

SUMMARY

The central question driving this PhD research is whether modularity, by mitigating the need for active coordination between distinct components, increases the potential number of contributors to a free/open source software (FOSS) project and has a positive effect on their labour productivity, allowing them to work independently of each other.

Chapter 1 situates the emergence of the design principle of modularity as a response mechanism to the organisational problem of decreasing returns to scale.

Chapter 2 reviews the literature on modularity as a design principle for complex product development, drawing the following hypotheses for subsequent empirical testing:

- *Product modularity reduces coordination costs in FOSS projects (H1)*
- *Product modularity increases the potential number of contributors to a FOSS project (H2)*
- *An increase of contributors to a FOSS project results in an increase of modularity (H2R)*
- *Product modularity has a positive effect on labour productivity in FOSS projects (H3)*
- *An increase of contributors to a FOSS project has a negative effect on labour productivity (H4)*

Chapter 3 describes the research methodology: it explains our indicators of modularity, coordination costs, group size and labour productivity and the manner in which we use panel data (a.k.a. longitudinal or time-series data) collected from FreeBSD's software repositories to put the hypotheses to the test. In specific, our analysis takes place on two levels: we examine the relationship between modularity, coordination costs, group size and productivity (a) at the project-level (that is, for the FreeBSD project as a whole) through a qualitative analysis of descriptive statistics and (b) at the component-level (i.e. at the level of the individual modules making up the FreeBSD operating system) through a quantitative analysis of a dataset that

includes thirty modules selected through stratified random sampling: modules were categorised into three strata based on their scale, as reflected in the number of developers contributing to them ($N=387$, $H=3$), and ten modules were randomly selected from each stratum. The statistical instrument used for the quantitative analysis is random-effects GLS regression.

Chapter 4 introduces the empirical setting – the FreeBSD project – and discusses its historical and organisational background.

Chapter 5 examines the extent to which modularity reduces coordination costs in FreeBSD ($H1$) but finds no empirical support for the hypothesis that higher levels of modularity correlate with lower levels of coordination costs.

The first part of chapter 6 examines whether modularity increases the potential number of contributors to FreeBSD ($H2$) and provides strong empirical support to the hypothesis. The second part of chapter 6 tests $H2R$, which reverses the directionality of the effect so that increasing group size is claimed to result in an increase of modularity. The statistical tests we performed verify the hypothesised effect, provided that conditions of large-scale development (i.e. committers > 8) apply.

Chapter 7 examines the effect of modularity on group performance and finds that – to the extent that conditions of large-scale development prevail – modularity has a positive effect on both average group performance and core developers' performance.

Chapter 8 examines the effect of increasing group size on labour productivity. Our analysis of descriptive statistics shows that the historical expansion of the FreeBSD committers' group brought about a fall in average group productivity, seemingly confirming $H4$, but it also resulted in a rise in core developers' output. This finding is qualified by arguing that large groups enable a more extensive division of labour (on a voluntary basis, of course) within the modules they develop, thanks to which core developers can focus on their task of choice, namely new code development, thereby suggesting that the fall in group productivity is not caused by a fall in core developers' performance, but by the disproportionate increase of 'lower-contribution' committers over time. In the light of these results, $H4$ cannot be wholly accepted, as the causal mechanism underlying the decrease of average productivity differs markedly from that which $H4$ postulates (i.e. that the fall in group productivity is due equally to the low performance of new members and the fall in core developers' performance that is caused by

the communication and coordination costs attendant upon increasing group size).

