
Governance von verteilten Innovationsprozessen mit Open Source Communities

Heidemarie Hanekop & Patrick Feuerstein
SOFI Göttingen

Abschlusskonferenz COLLIN „Kollaborative Innovationen“

Göttingen, 26.02.2016

SOFI Soziologisches Forschungsinstitut
(SOFI) an der Universität Göttingen

Ausgangsfrage:

Sind Open Source Communities ein geeignetes Modell um in verteilten Innovationsprozessen das vielfach über Organisationsgrenzen hinweg verstreute Wissen zu kombinieren und zu integrieren?

- OSS: gemeinschaftlicher Handlungskoordination mit Orientierungen, die das Teilen von Wissen und gemeinsame Praxis der Entwicklung fördern?
- Modell auch für Unternehmen? Chancen und Risiken
- Implikationen für Governance verteilter Innovationen?

Fallanalyse eines OSS Projektes mit Unternehmen

Open Source Projekte als Innovationsstrategie von Unternehmen in der IT-Branche

- OSS Bewegung in den 90ern peer-to-peer basierte Alternative zu proprietärer SW von Unternehmen
- seit 2000 zunehmendes Engagement von Unternehmen in Open Source Projekten (Schrape 2015)
- strategische Vorteile durch Interoperabilität, Netzeffekte, offene Systeme, insges. für neue dynamische Märkte, Infrastrukturfelder (Linux, Open Stack)
- zahlreiche Literatur, aber wenig zum konkretes Verhältnis zwischen Unternehmen und Communities im Entwicklungsprozess

Begriffe: Wissen & Gemeinschaft

Wissensbegriff:

- soziale Konstruktion von Wissen aus unterschiedlichen Kontexten
- Bedeutung von implizitem Wissen, das in beruflichen, situativen, sozialen Hintergründen verankert ist, auch praktisches Wissen
- Wissensaustausch über Organisationsgrenzen erfordert neben Zugang zu explizitem Wissen auch Austausch von Erfahrung, geteilte Kontexte, gemeinsame Praxis

Gemeinschaft als Governanceform: Merkmale

- kollektive Identität als Koordinationsmechanismus
- Handlungsorientierung: Zugehörigkeit, geteilte Ziele, solidarisches Verhalten, Reziprozität
- Institutionalisierung und soziale Strukturierung (Organisation)

(Streeck/Schmitter, Hollingsworth, Gläser, Ostrom, Wiesental)

Merkmale Open Source Community

- Ziele, kollektiver Akteur und Identität als Produktionsgemeinschaft
- Institutionalisierung von Normen und Regeln, insbes. für Entwicklungsprozess, technisch unterstützt (Github)
- selbstorganisierte Beiträge, kollektive Entscheidungen über OSS
- meritokratische soziale Struktur

O'Mahony, Lakhani, West, v. Krogh, Dolata/Schrape, Gläser, Dobusch/Quack

Ergebnisse der Fallanalyse Open Source Community

Typ: firm-sponsored, community-managed

- Start Mitte 90er, Interoperabilität Windows-Linux Netze, etablierte OSS im Linux-Umfeld, strategische Allianz mit Unternehmen gegen Microsoft
- Community: professionelle IT-ler, mehrheitlich aus Unternehmen
- Hauptentwickler je 25-40 (ca.100); ges. 364 Entwickler seit 2008
- involvierte Unternehmen: 10-20, wechselnd, teilw. IT-Konzerne; beteiligte Unternehmen durch Mitarbeiter: 6-8

Empirie

9 intensive Interviews (2-3 Std.), teilweise wiederholte Gespräche;
Teilnahme an 3-tägiger Entwicklerkonferenz (ca. 100 Teilnehmer)
Auswertung von Webseite, Archiv & Aufnahmen früherer Konferenzen

Ergebnis: typisch gemeinschaftliche Koordination

1. Kollektive Identität als Produktionsgemeinschaft

2. Kollektiver Entwicklungsprozesses

- selbstorganisierte Beiträge der Entwickler (keine Aufträge),
eingebracht als Bestandteil der OSS & Vorschlag,
kollektive Entscheidung über Aufnahme von Beiträgen
(ex post, peer review, transparent)
- informelle Teamstrukturen basieren auf gemeinsamer Praxis,
Verantwortung für bestimmten Code, Spezialwissen -Erfahrung,
pers. Beziehung, Reziprozität, Sympathie

