

Projektgruppe Interoperabilität, Standards, Freie Software

Expertengespräch zum Thema „Freie Software“

Schriftliche Stellungnahme von Dr. Johannes Loxen

Sie sprechen von den Marktrealitäten, denen sich OSS stellen muss. Welche Herausforderungen ergeben sich hier für Entwickler/ Hersteller/Beteiligte von OSS und ist ein Zwang bei Ausschreibungen für OSS und freie Software wirklich vorteilhaft? Gibt es hier aus Ihrer Sicht gesetzgeberischen Handlungsbedarf oder regeln es die Marktteilnehmer selbst?

Die Marktrealitäten für OSS sind inzwischen die gleichen wie für alle im Software-Markt. In der ersten OSS-Zeit konnten Entwickler/Hersteller für sich eine Sonderstellung reklamieren, da „Community“ in der öffentlichen Wahrnehmung mit den Begriffen „freiwillig“, „selbstlos“, „altruistisch“ konnotiert wurde. Im Markt gelten diese Begriffe nicht und _wenn_ OSS im Markt erfolgreich sein will, dann muss gelten: „wettbewerbsfähig“, „leistungsfähig“ und auch „durchsetzungsstark“.

Die größten Herausforderungen liegen aus zwei Gründen im juristischen Bereich:

- Zum einen sind Verträge für OSS etwas anders gelagert als für Closed Source. Allgemeines Vertragsrecht, Haftung, Gewährleistung, Export-Beschränkungen und vieles mehr können nicht eins zu eins übernommen werden. Dies ist für Anbieter und Abnehmer eine zwingend notwendig zu leistende Anpassungs-Arbeit. Idealerweise bleibt der Anbieter so nahe wie möglich an bestehenden Regeln. So kann z.B. für die Verträge der öffentliche Hand nach EVB-IT vieles direkt übernommen werden - aber leider nicht alles.
- Zum anderen sind OSS-Lizenzen in erster Linie Quellcode-Lizenzen und _keine_ Überlassungslizenzen im Sinne der „Volumenlizenzen“ oder „Einzelplatzlizenzen“ aus der Closed-Welt. SUSE hat zum Beispiel für Server-Lizenzen verschiedene Preise je nach gewählter Virtualisierungstechnologie unabhängig von der GPL-Lizenzierung des Quellcodes. OSS-Quell-Lizenzen sind also etwas zusätzlich Anderes/Neues, weil neben der auf OSS anzupassenden Vertragslogik für „Überlassung“ noch die Vertragslogik für den Quellcode hinzukommt, die es in der Closed-Software-Welt nicht gibt.

Dieser oft nicht gesehene duale Komplex ist wichtig für die öffentliche Vergabe. Die Frage des Zwangs zu OSS ist differenziert zu betrachten. Überall dort, wo Sicherheitsbelange eine Rolle spielen, sollte dem Beispiel der Industrie (z.B. Airbus) gefolgt werden, die Offenlegung von Programmen zu fordern - was am Besten bei OSS-Lizenzen gelingt. Weiterhin ist OSS immer dort im Vorteil, wo die Software breit eingesetzt wird (z.B. Linux, Libre-Office) und damit Skalierungseffekte bei individueller und frei übertragbarer Anpassung erzielt werden

können. Software dieser Art ist meistens auch preiswert in der Beschaffung und aufwändiger in der Anpassung, so dass Synergie-Effekte bei Anpassungen wirksamer greifen können.

Wenig Sinn macht ein Zwang zu OSS bei Spezial-Anwendungen und kleinen Märkten. So ist z.B. der Markt für CRM-Software enger. Das Beispiel von SugarCRM als OSS zeigt, dass Single-Vendor-OSS sogar nachteilig ist, wenn man als freie Variante eine nur sehr eingeschränkt verwendbare Rumpf-Software bekommt und die volle Funktionalität genauso kaufen muss wie Closed-Source. An diesem Beispiel ist der Unterschied zwischen Quellcode-Lizenzierung und Überlassungs-Lizenzierung mit allen Konsequenzen schön sichtbar. Bei Ausschreibungen mit Zwang zu OSS würde man also Single-Vendor-OSS oder Dual-License-OSS ausschliessen, was gerade in kleinen Märkten keinen Sinn macht - also den Zwang zu OSS verbietet.

Gesetzgebung ist für diese Marktzusammenhänge viel zu träge und kein geeignetes Werkzeug mit Ausnahme des Bereichs der Software mit Sicherheitsbezug. Beim Skandal um den Staatstrojaner wäre es sinnvoll gewesen, wenn dessen Software-Quellen zum Beispiel dem BSI von Anfang an vorgelegen hätten.

Gibt es Szenarien, in denen proprietäre einerseits oder offene Software andererseits für den Anwender aus ihrer Natur heraus eindeutig die bessere Wahl darstellen? Oder haben beide stets Vor- und Nachteile?

Auch offene Software kann sehr proprietär sein, wie die Beispiele von SugarCRM oder MySQL zeigen. Single-Vendor-OSS zeigt auch die gleichen Effekte im Markt wie Closed-Source-Software und reduziert sich im Extremfall auf Software-Escrow - also eine Art öffentliche Quellcode-Hinterlegung.

Besser betrachtet man es so: In den meisten Fällen hat die Veröffentlichung eines Quellcodes keinen Nachteil für den Hersteller. Denn entweder ist die Aufgabenstellung komplex genug, so dass auch der Quellcode komplex ist und nicht so einfach übernommen werden kann ohne einen aufwändigen KnowHow-Erwerb - oder die Aufgabenstellung der Software ist simpel, so dass es auch ohne Quellcode-Veröffentlichung Nachahmerprodukte schnell im Markt gibt.

Für den Anwender einer Software sind ganz andere Kriterien relevant, als ob der Hersteller irgendwo den Quellcode mitliefert oder nicht, nämlich: Support und Preis. Kaum ein professioneller Anwender einer OSS wird selber an der Software Änderungen vornehmen, sondern immer die Autoren fragen, damit Änderungen „upstream“ gehen, also auch in Zukunft in der Software vorhanden sind.

Es gibt also stets Vor- und Nachteile aus der Quellcode-Lizenzierung – noch mehr jedoch aus der Leistungsfähigkeit des Programms als solches.