

Sustainable Software Business Models

Obsidian Case Study & Arborist Strategy

Building Without Rent-Seeking

How to Live, Build, and Serve Users Ethically

Business Model Analysis

January 2026
Version 1.0

Abstract

This document analyzes Obsidian's business model as a case study in sustainable software development, mapping open-source vs. proprietary components, revenue streams, user pain points, and value propositions. It then proposes an ethical business model for Arborist that avoids rent-seeking while enabling the developer to sustain themselves and continue serving the user base. The goal is to find the balance between "free as in freedom" and "I need to eat."

Contents

I Obsidian Business Model Analysis	2
1 Component Architecture	2
1.1 What's Open vs. Closed	2
1.2 The Clever Part	2
2 Revenue Streams	2
2.1 Pricing Structure (2025)	2
2.2 Revenue Analysis	3
3 User Pain Points	3
3.1 Pain Point Analysis	3
3.2 The "Rent-Seeking" Perception	4
4 Value Proposition Analysis	4
II Business Model Philosophy	5
5 The Spectrum of Software Business Models	5
5.1 Defining "Rent-Seeking" in Software	5
5.2 The Ethical Software Business Test	5

6 Sustainable Revenue Models	6
6.1 Model Comparison	6
6.2 The “Freemium Done Right” Formula	6
 III Arborist Business Model	 6
7 Proposed Architecture	6
7.1 Open Source vs. Monetized Components	6
7.2 Key Differentiator from Obsidian	7
 8 Revenue Strategy	 7
8.1 Pricing Philosophy	7
8.2 Proposed Pricing	8
8.3 Revenue Projections	8
 9 The Self-Hosting Escape Valve	 8
9.1 Why Open Source the Server?	8
9.2 Self-Hosting Documentation	9
 10 Additional Revenue Streams	 9
10.1 Non-Extractive Options	9
10.2 What NOT to Do	9
 11 Sustainability Roadmap	 10
11.1 Phase 1: Foundation (Months 1-6)	10
11.2 Phase 2: Services (Months 6-12)	10
11.3 Phase 3: Growth (Months 12-24)	10
11.4 Phase 4: Sustainability (Year 2+)	10
 12 The “Living Wage” Math	 11
12.1 What Do You Actually Need?	11
12.2 Path to Sustainability	11
 13 Summary: The Arborist Compact	 12

Part I

Obsidian Business Model Analysis

1 Component Architecture

1.1 What's Open vs. Closed

Component	Open Source	Free	Notes
Core Application			
Desktop App	No	Yes	Electron-based, proprietary code
Mobile App	No	Yes	Capacitor-based, proprietary
Markdown Renderer	No	Yes	Custom implementation
Graph View	No	Yes	Built into app
Plugin API	Yes	Yes	API is documented
Data & Storage			
Local Vault (files)	Yes	Yes	Plain .md files you own
File Format	Yes	Yes	Standard Markdown
Plugin Settings	Yes	Yes	JSON files in vault
Ecosystem			
Community Plugins	Yes	Yes	User-created, open source
Community Themes	Yes	Yes	CSS, open source
Help Documentation	Yes	Yes	GitHub repository
Paid Services			
Obsidian Sync	No	No	\$5/mo - E2E encrypted sync
Obsidian Publish	No	No	\$10/mo - Web publishing
Commercial License	N/A	No	\$50/user/year - Honor system

Table 1: Obsidian Component Breakdown

1.2 The Clever Part

Obsidian's genius is in **what they made free**:

- **Your data is always yours** — Plain markdown files, no lock-in
- **The core app is genuinely useful** — Not a crippled demo
- **Extensibility is free** — Plugins can replace paid features
- **No telemetry** — They can't even enforce commercial licenses

They make money on **convenience and infrastructure**, not on features you need.

