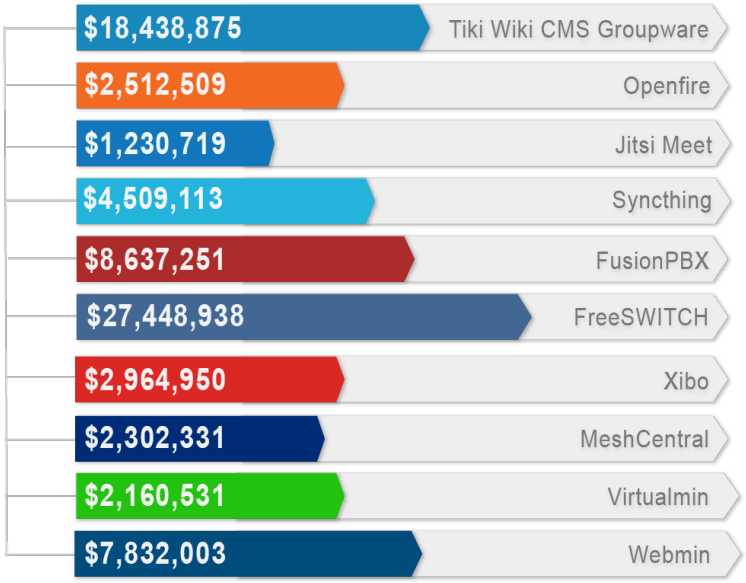
Constructive Cost Model (COCOMO)

WikiSuite is the most comprehensive and integrated Open Source enterprise solution, so you'd expect that it took a massive amount of work. But how much? WikiSuite's cost to develop is well over US$50 million.

WikiSuite.org
collaborate efficiently

The cost to develop is well over
50 million $ US



Calculated by Constructive Cost Model (COCOMO)
Courtesy of Black Duck Open Hub - The Open Source Network

www.wikisuite.org/Constructive-Cost-Model-COCOMO

Really? How is this calculated?

Yes, really: If you tried to re-code WikiSuite using proprietary models. Below are the reports for various WikiSuite components generated by Black Duck Open Hub. Here is one example: https://www.openhub.net/p/openfire/estimated_cost, and it has this note: "COCOMO is meant to include the design, specification drafting, reviewing and management overhead that goes along with producing quality software." More info:

- https://en.wikipedia.org/wiki/COCOMO

How can it be free and who paid for this?

Like Wikipedia, GNU/Linux and Firefox, WikiSuite is the result of massive collaboration. And you are warmly invited to join!

How do you arrive at "well over US$50 million"? Are

you including the cost of GNU/Linux?

No. Just the Linux Kernel would cost billions to redevelop.

This is intentionally as small as possible since we focus on upstreaming code to the various Software Components.

The main software components:

Tiki Wiki CMS Groupware

- Well over 1 million downloads
- 345 code committers
- Over 77,000 commits
- Over 52,000 messages on the developer's mailing list
- Tiki is the Free/Libre/Open Source Web Application with the most built-in features

And this includes the Database Web Apps builder (Tiki Trackers)

Tiki dependencies

- Plus 150 external dependencies, managed by Composer and https://dev.tiki.org/The-Tiki-27-plus-Build-System

Virtualmin / Webmin

Manticore Search

Openfire

- Over 8,500 commits

Jitsi Meet

MeshCentral

Syncthing

Over 9,500 commits

Optional components

FusionPBX

FreeSWITCH

Xibo

Isn't the figure exaggerated?

In some cases, it's actually underestimated. COCOMO counts the number of lines of code. But a project like Tiki Wiki CMS Groupware has been in active development for over 20 years, and there have been over 82,000 code commits. This battle-tested code costs more to develop than a younger application that would have the same number of lines of code.

As of December 31st, 2020, Tiki's cost of development is estimated at USD$16,701,644, and the latest revision is #77641, which represents about $215 per commit. Given the work that goes into software development, and since "COCOMO is meant to include the design, specification drafting, reviewing and management overhead that goes along with producing quality software.", the cost is clearly not overestimated. And that doesn't even count the 150+ external dependencies. Think about it. Each dependency represents a distinct community / code base / bug tracker / etc.

Just the direct Tiki dependencies are estimated at over US$100 million. This is code the Tiki community uses but rarely participates to.

Ok, but is this for plugins / duplicated functionality?

No. There is minimal feature overlap / code duplication in WikiSuite because our philosophy is to avoid it as per our component criteria model. While it's true that some of WikiSuite's components have apps or plugins (Openfire), it's mostly a deployment mechanism, and there is no mess (community fragmentation, etc.) as described at: http://pluginproblems.com/

Given this huge value, isn't it a high cost to maintain

the apps?

Many of the software components are 10 to 15 years old, so they are quite mature. Maintaining a mature app, even with it undergoing occasional revamps, is a tiny fraction of the effort of rewriting it. And now, each of the applications has a community / ecosystem that uses it and is highly motivated to maintain and develop it.

If another suite has more lines of code (and thus a higher dollar amount), does it make it better?

No. The goal is to have all desired features with minimal complexity, and a sufficient community to sustain it. Some other suites are focused on facilitating self-hosting of many apps, including apps with identical functionality, but WikiSuite is very opinionated about the apps that compose it. We believe supporting more than one application for the same need adds unnecessary complexity and fragmentation. We believe WikiSuite's model is the best way to have the most features, with the least complexity. Please see: http://pluginproblems.com/

Ok, but I am not going to use all the features

That is true of all feature-rich software. But the economics of software development don't make it feasible to build software exclusively tailored for one use case. Community Free / Libre / Open Source software will necessarily want to cater to a wider community and get more users and contributors.

But adding up these figures is way more than US$50 million

Indeed. The total of the figures in the screenshot above is US$78 million. But then

- Someone could argue that such or such component is over-evaluated.
- One of the projects could reduce the number of lines of code (ex.: after a refactor) and then, the figure would go down.
- Someone could argue that the COCOMO formula is inappropriate.
- In some projects, the dependencies could have been counted, which was not done in the case of Tiki Wiki CMS Groupware. This would add over US$100 million!
- Someone could say (for example) that FusionPBX is not well integrated in WikiSuite, and thus, it should not be counted.
- Someone could say (for example) that FreeSWITCH is so popular (and integrated in many things) that it should be considered as an infrastructure component like GNU/Linux and not part of WikiSuite.
- etc.

So if we go with US$78 million, the figure can be fairly challenged as being exaggerated. And then, the narrative shifts to this topic (are we being honest with the numbers). This would be a distraction. We will never know the precise "cost to develop". It's an approximation to provide information on the scale/magnitude. Thus, the chosen strategy is to claim that WikiSuite's cost to develop is "well over US$50 million", which is really

hard to discredit. And most humans are not that good with big numbers: US$50 million is just as impressive as US$78 million to most humans 😀

Conclusions

Even if we exclude dependencies (which is debatable), and not count all the components, we can see that the cost to develop is "well over US$50 million". This demonstrates the following:

- Software development is expensive.
- Only organizations with very deep pockets can think of entering the enterprise suite space in a proprietary software development model. Ex.: WikiSuite vs Google, WikiSuite vs Microsoft.
- WikiSuite components are feature-rich as per component criteria.
- Free / Libre / Open Source development models are fantastic, even if they are not always well understood. Please see this analysis from 2009: https://www.computer.org/csdl/mags/so/2009/06/mso2009060004.pdf.
- Developing a full enterprise suite as Free / Libre / Open Source software is a huge amount of work which is why there are a limited number of alternatives.