Chapter 9 examines the transformation of governance to which FreeBSD resorted in order to accommodate itself to expanding scale. Catalysed by the growing criticism of the distribution of authority in the project, the adoption of the elective principle for the selection of the FreeBSD administrative team brought about a shift in the conception of leadership from the informal rule of a self-selected group of veteran developers to the democratic authority of an elected group that is revocable and bound to formal rules. Since, FreeBSD has evolved a collectivist governance system, based on a direct-democratic, consensus-oriented process of decision making. Furthermore, in keeping with the normative standard of individual autonomy of action, FreeBSD did not attempt to manage increased scale by supervising developers' work process but rather tried to achieve coordination through the standardisation of the induction process for new developers and of outputs through frequent building. Interestingly, the transformation of FreeBSD's governance structure contrasts sharply with how other large FOSS projects have attempted to manage increased scale. Characteristically, to facilitate coordination in an expanding group of developers, the Linux project introduced an additional layer of managerial hierarchy, as Linus Torvalds, the project leader, delegated authority to a cadre of subsystem maintainers – the so-called 'trusted lieutenants' – to filter the contributions of the wider base of Linux developers (Corbet et al. 2010, pp. 15-17; Moody 2001). Such a hierarchical response to increased scale points not only to the presence of important differences in the distribution of authority between FOSS projects but also to a strong element of trial-and-error experimentation with varying degrees of control over the process of integrating changes in the project code repository (Holck & Jørgensen 2004; Weber 2004).

Chapter 10 sums up the empirical findings and reflects on the role of modularity as a governance mechanism. As regards the effect of product structure on group dynamics:

- (a) Modularity makes decentralisation scalable by mitigating the need for active coordination between distinct modules.
- (b) Modularity reinforces the emergent division of labour: enlarging the scale of the project militates in favour of committers' specialisation

(because of the learning costs involved in familiarising oneself with the codebase) to which modularity conduces by enabling the independent development of distinct product components.

- (c) Modularity has a positive effect on average group productivity in large-scale conditions, for it allows developers to work independently of each other. But in small-scale conditions (i.e. committers < 9), its effect is insignificant: this implies that the potential of modularity can be fully exploited only in settings in which the need to mitigate the adverse effects of increasing scale takes on a pressing character.

With respect to the effect of group dynamics on product structure:

- (a) Product structure mirrors organisational structure: the distributed character of the development process of large FOSS projects such as FreeBSD implies that the scope for face-to-face communications is drastically narrowed. Because of the inherent limitations on communication, therefore, the product architecture that evolves is more modular.
- (b) Product structure constitutes a coordination mechanism. As FreeBSD (and FOSS projects in general) is devoid of an authority structure by which to effect coordination, FOSS developers are induced to use software structure as a variable that can be fine-tuned to reduce the need for active coordination between product components.
- (c) However, as long as the overall group of developers working on a module remains small (i.e. it does not exceed eight developers), adding more developers to the group prompts no changes in the direction of increased modularity. The product structure that evolves is then non-modular because it reflects the work patterns of a tightly-coupled group.

With respect to the effect of group size on labour productivity:

- (a) The high performance of core developers is not due to the absence of coordination costs but to the temporally increased scope of their participation.

- (b) Large groups enable a more extensive division of labour (on a voluntary basis) within the modules they develop, thanks to which core developers can focus on their task of choice, namely new code development. Hence, increasing group size results in boosting core developers' performance.
- (c) Crucially, the potential for decreasing returns to scale is blunted by the motivational forces at work: when one is working on tasks perceived as meaningful and engaging as well as in groups one highly values, then adding more persons to the group no longer has a demotivating effect. The ability to evaluate (a) individual committers' contributions to the collective outcome and (b) the performance of the committers' group as a whole against other FOSS projects' group performance also reinforces the motive to contribute. Furthermore, the pattern of scale expansion encountered in FreeBSD has been equally important in averting the manifestation of decreasing returns to scale: as the expansion of scale in FreeBSD was not accompanied by a taller hierarchy, it did not result in the communication distortions commonly besetting hierarchical structures.

In the *Epilogue*, the findings of our study are collated against a long tradition in the social sciences which holds that an increase of scale in numbers undermines a group's ability to self-organise and self-govern. We challenge this theory by arguing that, on the one hand, the conditions governing the emergence and development of hierarchy in a group are not independent of the values held by its members and, on the other, that the characteristics of the distributed environment in which FOSS projects operate render it incompatible with the exercise of coercive authority.

SAMENVATTING

De centrale vraag die ten grondslag ligt aan dit promotieonderzoek is of modulariteit – door het verminderen van een behoefte aan actieve coördinatie tussen gescheiden software modules – het potentiële aantal ontwikkelaars die hun bijdrage leveren aan een FOSS project vergroot en hun arbeidsproductiviteit verhoogt doordat zij onafhankelijk van elkaar kunnen werken.

