



# Composr

## Opening up Composr development

By Chris Graham (Composr lead developer / ocProducts CEO)



# Background – The Past

- Composr (previously ocPortal) started out as a “shareware” style product. It was free, unless you wanted the logo removed, in which case there was a reasonably-priced fee. Relatively soon after it became fully Open Source, and even gave an option for agencies to make the CMS “white label” for no cost.
- Throughout, the development model has been a “cathedral” model, with private decision making and a core team of salaried developers working for ocProducts (the lead developer’s company). This model has been intentional and necessary to create a very elegant system done to high-professional standards.
- This model relies on a large range of low-to-medium budget web design projects, with innovations from these projects going back as new Composr features, as well as the profit-margin going back into funding development (instead of a marketing budget – Composr being Open Source was in effect ocProduct’s marketing).



# Background – The Present

- This model has become increasingly broken, as economic-equality problems, enormous developer salaries, the rising cost of web development, and the maturation of the industry around wealthy incumbents, has put the cost of a high-quality feature-filled website well out of reach the average middle-class ‘Western’ person. Even just doing a good-quality custom responsive design is several times the typical budget, let alone coding major new functionality. Even composr support credits, which ocProducts makes a loss on, are outside the budget of the vast majority of Composr CMS users.
- This cost problem affects all web development projects across the industry profoundly, but is not being openly discussed. People do seem to now widely realise it, although perception has always been a few years behind reality.
- ocProducts is financially stable, but relies on much higher-end projects for income nowadays. The large commitment for these projects (including travel, and very complex multi-disciplinary team-based work) makes it difficult to focus 2 to 3 days a week on core Composr development – and the features from enterprise projects don’t tend to be the typical new features the average user will benefit from.
- Chris (me) is also recently married, and has spent much of 2016/2017 split between the USA and UK, in legal-limbo before becoming an immigrant to the USA.



# Background – Composr growth (1)

- Composr has always had a smaller community than other important CMSs. There are many reasons:
  - **Age.** Composr is a few years younger than other CMSs, so was always up against more-established systems.
  - **Complexity.** Composr is the most flexible feature-filled CMS. The focus on providing a lot of functionality, but building it on a flexible architecture rather than as an “out of the box” product, has traditionally meant Composr users had to be smarter than average, and Composr themers much smarter than average. Composr has got far better at compensating for this over the years via some clever innovations, so this is not such a problem as it was, and will be even better handled in version 11.  
It’s important to stress that combining flexibility and features is key providing the product users need, it’s just something that has presented major challenges.
  - **Feature-focus.** We have traditionally focused more on features than design (expecting users to do their own theming to get the look they want). Realistically the average user cares about the default design more than features, and increasingly so – and cannot afford to have a high-quality design of their own made. We had to do a major refocus to address this, and continue to make design our top priority into v11.



# Background – Composr growth (2)

- **Corporate-feel and lack of buy-in.** Composr is developed by a for-profit company (\*). Users may be put off by the fact that there is a barrier between development priorities (framed as investments by ocProducts), and the fact they just want a cool free product that does what they want. For your typical Open Source project, passionate volunteers will make uneconomical things happen through passion, but if they don't feel they have true collective-credit they aren't going to be as forthcoming. This probably is subconscious, nobody has ever expressed it to me, but I think it is true.

As explained on slide 2 our development model is for good reason, but this is the down-side of the model.

\* Technically. In reality Chris [me] always cared much more about Composr than ocProduct's profit.

- **Chicken and egg.** A smaller user-base has led to a smaller number of themes being released, which has led to a smaller user-base.

- **No external investors.** We have no institutional investors in ocProducts like Wordpress and Drupal have in their developer-led companies. This is to avoid our direction being too tainted by the interests of bankers. I am very glad we made this decision – it has tied Wordpress into being a system for the common-denominator of users, and Drupal into moving into an enterprise-focused project. A middle-ground wide-appeal CMS product would be destroyed by investors. Never-the-less, it has had an impact on the amount of money in the ecosystem. Acquia took over \$173 million so far.



# Background – Composr growth (3)

- To summarise, the average user of an Open Source product is doing something out of passion. They are not guiding their efforts as investment decisions, and will not feel equals if existing in an environment where there's a higher authority doing that. This has unintentionally caused a lot of friction against the passions of the average user. A friction that largely will have existed subconsciously in most people's minds, and therefore not ever clearly expressed.
- On the plus side, the development model has led to an awesome product that can run circles around other Open Source CMSs (at least for what it is trying to accomplish).
- The original developers of Composr (Chris Graham, Robbie Goacher, Allen Ellis, Philip Withnall) have always been very open liberal-minded people, aiming to create a product for the world, and avoiding getting pulled off-course by taking investments – as opposed to ardent capitalists trying to make a fortune. The mistake, largely by Chris (me), has been trying to walk a tight-rope between the capitalist and community-orientated worlds; a tight-rope that only a relatively small number of people are happy to be on (most people prefer to work in a more clear-cut environment). Perception is key, and programmers with a INTJ (“mastermind”) personality may be a bit slow to pick up on that ;-).