3. Institutionalisierung

- Autorenschaft“ als Institution, technisch unterstützt: Grundlage für (Selbst-)Verantwortlichkeit, Wissensaustausch, meritokratische Position in Entscheidungsprozess
- Solidarisches Verhalten, Fragen beantworten (auch zum Code), Teilen von Wissen und Erfahrung
- Regeln, Konventionen für guten Code (auch Diskussion)
- Konsensprinzip bei Entscheidungen: Austausch von Argumenten bis zu Konsens oder Kompromiss; Abstimmung als letztes Mittel

4. Dauerhafte meritokratische Strukturen

- Mitgliedschaft & meritokratische Position durch Beiträge zur OSS
- informelle Teamstrukturen
- offen für Neue

Strategische Beteiligung von Unternehmen

Empirie: zwei Unternehmen mit den meisten Mitarbeitern in der Community, deren Wertschöpfungsstrategie an „MSB“ hängt (typisch)

1. großer Linux-Distributor, „MSB“ in eig. Produkt integriert, aktuell in Kooperation mit MSB eigene Innovation, 6 Hauptentwickler
2. Weltweit Dienstleister für „MSB“, Entwicklung & Support, Aufträge von Anwendern, Alleinstellung d. Wissen & Einfluss, 6 Hauptentw.

Form der Beteiligung

- durch Mitarbeit (führender) UN-MitarbeiterInnen als Hauptentwickler in OSS Community (Sponsoring)
- im Unternehmen entwickelte Software wird vom Entwickler als Beitrag eingebracht (upstream)
- gleichzeitige Verwertung für betriebliche Wertschöpfung

Doppelrolle der Entwickler

- zwei unterschiedliche Handlungsorientierungen
- im Unternehmen: Arbeitsverhältnis, Aufträge, betriebl. Planung
- in Community: Entwicklungsziele, Selbstverpflichtung, soziale Erwartungen
- Zeitdisposition: betriebliche Aufgaben gehen vor, aber Spielräume für Aufgaben in der Community; im Zweifel in der Freizeit?

Unternehmen: Steuerung als Balance-Akt

- Umgang mit Mitarbeitern, die Hauptentwickler in der Community sind, *“man kann sie nicht zu 100% fürs UN einsetzen“*
- Interessenausgleich mit Community kann nicht primär einzelwirtschaftlicher Logik folgen: *„Multi-stakeholder-Konstellation“*

Unternehmen: Steuerungsprobleme

- Einfluss nur vermittelt über Mitarbeiter als Hauptentwickler in der Community
 - Personalressourcen einsetzen, Beiträge für OSS im UN
 - nicht sicher, ob die OSS-Community folgt
- Unmöglichkeit der zeitlichen Planung und Steuerung des Innovationsprozesses in der Community
 - Offen, ob und wann funktionsfähige Features in OSS
 - Kein direkter Einfluss auf Entscheidungen in der Community
 - Keine Planungsentscheidungen in Community (Bazarmodell)
- Was tun bei Entwicklungsblockade oder drohendem Zerfall der Community? Beispiel Krisenfall

Strategien des Umgangs mit Steuerungsproblemen

Personalpolitik

- Einstellung von Hauptentwicklern
- Auftrag an Mitarbeiter zur Mitarbeit in Community

Beteiligung an den Meinungsbildungsprozessen in der Community

- Beteiligung der „eigenen“ Hauptentwickler an Diskussionen
- Organisation von Konferenzen und Meetings, Workshops
- Vorschläge für Entwicklungsperspektiven und Werbung um Mitarbeit („Rallying Calls“)

Alleingang als Fallback-Lösung:

- wichtige Entwicklungen in Eigenregie mit anschl. Versuch sie in OSS zu integrieren
- Risiken, Kosten, Grenzen eines Alleingangs

Fazit: Mix aus gemeinschaftlicher & betrieblicher Governance

Wie? mit welchen Implikationen für die Akteure?

Open Source Community - intern gemeinschaftlich

- Austausch und Integration von Wissen und Entwicklern
- Koordination der verteilten Innovation

aber: keine Verfügung über Ressourcen,

- Risiken für nachhaltige Entwicklung der OSS, Abhängigkeit von UN

Unternehmen - intern hierarchisch

- Personalressourcen & Beiträge für OSS Entwicklung
- betriebliche Wertschöpfung mit der OSS

aber: Steuerungsprobleme bezogen auf die verteilte Innovation

- Abhängigkeit von OSS

Schnittmenge in Bezug auf die Entwickler, die in beiden Feldern agieren, ihr Wissen und ihre Arbeit in beide Felder einbringen (sollen)