Business models for open source projects in environmental informatics

Andreas Ciroth

12 September 2008

Outline

1. Business models, def., why, & how
2. Elements of a business model,
   for open source projects
3. Three criteria and three parameters for BM
4. Niches for successful projects
5. A bad example, and maybe a good one
6. Final remark

GreenDeLTa TC

GreenDeLTa TC

1. Business models def., why, & how

Business models

Put very simple:

Business model =
model of a business, in a longer term perspective.

Revenues and costs obviously relate to a BM.

Why talk about business models,

Especially why for open source projects

- open source projects are long term undertakings
- they obviously deal with costs, and are businesses
- a whole zoo of different projects, range from volunteer / missionary / philosophical to the next business hype and motorway to huge profit.

Let's assume a neutral position about money, for the talk

Money is not good or bad, but simply a needed element for a successful open source project (like water for a tree).

And money is not the only element that is needed.

Plus there can, probably, also be too much money (as there can be too much water for a tree)

Thinking about a sound (healthy) business model for a project helps creating a healthy, successful project

The business model can be laid down in a plan, but this plan needs to be lived (no 60 pages with financial quarterly figures for the next 4 years..)

2. Elements of a business model, for open source projects

Business model, the financial side

Consideration of
all costs and revenues (…)
related to
all products
of the open source project.

Potential costs, software projects in general

- coding
    - *salaries
    - *hardware
    - *licence fees
    - *subcontracting
- distribution
    - *webserver?
    - *documentation
- sales & support (salaries, travel, documentation)
- management, overhead

Potential benefits, software projects in general

- licence sales
- information material sales
- training courses
- renting the software
- projects where the software is specifically adapted to user needs
- benefits from using existing „software pieces“
- donations
- recognition for the company / people involved

GreenDeLTa TC

Potential costs, open source software projects

- coding
  - *salaries
  - *hardware
  - (*licence fees)
  - *subcontracting
- (distribution
  - *webserver?
  - *documentation)
- (sales&support (salaries, travel, documentation))
- management, overhead

Potential benefits, open source software projects

- ~~licence sales~~
- information material sales +
- training courses +
- ~~renting the software~~
- projects where the software is specifically adapted to user needs (+) / +
- benefits from using existing „software pieces“ + +
- donations (+)
- recognition for the company / people involved +

Influence of the number of licences used, software in general

costs:

Coding 0

Distribution (+)

Sales (+)

---

benefits:

Training +

Licence fees + (closed source)

Request for specific adaptations +

Benefits from 3rd party software + (!; open source)

3. Three knock-out criteria, and three parameters, for open source business models

Three knock-out criteria essential for every open source project

1. a core team that is able to both develop the software, and to manage and guide the whole project.
   "The key developers need to work on it full time."

2. an agile, interested, and to some extent integrated and supporting community.

3. a project that offers something for users, attracting them from other software they use, or convincing them to use software at all for a problem they have.

Three parameters shaping an open source business model

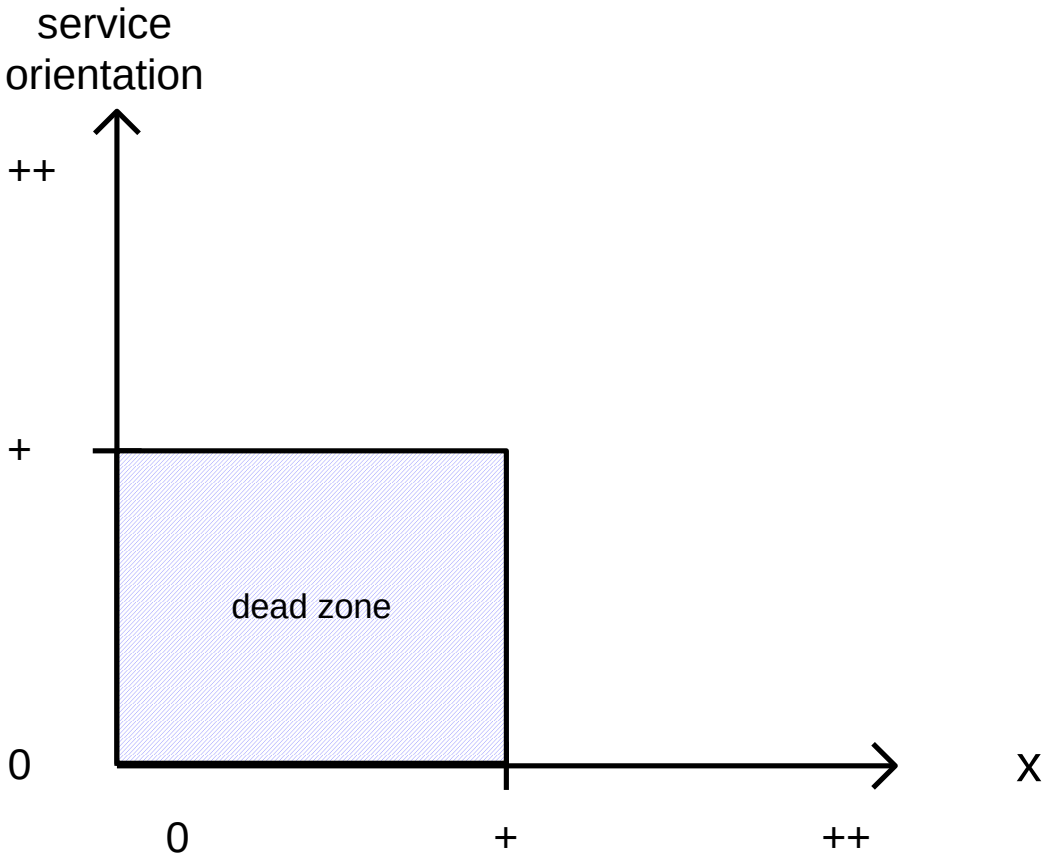
1 the number of licences used;