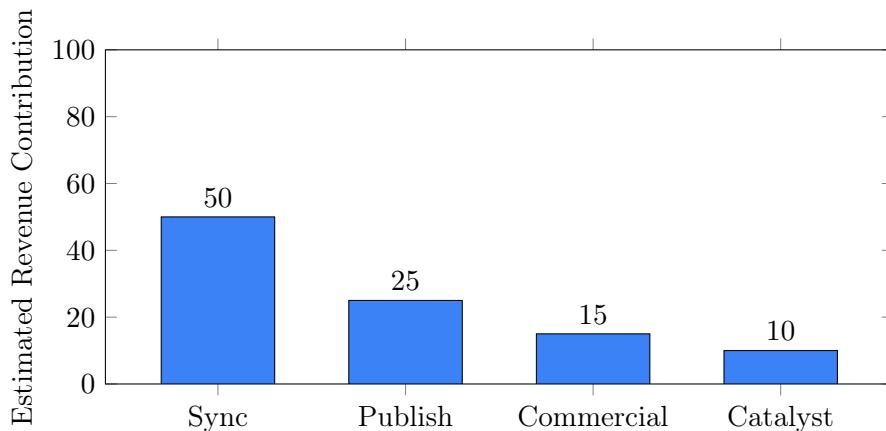
2 Revenue Streams

2.1 Pricing Structure (2025)

Product	Monthly	Annual	What You Get
Personal Use	Free	Free	Full app, unlimited vaults
Sync	\$5	\$48	Cross-device sync, E2E encryption, 1yr history
Publish	\$10	\$96	Web hosting, 4GB, custom domain, graph view
Sync + Publish	\$15	\$144	Both services
Commercial	—	\$50/user	License for work use (honor system)
Catalyst	\$25+	One-time	Beta access, badges, support development

Table 2: Obsidian Pricing

2.2 Revenue Analysis



Key insight: Sync is likely their biggest revenue driver because:

- It solves a universal problem (multi-device access)
- The free alternatives (iCloud, Dropbox) have real friction
- E2E encryption is a genuine differentiator
- It's "set and forget" — low churn

3 User Pain Points

3.1 Pain Point Analysis

Pain Points by User Segment

Free Users:

- Multi-device sync requires manual setup (iCloud, Syncthing, Git)
- Publishing requires technical knowledge (Quartz, Hugo, etc.)
- No official support channel
- Mobile app without sync is limited utility

Sync Subscribers (\$5/mo):

- Can't share vaults with non-subscribers easily
- No real-time collaboration
- 10GB storage limit (fills up with attachments)
- If you stop paying, sync stops (but files remain)

Publish Subscribers (\$10/mo):

- \$120/year is expensive vs. alternatives (\$0 for Quartz)
- 4GB storage limit
- Limited customization vs. self-hosted
- Can't password-protect individual pages (all or nothing)
- No collaborative editing for teams

Commercial Users (\$50/user/year):

- Honor system feels awkward
- No additional features for the money
- Teams still can't collaborate in real-time
- No audit logs, admin controls, SSO

3.2 The “Rent-Seeking” Perception

Some users feel Obsidian's pricing is rent-seeking because:

1. **Publish competes with free tools** — Quartz does 90% of what Publish does for \$0
2. **Sync competes with free tools** — iCloud/Dropbox work, just with friction
3. **No bundle discount** — \$15/mo for both feels like nickel-and-diming
4. **Annual cost creep** — \$180/year for Sync+Publish adds up

However, Obsidian largely avoids true rent-seeking because:

- Your data is never locked in
- The free tier is genuinely complete
- You can replicate paid features with effort
- They're not harvesting/selling data

4 Value Proposition Analysis

Value Proposition by Tier

Free Tier Value:

- Full-featured note-taking app
- Graph view, backlinks, search
- Plugin ecosystem (1000+ plugins)
- Theme customization
- Canvas for visual thinking
- Local-first, privacy-respecting
- **Value:** \$0 cost, high utility

Sync Value (\$5/mo):

- Zero-config cross-device sync
- End-to-end encryption (they can't read your notes)
- 1-year version history
- Works instantly on new devices

- **Value:** Time saved > \$5/mo for most users

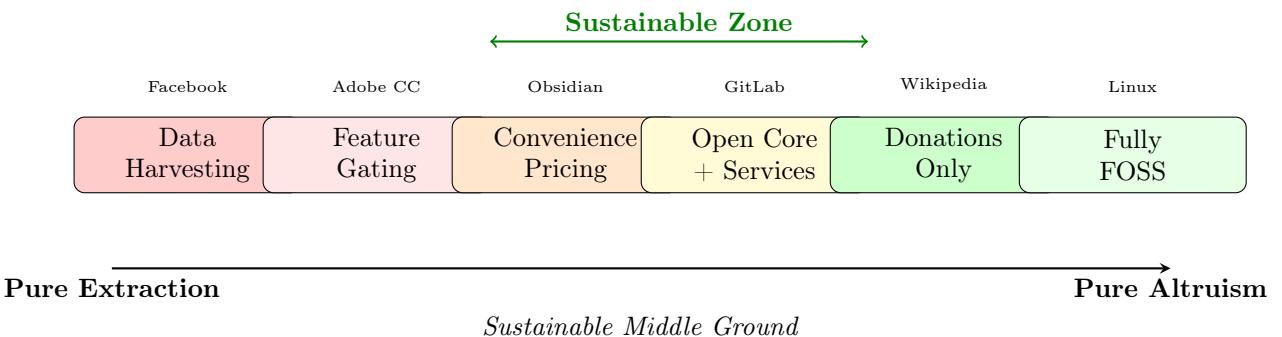
Publish Value (\$10/mo):