Hoofdstuk 1 schetst een beeld van de opkomst van het concept modulariteit als ontwerpprincipie, in reactie op het organisationele vraagstuk van afnemende productiviteit bij toenemende schaalgrootte.

Hoofdstuk 2 geeft een overzicht van de literatuur over modulariteit als ontwerpprincipie voor de ontwikkeling van complexe producten. Op basis van deze literatuur zijn de volgende hypothesen opgesteld, welke worden getoetst in het empirisch deel van dit onderzoek:

- *Productmodulariteit leidt tot een afname in coördinatie kosten in FOSS projects (H1)*
- *Productmodulariteit leidt tot een toename in het potentiële aantal ontwikkelaars die bijdragen aan een FOSS project (H2)*
- *Toename in het aantal ontwikkelaars die bijdragen aan een FOSS project resulteert in een toename van de productmodulariteit (H2R)*
- *Productmodulariteit heeft een positief effect op arbeidsproductiviteit in FOSS projects (H3)*
- *Toename in het aantal ontwikkelaars die bijdragen aan een FOSS project heeft een negatief effect op arbeidsproductiviteit (H4)*

Hoofdstuk 3 beschrijft de onderzoeksmethodologie: het geeft uitleg over de indicatoren van modulariteit, coördinatie kosten, groeps grootte en arbeidsproductiviteit, die in dit onderzoek worden gebruikt. Hiernaast beschrijft dit hoofdstuk de wijze waarop panel data (i.e. longitudinale of time-series data afkomstig van de FreeBSD repositories) wordt gebruikt om de onderzoekshypothesen te testen. In de onderzoeksanalyse onderscheiden wij twee niveaus: We analyseren de relatie tussen modulariteit, coördinatie

kosten, groepsgrootte en productiviteit (a) op project-niveau (dwz voor het gehele FreeBSD project) door middel van een kwalitatieve analyse van beschrijvende statistiek en (b) op de module-niveau door een kwantitatieve analyse van een gestratificeerde a-selecte steekproef van 30 modules uit het FreeBSD project. Op basis van het aantal ontwikkelaars dat een bijdrage levert zijn de modules ingedeeld in drie categorieën van schaalgrootte (klein, midden, groot). Voor dit onderzoek zijn uit elk van deze drie categorieën, op a-selecte wijze, tien modules geïncludeerd in de steekproef. Voor de kwantitatieve analyse van de steekproef data is gebruikt gemaakt van random-effects GLS regressie-analyse.

Hoofdstuk 4 gaat in op de historische en organisatorische achtergrond van het FreeBSD project waarop het empirisch deel van dit onderzoek op is gericht.

Hoofdstuk 5 gaat in op de vraag in hoeverre modulariteit de coördinatie kosten vermindert (H1). Dit hoofdstuk concludeert dat deze hypothese niet kan worden geverifieerd.

Het eerste deel van hoofdstuk 6 richt zich op de vraag of sterkere mate van modulariteit leidt tot een toename in het aantal ontwikkelaars dat een bijdrage levert aan een FOSS project (H2). Dit hoofdstuk concludeert dat de empirische bevindingen deze hypothese ondersteunen. Het tweede deel van hoofdstuk 6 beantwoordt de omgekeerde vraag, namelijk of een toename in de groepsgrootte van de ontwikkelaars resulteert in een toenemende mate van modulariteit (H2R). Dit veronderstelde effect wordt bevestigd door de statische analyses, maar dit geldt alleen voor modules die zijn gecategoriseerd als modules met een grote schaalgrootte (i.e. met meer dan acht ontwikkelaars).

Hoofdstuk 7 gaat in op het effect van modulariteit op de productiviteit van de groep, en vindt dat – zolang de voorwaarden voor grote schaal ontwikkeling van kracht zijn – modulariteit een positief effect heeft op zowel de gemiddelde productiviteit als op de productiviteit van core-ontwikkelaars.

