

# The package bazaars and the cathedrals



source: [https://en.wikipedia.org/wiki/Sagrada\\_Fam%C3%ADlia](https://en.wikipedia.org/wiki/Sagrada_Fam%C3%ADlia) CC BY-SA 3.0 by [Bernard Gagnon](#)

© 2018 nexB Inc. All rights reserved. Licensed under the CC-BY-SA-4.0 International license <https://creativecommons.org/licenses/by-sa/4.0/>

# Philippe Ombredanne

My mission: **make it easier to reuse FLOSS**

Enthusiast FLOSS developer

AboutCode, Linux kernel, a bit on strace, SPDX (and Eclipse, JBoss, and more)

CTO at nexB Inc. a software company helping software teams understand where their code comes from (and its licensing, vulnerability, quality, etc) with a combo of:

- FLOSS tools
- a commercial enterprise Dashboard

# Why do

`$ apt/rpm install/update`

# work all the times?

.... and why

```
$ pip/npm/bundle...install/update
```

does NOT work always?

# catb?

Last century..., esr wrote the seminal catb essay:

*“I had been preaching the Unix gospel of small tools, rapid prototyping and evolutionary programming for years. [...] I believed that the most important software (operating systems and really large tools like the Emacs programming editor) needed to be built like **cathedrals**, carefully crafted by individual wizards or small bands of mages working in splendid isolation, with no beta to be released before its time.*

*Linus Torvalds’s style of development—release early and often, delegate everything you can, be open to the point of promiscuity—came as a surprise.*

<http://www.catb.org/esr/writings/cathedral-bazaar/cathedral-bazaar/index.html>

# PACKAGE CATHEDRALS



# pAckage bAZaaRs



# Not the same packages!

- multiple OSes vs. single OS
- Language-specific vs. general purpose
- Author vs. Maintainers
- Builder vs. Users
- Test one vs. test them all
- Release alone vs. release together
- Simple deps vs. rich dep semantics
- Latest and greatest vs. working and compatible set
- As-is & unmodified vs. patched & deconstructed



VS.



sources: <https://www.amazon.com/Bucket-Building-Bricks-Blocks-Compatible/dp/B01JHGJSNC>  
<https://www.amazon.com/Nanoblock-Sagrada-Familia-Building-Construction/dp/B01DIQJSZA>

# Perspectives

As a **user**, I want a stable, integrated and working whole system

As a **builder**, I want a function-specific, single purpose and simple building block

# Dependencies

Language-specific PMs have simpler dependency models

- build or runtime
- limited conflicts resolution

Distro PM dependencies use rich vocabulary

- Build, Depends, Pre-Depends, Recommends, Suggests, Enhances, Breaks, Conflicts, Obsolete, etc.
- Boolean satisfiability based (e.g. SAT) complex conflicts resolution

# Social contract & coordination

Language-specific Packages are built by an individual

- single "function" focus
- no coordination
- utilitarian injection molding

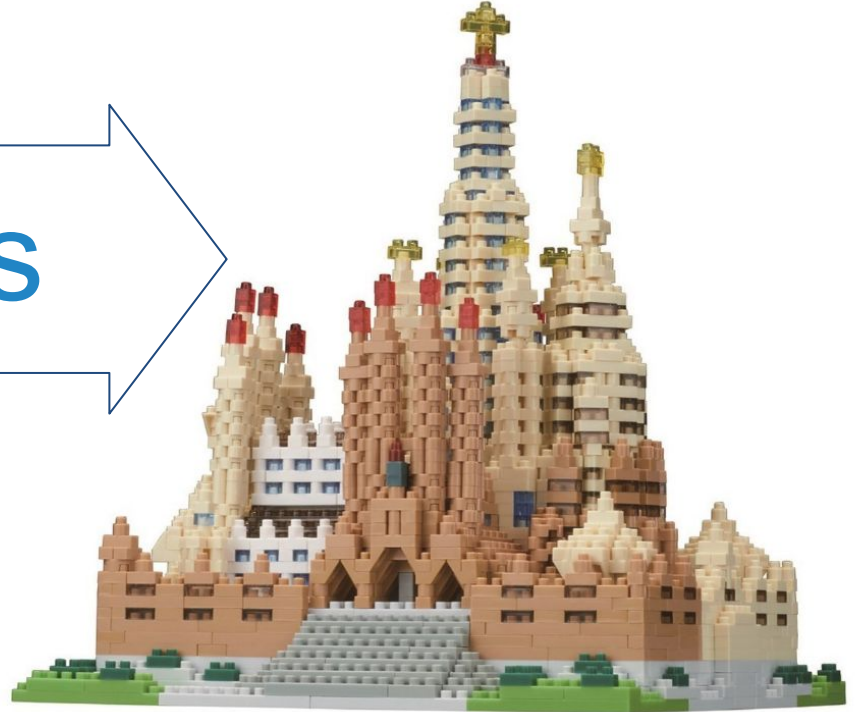
Distro Packages are built by a team

- whole system focus
- extensive coordination and collaboration
- art and architecture!

# bazaar & cathedrals all the way



yields



sources: <https://www.amazon.com/Bucket-Building-Bricks-Blocks->  
<https://www.amazon.com/Nanoblock-Sagrada-Familia-Building-Construction/dp/B01DIQJSZA>

# bazaar & cathedrals all the way

Each Language-specific Package is a mini cathedral

- single "function" focus
- no coordination

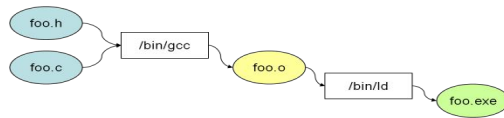
Distro Packages are built by a team

- whole system focus
- extensive coordination and collaboration

# We are all system integrators!

- Coordination is HARD!
- Call to action:
- I am a user and builder at the same time
- I will stop bitching about out-of-date distro packages
- **Distro packages need our love!**

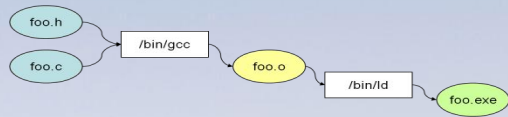
# Plug



Join me for **Build tracing with strace** 13:00 in the debugging tools dev room

# Thank you!





# Credits

Special thanks to all the people who made and released these awesome free resources:

- Presentation template by [SlidesCarnival](#)
- Photographs by [Unsplash](#)
- Icons from [openclipart.org](#)

And all the FLOSS software authors!