- One-click publish from app
- No technical knowledge required
- Custom domain support
- Graph view, backlinks on web
- **Value:** Debatable — technical users prefer Quartz

Part II

Business Model Philosophy

5 The Spectrum of Software Business Models



5.1 Defining “Rent-Seeking” in Software

What Makes Software Rent-Seeking?

Classic Rent-Seeking Behaviors:

1. **Artificial Scarcity** — Limiting features that cost nothing to provide
2. **Lock-in** — Making it hard/impossible to leave
3. **Regulatory Capture** — Using legal means to block competition
4. **Data Hostage** — “Pay or lose your data”
5. **Subscription Creep** — Converting one-time purchases to recurring
6. **Enshittification** — Degrading free tier to force upgrades

What’s NOT Rent-Seeking:

1. Charging for infrastructure you actually run (servers, bandwidth)
2. Charging for support/SLAs
3. Charging for convenience (time savings)
4. Offering premium features that cost more to develop
5. Accepting voluntary support (donations, sponsorships)

5.2 The Ethical Software Business Test

Ask these questions about any monetization strategy:

1. **Can users leave?** — Is their data portable?
2. **Can users self-host?** — Are alternatives available?
3. **Is the free tier crippled?** — Or genuinely useful?
4. **Are you charging for real costs?** — Or artificial scarcity?
5. **Would you pay for this?** — At this price, for this value?

6 Sustainable Revenue Models

6.1 Model Comparison

Model	Sustainable	Ethical	Scalable	Notes
Pure Donations	Maybe	Yes	No	Works for individuals, not teams
Sponsorships	Yes	Yes	Maybe	Depends on finding sponsors
Open Core	Yes	Yes	Yes	Free core + paid enterprise
Hosted Services	Yes	Yes	Yes	Charge for infrastructure
Support/SLA	Yes	Yes	Yes	Charge for guarantees
Dual License	Yes	Mostly	Yes	AGPL + Commercial
Marketplace Cut	Yes	Mostly	Yes	Tax on plugin ecosystem
Consulting	Yes	Yes	No	Doesn't scale

Table 3: Revenue Model Comparison

6.2 The “Freemium Done Right” Formula

Sustainable Freemium Principles

1. **Free tier must be genuinely useful** — Not a demo
2. **Paid tier must solve real problems** — Not artificial ones
3. **Price must reflect value delivered** — Not what market will bear
4. **Data must always be portable** — No lock-in
5. **Self-hosting must be possible** — Even if inconvenient
6. **Core functionality never paywalled** — Only convenience

Part III

Arborist Business Model

7 Proposed Architecture

7.1 Open Source vs. Monetized Components

Component	Open Source	Free	Rationale
Core Application (MIT/Apache License)			
Desktop App	Yes	Yes	Build trust, allow contributions
File Format Spec	Yes	Yes	Prevent lock-in
Local-first CRDT Engine	Yes	Yes	Differentiator, community benefit
Plugin API	Yes	Yes	Ecosystem growth
Export/Import	Yes	Yes	Data freedom
Self-Hostable (AGPL License)			
Sync Server	Yes	Yes	AGPL keeps it open if modified
Publish/Static Export	Yes	Yes	Users can host themselves
Hosted Services (Paid)			
Arborist Cloud Sync	Source Available	No	Real infrastructure cost
Arborist Publish	Source Available	No	Real hosting cost
Team Management	No	No	Enterprise feature
Priority Support	N/A	No	Time cost

Table 4: Arborist Component Strategy

7.2 Key Differentiator from Obsidian

The Arborist Difference

Unlike Obsidian, Arborist will:

1. **Open source the sync server** — Users can self-host for free
2. **Open source the publish system** — Like Quartz, but official
3. **Charge only for hosting** — Not for software features
4. **Never paywall core features** — Collaboration, graph view, etc.