Hoofdstuk 8 richt zich op het effect van toename van het aantal betrokken ontwikkelaars op arbeidsproductiviteit. Onze analyse toont aan dat de groei van het FreeBSD groep over de jaren heeft geleid tot een afname in gemiddelde arbeidsproductiviteit. Op het eerste gezicht lijkt de analyse hiermee H4 te bevestigen. Echter de toename in het aantal betrokken ontwikkelaars is ook gepaard gegaan met een toenemende

productiviteit van de core-ontwikkelaars. Deze bevinding kan worden worden verklaard door het feit dat grotere groepen van ontwikkelaars het mogelijk maken om werkzaamheden binnen een module beter te verdelen. Hierdoor kunnen core-ontwikkelaars zich meer toeleggen op hun primaire interessegebied, namelijk het programmeren. De afname in productiviteit komt dus niet doordat de core-ontwikkelaars minder produceren, maar doordat het aantal minder productieve ontwikkelaars die betrokken zijn bij de module toeneemt. In het licht van deze bevindingen kan H4 daarom niet zonder meer worden aangenomen omdat de gevonden verklaring voor de afname van de gemiddelde productiviteit afwijkt van het standpunt dat is weergegeven in H4. Deze hypothese veronderstelt namelijk dat zowel nieuwe ontwikkelaars als de core-ontwikkelaars minder productief worden naarmate de groepsgrootte van het aantal betrokken ontwikkelaars en door toenemende coördinatie kosten.

Hoofdstuk 9 reflecteert op de wijze waarop FreeBSD zelf heeft gereageerd op de toenemende schaalvergroting door wijzigingen in aansturing en governance door te voeren. Onder invloed van groeiende kritiek op de verdeling en toekenning van autoriteit binnen het project werd een gekozen FreeBSD bestuursteam ingesteld. Deze transformatie heeft een verschuiving teweeggebracht van informeel leiderschap in handen van een selecte groep veteranen naar een democratischer vorm van leiderschap in handen van een gekozen bestuursteam dat is gebonden aan formele regels. Met deze verschuiving in leiderschap heeft FreeBSD een collectivistisch governance model ontwikkeld dat is gebaseerd op een direct-democratisch en consensus-georiënteerd besluitvormingsproces. Door de handhaving van de normatieve standaard van individuele autonomie heeft FreeBSD met de ontwikkeling van het nieuwe governance model niet geprobeerd om op schaalvergroting te sturen door over de schouders van ontwikkelaars mee te kijken. In plaats van hiervan is geprobeerd de coördinatie te verbeteren door het standardiseren van het inductie proces van nieuwe ontwikkelaars en door het standardiseren van de output, door regelmatig te compileren. Interessant is dat de transformatie van de FreeBSD governance structuur in scherp contrast staat met de wijze waarop andere grote FOSS projecten invulling hebben gegeven aan sturing op (effecten van) schaalvergroting. Bijvoorbeeld, het Linux project een hiërarchische managementlaag geïntroduceerd om coördinatie binnen een groeiende groep betrokken ontwikkelaars te faciliteren. Hiertoe heeft Linus Torvalds, de leider van het

Linux project, autoriteit toegekend aan een kader van subsysteem-beheerders – de zogenoemde 'trusted lieutenants'- om de bijdragen van de ontwikkelaars te filteren (Corbet et al. 2010, pp. 15-17; Moody 2001). Deze hiërarchische benadering van sturing op schaalvergroting wijst niet alleen op significante verschillen in de distributie van autoriteit tussen FOSS projecten, maar ook op trial-and-error experimenteren met varierende gradaties van controle over het proces van het integreren van wijzingen in de project code repository (Holck & Jørgensen 2004; Weber 2004).

Hoofdstuk 10 vat de empirische bevindingen van dit onderzoek samen en reflecteert op de rol van modulariteit binnen governance structuren. De belangrijkste bevindingen met betrekking tot de effecten van productstructuur op groepsdynamiek zijn de volgende:

- (a) Modulariteit maakt decentralisatie schaalbaar doordat de behoefte aan coordinatie tussen gescheiden modules afneemt.
- (b) Modulariteit versterkt een emergente arbeidsverdeling: toenemende schaalgrootte van projecten werkt specialisatie van ontwikkelaars in de hand, door de leerkosten die het doorgronden van de codebase met zich meebrengen. Modulariteit stimuleert specialisatie door onafhankelijke ontwikkeling van gescheiden product componenten.
- (c) Modulariteit heeft ook een positief effect op de gemiddelde productiviteit binnen projecten met grotere schaalgrootte (committers > 8) omdat het ontwikkelaars in staat stelt om onafhankelijk van elkaar te werken. Echter, binnen projecten met een kleinere schaalgrootte (committers < 9), is dit effect van modulariteit niet significant. Dit betekent dat modulariteit alleen ten volle kan worden benut in een context waar de behoefte aan het afzwakken van negatieve effecten van schaalvergroting urgent is.

