

SYMPTOM 3: CONFLICT ERRORS*

- ***Cause 1: Conflict resolving is not enabled***
- ***Remedy***
  - Add application code to the zeoserver

```
[zeoserver]
eggs = ${buildout:eggs}
```

*) *ConflictErrors happen when two transactions try to modify the same object at once. Example: transaction1 (t1) reads a obj and wants to change it but it takes a while. Meanwhile (after t1 has started) transaction2 (t2) reads the same obj und changes it quickly before t2 is done. When t1 is finally done and its change to obj are written to the database the state of obj is no longer the same as when it was read at the beginning of t1. This raises a ConflictError. If a ConflictError occurs Zope will attempt to replay a transaction up to three times.*

SYMPTOM 3: CONFLICT ERRORS

- ***Cause 2: Long running requests change data***
- ***Remedies***
  - Prevent writes
  - Do intermediate commits
  - Prevent Crossfire
    - Disable cronjobs and editors
  - Async

SYMPTOM 4: POSKEYERROR

SYMPTOM 4: POSKEYERRORS

- ***Cause 1: Missing blobs***
- ***Remedies***
  - Copy all blobs :)
  - Use `experimental.gracefulblobmissing`
  - Find and delete afflicted content

SYMPTOM 5: BROKEN DATA

ModuleNotFoundError

AttributeError

ImportError

POSKeyError

BrokenObject

www.starzel.de/blog/zodb-debugging

SYMPTOM 5: BROKEN DATA

- ***Cause 1: Code to unpickle data is missing***
- ***Remedies:***

1. Ignore the errors
2. Fix it with a rename_dict
3. Work around with a `alias_module` patch
4. Find our what and where broken objects are and then fix or remove them safely

SYMPTOM 5: BROKEN DATA

- ***Remedy 2: zodbupdate_rename_dict***
  - In setup.py:
    `entry_points={'zodbupdate': ['renames = mypackage:rename_dict']}`
    or in setup.cfg:

```
[options.entry_points]
zodbupdate =
    renames = mypackage:rename_dict
```

  - In `__init__.py` of mypackage:

```
iface = "zope.interface Interface"
rename_dict = {
    "App.interfaces IPersistentExtra": iface,
    "App.interfaces IUndoSupport": iface,
    "Products.ResourceRegistries.interfaces.settings IResourceRegistriesSettings":
     iface}
```

  - See `zest.zodbupdate`

SYMPTOM 5: BROKEN DATA

- ***Remedy 3: Work around with a alias_module patch***
  - In __init__.py:

```
from OFS.SimpleItem import SimpleItem
from plone.app.upgrade.utils import alias_module

class BBB(object):
    pass

SimpleBBB = SimpleItem

try:
    from collective.solr import interfaces
except ImportError:
    alias_module('collective.solr.indexer.SolrIndexProcessor', BBB)

try:
    from collective.easyslideshow.descriptors import SlideshowDescriptor
except ImportError:
    alias_module('collective.easyslideshow.descriptors.SlideshowDescriptor', SimpleBBB)
```

SYMPTOM 5: BROKEN DATA

- ***Remedy 4: Find our what and where broken objects are and then fix or remove them safely***

  1. Use **`zodbverify`** to get all broken objects
  2. Pick one error-type at a time
  3. Use **`zodbverify`** with **`-o <OID> -D`** to inspect one object and find out where that object is referenced
  4. Remove or fix the object

SYMPTOM 5: BROKEN DATA

- ***Step 1: Use zodbverify to get all broken objects***
  - Checkout zodbverify
  - Use https://github.com/plone/zodbverify/pull/8
  - `./bin/zodbverify -f var/filestorage/Data.fs`

    or

    `./bin/instance zodbverify`

```
2020-12-08 12:26:57,602 INFO    [zodbverify:48][MainThread] Done! Scanned 154351 records.
Found 43 records that could not be loaded.
Exceptions, how often they happened and which oids are affected:

AttributeError: module 'plone.app.upgrade.atcontentypes_bbb' has no attribute 'MetadataElementPolicy': 12
0x0126 0x0127 0x0128 0x0129 0x012a 0x012b 0x012c 0x012d 0x012e 0x012f 0x0130 0x0131

ModuleNotFoundError: No module named 'fourdigits': 8
0x28030f 0x280310 0x280311 0x280312 0x280313 0x280314 0x280315 0x280316

AttributeError: module 'plone.app.upgrade.atcontentypes_bbb' has no attribute 'ElementSpec': 7
0x011f 0x0120 0x0121 0x0122 0x0123 0x0124 0x0125

ModuleNotFoundError: No module named 'Products.Archetypes': 5
0x0e00eb 0x0e00ee 0x0e00ef 0x0e00f0 0x0e00f1

ModuleNotFoundError: No module named 'Products.ATContentTypes': 4
0x0e00e9 0x0e011a 0x0e01b3 0x0e0cb3

AttributeError: module 'plone.app.event.interfaces' has no attribute 'IEventSettings': 3
0x2a712b 0x2a712c 0x2a712d

AttributeError: module 'plone.app.upgrade.atcontentypes_bbb' has no attribute 'MetadataSchema': 2
0x25 0x011e

ModuleNotFoundError: No module named 'plone.app.controlpanel': 1
0x0f4c2b
```

SYMPTOM 5: BROKEN DATA

- ***Step 2: Pick one error-type at a time***
  - You will forget what you did!
  - Make notes
  - Write upgrade-steps
  - Keep the terminal log!

SYMPTOM 5: BROKEN DATA

- ***Step 3: Inspect one broken object***
  - `./bin/zodbverify -f var/filestorage/Data.fs -o 0x280311 -D`
  - Find out what it is
    - Look at the error
    - Look at the obj
    - Look at the pickle
  - Find out where it is
    - See where it is referenced

SYMPTOM 5: BROKEN DATA

- ***Step 4: Fix it***
  - Hack it away
  - Then write a repeatable upgrade-step

SYMPTOM 5: BROKEN DATA

- ***Step 5: Check that is is gone***
  - The broken object still exists
  - OID should not be referenced any more
  - Pack your database
  - Done :)

SYMPTOM 6: BAD CODE

SYMPTOM 6: BAD CODE

- ***Unreadable Code***
- ***Untested Code***
- ***Unused Code***
- ***Undocumented Code***
- ***Unmaintained Code***
- ***Complicated Code***
- ***Overtly Complex Code***
- ***Too Much Code***

starzelde

THANK YOU

** yes, we are for hire*