The value proposition for paid services becomes:

- “You could run this yourself, but we’ll do it better and cheaper”
- “Your time is worth more than \$X/month”
- “You’re supporting continued development”

8 Revenue Strategy

8.1 Pricing Philosophy

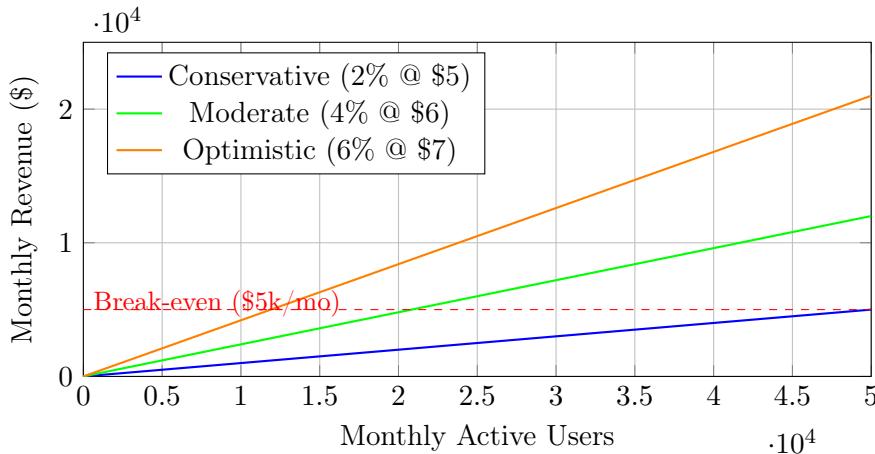
1. **Price based on cost + margin, not “value”** — Avoid extractive pricing
2. **Be transparent about costs** — Show users what they’re paying for
3. **Offer annual discounts** — Reward commitment, reduce churn
4. **Grandfather early users** — Build loyalty
5. **Student/nonprofit discounts** — Expand access

Tier	Monthly	Annual	Includes
Free (Self-Host)	\$0	\$0	Full app, sync server code, publish tools
Seedling (Solo)	\$3	\$30	5GB sync, 1 publish site, email support
Sapling (Pro)	\$8	\$80	25GB sync, 3 publish sites, priority support
Grove (Team)	\$6/user	\$60/user	Shared Trunks, admin panel, SSO, SLA
Supporter	\$5+	\$50+	Badge, beta access, Discord role

Table 5: Proposed Arborist Pricing

8.2 Proposed Pricing

8.3 Revenue Projections



Key insight: With conservative assumptions (2% conversion, \$5 avg):

- 10,000 MAU = \$1,000/mo (side project)
- 25,000 MAU = \$2,500/mo (ramen profitable)
- 50,000 MAU = \$5,000/mo (sustainable solo dev)

9 The Self-Hosting Escape Valve

9.1 Why Open Source the Server?

The Counterintuitive Advantage

Making sync/publish self-hostable seems like giving away revenue, but:

1. **Builds trust** — Users know they can leave
2. **Attracts developers** — Who contribute improvements
3. **Converts some self-hosters** — Who get tired of maintenance
4. **Creates word-of-mouth** — “They’re not trying to lock me in”
5. **Reduces support burden** — Power users self-support

The reality: Most users **won’t** self-host because:

- It requires a server (\$5-20/mo on its own)

- It requires maintenance (SSL certs, updates, backups)
- It requires technical knowledge
- Their time is worth more than \$3-8/mo

9.2 Self-Hosting Documentation

Provide excellent self-hosting docs:

- Docker Compose one-liner
- Kubernetes Helm chart
- Terraform modules for AWS/GCP/Azure
- Raspberry Pi guide (for hobbyists)

This **increases** paid conversions by demonstrating the complexity you're saving them from.

10 Additional Revenue Streams

10.1 Non-Extractive Options

Revenue Source	Potential	Notes
GitHub Sponsors	\$500-5k/mo	For open source work
Open Collective	\$200-2k/mo	Corporate sponsors
Consulting	\$150/hr	Custom deployments, training
Plugin Marketplace (10%)	Variable	If ecosystem grows
Merch	Minimal	Stickers, shirts (brand building)
Courses/Books	\$1-10k one-time	“Mastering Arborist”
Conference Talks	\$500-2k/talk	Plus travel, exposure

Table 6: Supplementary Revenue

10.2 What NOT to Do

Avoid These Traps

1. **Don't sell user data** — Ever
2. **Don't add ads** — Degrades experience
3. **Don't cripple free tier over time** — Enshittification
4. **Don't acquire and rug-pull** — If you sell, protect users
5. **Don't paywall security updates** — Ethical obligation
6. **Don't bundle crypto/AI grafts** — Stay focused