De belangrijkste bevindingen met betrekking tot de effecten van groepsdynamiek op productstructuur:

- (a) De productstructuur weerspiegelt de organisatiestructuur: het gedecentraliseerde karakter van het ontwikkelproces in grote FOSS projecten – zoals FreeBSD – impliceert dat mogelijkheden voor face-to-face communicatie drastisch afnemen. Deze inherente beperkingen voor communicatie zorgen ervoor dat de product

structuur een sterker modulair karakter ontwikkeld.

- (b) De productstructuur vormt een coordinatiemechanisme. Omdat FreeBSD (en FOSS projecten in het algemeen) geen coördinerende autoriteit kennen, gebruiken FOSS developers de productstructuur als een variable die gefinetuned kan worden, om een behoefte aan coördinatie tussen de product componenten tot een minimum te beperken.
- (c) Echter, zolang als de totale groep ontwikkelaars die betrokken zijn bij een module klein blijft (committers < 9), leidt het toevoegen van meer ontwikkelaars aan de groep niet voor meer modulariteit. De productstructuur die eruit voortkomt is dan niet modulair omdat het de manier van werken van een hechte groep weerspiegelt.

De belangrijkste bevindingen met betrekking tot de effecten van groeps grootte op arbeidsproductiviteit:

- (a) De hoge productiviteit van core-ontwikkelaars is niet toe te schrijven aan lage coördinatie kosten, maar eerder door een tijdsgebonden toename in hun participatie in het project.
- (b) Grote groepen betrokken ontwikkelaars bieden mogelijkheid tot een omvangrijker arbeidsverdeling (op vrijwillige basis) binnen de modules die ze ontwikkelen. Daardoor kunnen core-ontwikkelaars zich beter focussen op de dingen ze leuk vinden, namelijk programmeren. Daarom resulteert een toenemende groeps grootte in een toename van de productiviteit van de core-ontwikkelaars.
- (c) Door de motivatie van ontwikkelaars kunnen afnemende schaalopbrengsten worden opgeheven. Wanneer men werkt aan taken die als betekenisvol en boeiend worden ervaren binnen een groep waar men zich mee identificeert, dan heeft uitbreiding van deze groep niet langer een demotiverend effect. De zichtbaarheid van individuele bijdragen aan het collectieve resultaat en de herkenbaarheid van de groepsprestatie in relatie tot andere FOSS projecten vormen een prikkel om bij te dragen. Bovendien, heeft de wijze waarop de governance structuur van FreeBSD zich heeft ontwikkeld onder de omstandigheden van schaalvergroting een belangrijke rol gespeeld in het afwenden van de negatieve effecten van schaalvergroting. Omdat de toenemende schaal grootte van

FreeBSD niet gepaard ging met een toenemend hiërarchische organisatie van het project, kwamen de communicatie verstoringen die kenmerkend zijn hiërarchische organisatiestructuren niet voor.

In de Epiloog, worden de bevindingen van onze studie vergeleken met een lange traditie in de sociale wetenschappen welke voorstaat dat schaalvergroting het vermogen van een groep om zichzelf te organiseren en besturen ondermijnt. Wij weerleggen deze stellingname, enerzijds door te beargumenteren dat de condities die leiden tot de ontwikkeling van hiërarchische organisatiestructuren in groepen niet los gezien kunnen worden van de normen en waarden van de groepsleden. Anderzijds, beargumenteren wij dat de kenmerken van de gedecentraliseerde context waarbinnen FOSS projecten opereren onverenigbaar zijn met een dwingende en directieve invulling van autoriteit binnen hiërarchische organisatiestructuren.