2 the usual licence fee; and

3 the degree service, and modification projects are to be expected for the software (“service orientation”)

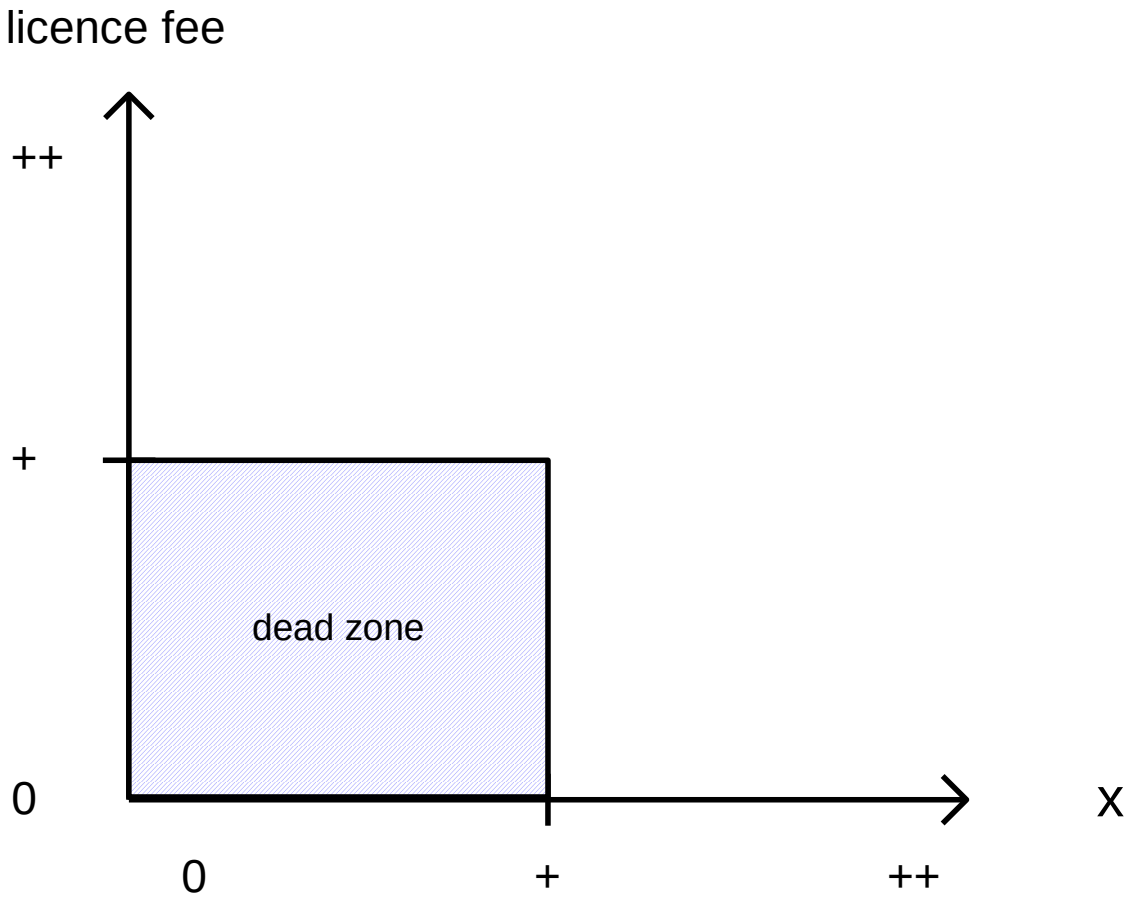
4. Niches for successful projects

Dead zone 1:
low service orientation, low number of licences



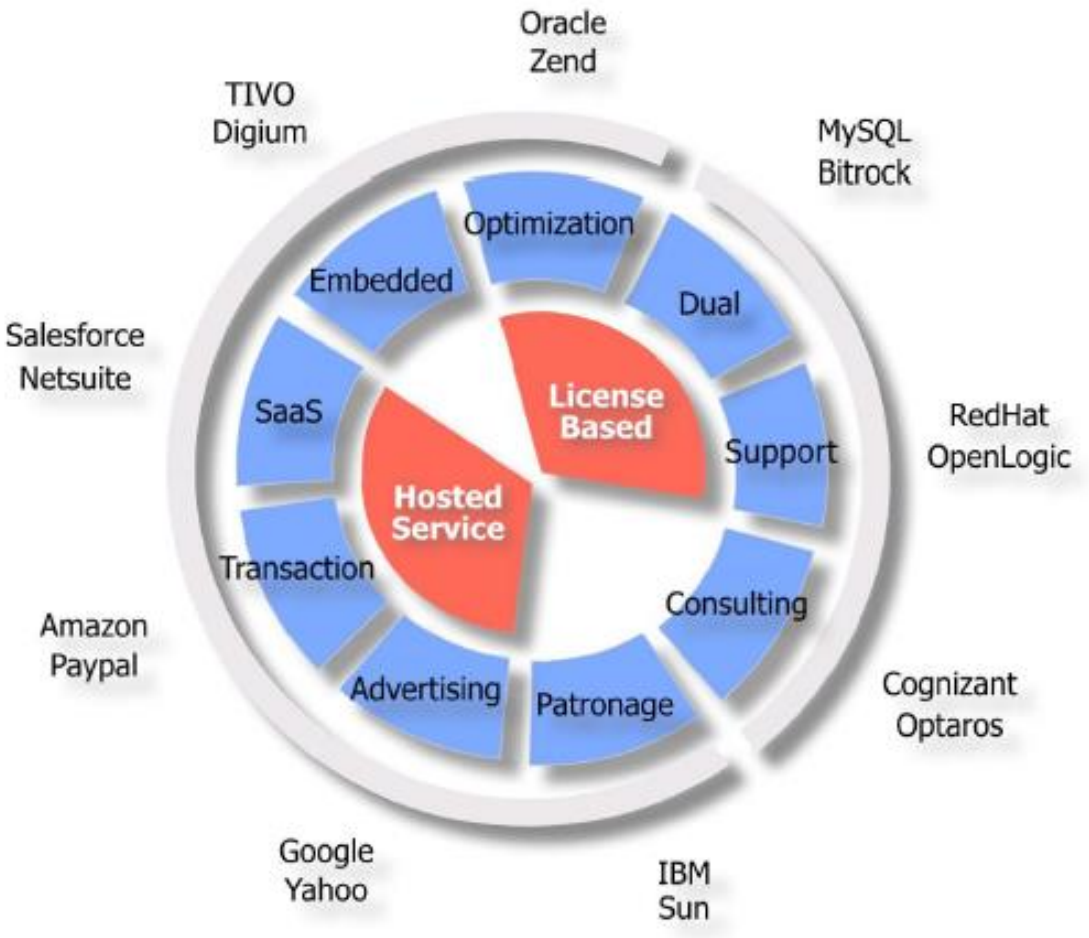
++ means: Very high, x means number of licences

Dead zone 2: low licence fee of comparable software, low number of licences



++ means: Very high, x means number of licences

The specific business model you choose may vary..



Koenig, J.: Seven open source business strategies for competitive advantage, Riseforth Special Report, issue 15984, Half Moon Bay, CA, 2007

..but it should always reflect
- market changes,
- changeing user needs,
- feedback on previous releases

Design a pattern that is followed rather than a monolithic plan or roadmap
that is strictly adhered to.

4. A bad example, and maybe a good one

openLCA – application for public funding (Investitionsbank Berlin, ProfIT)

ProfIT: Local program for supporting SMEs in IT

Approached spring 2006 with the idea of an open source, Eclipse RCP Life Cycle Assessment application

„Quick evaluation“: One reviewer.

First Business plan (ca. 40 pages): „Will not work“

Revision: 65 pages, quarterly overview over 4 years, 450‘ €; approved, as a credit

(we turned this down)

openLCA – low level start

Conference presentation: Three interested financial supporters

Piecewise release (format converter summer and autumn 2007)

Testing possible if someone sends email and shows earnest interest → 40 testers worldwide

For GreenDeltaTC: Unique recognition, various won contratcs in connection with openLCA

openLCA – low level start

Business model pattern: Provide LCA expertise and a free and configurable LCA software.

High license fees of comparable software, high degree of service orientation, but rather low numbers of licences → we are not in a dead zone..

(see also presentation yesterday & in proceedings:
openLCA - implications of an emerging open source software for sustainability assessment, Andreas Ciroth, Michael Srocka, Jutta Hildenbrand)

6. Final remark

Dead zone 2:
low licence fee of comparable software,
low number of licences

→influenced by
other available open source software!

→ Collaborate.

Vielen Dank!

Dr. Andreas Ciroth

GreenDeltaTC GmbH, Berlin, Germany

ciroth@greendeltatc.com