11 Sustainability Roadmap

11.1 Phase 1: Foundation (Months 1-6)

Focus: Build core product, establish trust

- Release open source desktop app (MIT license)
- Document file format specification
- Publish sync protocol specification
- Accept GitHub Sponsors
- Build community (Discord, forum)
- Goal: 1,000 users, \$0-500/mo sponsors

11.2 Phase 2: Services (Months 6-12)

Focus: Launch hosted services

- Launch Arborist Cloud Sync (beta, free)
- Launch Arborist Publish (beta, free)
- Gather feedback, iterate
- Convert beta users to paid (\$3/mo early-bird)
- Release self-hosted sync server (AGPL)
- Goal: 5,000 users, \$1,500/mo revenue

11.3 Phase 3: Growth (Months 12-24)

Focus: Scale sustainably

- Full pricing launch
- Team/enterprise tier
- Plugin marketplace
- Mobile apps
- Hire first contractor/employee
- Goal: 25,000 users, \$5,000/mo revenue

11.4 Phase 4: Sustainability (Year 2+)

Focus: Long-term stability

- Profitable enough for 2-3 person team
- Community governance for open source parts
- Consider foundation/co-op structure
- Resist acquisition pressure
- Goal: Indefinite sustainability, not infinite growth

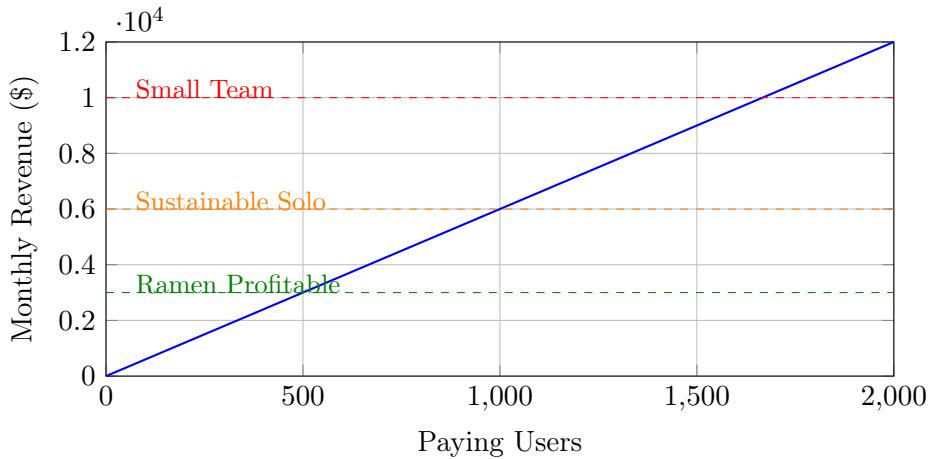
12 The “Living Wage” Math

12.1 What Do You Actually Need?

Expense	Monthly	Notes
Personal		
Living expenses	\$3,000-5,000	Location dependent
Health insurance	\$500-1,000	US freelancer rates
Retirement savings	\$500-1,000	15-20% of income
Taxes (set aside)	25-30%	Self-employment tax
Business		
Servers (small)	\$50-200	DigitalOcean/Hetzner
Servers (scaled)	\$500-2,000	As users grow
Domain/SSL	\$20	Annual amortized
Tools/services	\$100-300	GitHub, email, monitoring
Legal/accounting	\$100-500	Annual amortized
Total needed	\$6,000-10,000	To live + run business

Table 7: Monthly Financial Requirements

12.2 Path to Sustainability



The magic numbers (at \$6 avg revenue per paying user):

- **500 paying users** = \$3,000/mo (quit your job, barely)
- **1,000 paying users** = \$6,000/mo (comfortable solo)
- **1,500 paying users** = \$9,000/mo (hire help)
- **2,000 paying users** = \$12,000/mo (small team)

13 Summary: The Arborist Compact

The Promise to Users

1. **Your data is yours** — Always exportable, standard formats
2. **Core features are free forever** — Not a demo, not crippled
3. **You can self-host everything** — We provide the tools
4. **We charge for hosting, not features** — Real costs, fair prices
5. **We'll never sell your data** — Privacy is fundamental
6. **We'll communicate openly** — About pricing, roadmap, struggles

The Promise to Yourself

1. **You deserve to be paid** — This is real work
2. **Sustainability > growth** — Don't chase VC metrics
3. **Users who pay are partners** — Treat them well
4. **It's okay to be small** — 1,000 happy users beats 100,000 unhappy ones
5. **Your values matter** — Don't compromise for money

Build something you're proud of.

Charge fairly for real value.

Treat users like humans.

That's not rent-seeking. That's a business.