# The Future

- At this point Composr has very solid foundations, so the “too many cooks” problem that the cathedral development model solves no longer needs solving.
- We should now move development to a fully open model, and firewall ocProducts from any direct control of Compor CMS.
- This may be done by creating an official legal entity, “The Composr Foundation”, or it may be done on a less formal basis. Likely we will start simple, involving more and more people, and formalise things as participation increases.
- It is expected to get this done after Composr v11 is released, hopefully some time in 2018. Composr v11 should come with a relaunched compo.sr website that is fully community-orientated. A basic constitution would be put in place. Elections would follow to pick a Chairman and a Lead Programmer. Chris (me) would put in candidacy for Lead Programmer, and we hope someone who has never worked for ocProducts could be elected Chairman. As a free election the community could vote for whoever they wanted, and there’d be regular fresh elections.



# Constitution

- A constitution needs to cover:
  - The elected posts that exist (initially Chairman and Lead Programmer); and frequency of elections
  - Mechanism for amending the constitution
  - Declaration of vision / principles (likely lifted from the current compo.sr Vision webpage)
  - A community code of conduct (personally I'd vote to keep it very simple, as this can get very political, and I don't believe Composr should get embroiled in battles between political ideologies – but we'd find some kind of democratic way of dealing with this)
  - Complaints/impeachment procedures
  - How Composr code copyright will be assigned (e.g. dual-copyright), and if a contributor's agreement is needed
  - How Github access, and Tracker private-issue-access, is assigned
  - How basic costs are financed (e.g. it is likely ocProducts will need to regularly sponsor some amount of money to get things going); and any reward major sponsors should expect (e.g. advertising).
- We may want to look at other major CMSs do (e.g. Drupal, Joomla) to avoid reinventing the wheel. At the same time, we probably don't want to guide Composr into being a clone of another system, and we probably don't want to create a massive bureaucracy.



# Google Hangouts

- We need to make the Composr community much more social, so that people feel connected and like they are truly a part of a family. This is key to the growth of the other big CMS communities.
- This doesn't come easy for everyone, as a lot of people are introverts. But when extroverts really are 100% in on a community, that community will grow a lot faster. And even introverts like to talk.
- Regular Google Hangouts, for perhaps say 2 hours, weekly, would probably be a good solution. Chris is willing to attend at least most of these, and we can talk about pressing issues, write code, answer support questions, or just hang out.
- It is a good way for developers like Chris to do some free support for people. Realistically doing an open-ended amount (e.g. by e-mail) isn't practical – but doing free support for limited concentrated periods of time is practical and implicitly people will understand the reasonable limitations if it is done this way.
- If a particular developer is leading a session then “Super Chat” can be used to fund the activity within work hours (people paying to ask questions would get priority in the chat, a successful model for many YouTubers).



# Roadmap outline

- Finish Composr version 11
- Finish new ocproducts.com website (ocProducts may also be reincorporated in the USA, due to Chris's move)
- Finish new compo.sr website, with compo.sr firewalled from ocProducts (any outstanding support credits would be honoured via e-mail support from ocProducts of course)
- Write the new constitution, democratically as a community, likely using Google Hangouts. Possibility of physical meetups if interested people happen to be near
- Some work updating a lot of code, text, and documents, to reflect all the organisational changes
- Run elections for Chairman and Lead Developer
- *Proceed along our new path!*
- Possibility of more formal Composr roles, e.g. head of marketing, lead Denmark evangelist
- Formalise Composr Foundation at some point in the future, if useful



# Addressing concerns

- Anticipating some possible concerns:
  - “ocProducts is abandoning Composr, the product will now fall into disrepair”  
ocProducts is profitable and will continue to support Composr. With a greater number of contributors, Chris can actually better focus his expertise where it will count the most.
  - “Standards will fall”  
Composr is a mature product, v11 has an enormous coding standards guide, and we’re investing in additional unit testing. We’re at a point where this should not be a major risk.
  - “Somebody needs to take responsibility for bug fixing”  
Contributors are going to want to fix bugs still. Corporations needing bugs fixed within a certain timeline, or wanting bugs to be hand-patched, can have a contract directly with any company offering it (e.g. ocProducts).
  - “ocProducts is going to become far too expensive for the little guy; reasonably priced support plans will not be available now that ocProducts is not explicitly serving this market via compo.sr”  
Honestly, it already is, as is every other company, as are the existing support plans – and this can’t realistically be changed unless the world economy goes back to 2008. Any skilled developers hired commercially are not going to work at less than market rate, including employees of agencies like ocProducts. Opening up Composr so there are more enthusiastic volunteers is the only solution to the problem.

# Thank you.

I hope you're enthusiastic for what could be the next big phase of the product you love.





# Appendix: Business models

The primary reasoning for opening Composr development is to better pull in people's passions, to create a family that user's feel a part of. But the business reasoning is important too. To clarify the decision making process behind this from a business point of view, here are all the options ocProducts has:

- ocProducts targets small business again  
Problem: Chris would have to work for around minimum wage, and we could not hire anyone new.
- ocProducts makes Composr CMS a paid product.  
Problem: This goes against our fundamental values.
- ocProducts continues as-is.  
Problem: Composr would not get the focus it deserves.
- ocProducts grows much bigger to fund full-time resources back into Composr  
Problem: Growing ocProducts into a much larger company would actually lead to a business that didn't invest in Composr at all. Maintaining a business like that would take too much focus.
- ocProducts takes a big investment or a big corporate sponsor  
Problem: The money would come with too many demands, and corrupt the mission.