

## Free and Open Source Software – An Empirical Study

HarmaninderJit Singh Sidhu<sup>1</sup>, Sawtantar Singh Khurmi<sup>2</sup>

<sup>1</sup>*Desh Bhagat University, MandiGobindgarh, Punjab, India.*

<sup>2</sup>*Yadavindra College of Engineering, Talwandi Sabo, Bathinda, Punjab, India.*

**Abstract** - A software can be described in terms of its characteristics. These characteristics can be further subdivided into sub-characteristics and their corresponding attributes. The basic idea generally remains the same i.e. to analyse the software in terms of its working, performance, cost and complexity. All these aspects lead to the software evaluation. Free and Open Source Software (FOSS) is no exception in this regard. The basic idea behind carrying out this research endeavour was to analyse FOSS in terms of its important characteristics like Deployability and Maintainability. The purpose of such study was to establish the association between these two vital characteristics empirically. Apart from this, two very important issues related to FOSS, were also taken into account. It was tried to find out how and what help FOSS professionals seek when they encounter some problem in their routine or specialized tasks. Another matter of concern was to know if people involved in FOSS do require some training sort of activity for the betterment of their FOSS related activities. The survey results signify that seeking additional knowledge through one way or the other does improve proficiency. The probable dependency of the two characteristics on each other was also established.

**Keywords** -Open Source Software (OSS)\Free and Open Source Software (FOSS),FOSS Empirical Study/Analysis, FOSS Deployability, FOSS Maintainability, FOSS Characteristics.

### I. INTRODUCTION

The very first characteristic of interest in the present study is Deployability. It refers to the capability of a software to be packaged for usage in a particular environment (Sanga, 2010). When a software is able to be transferred from one environment to another then it is termed as portable and the process is defined as Portability (Bertoa and Vallecillo, 2002). Therefore, the term Portability is integrated in the term Deployability. A number of software quality models include Deployability and various quality models (i.e. Boehm, IEEE, ISO 9126, MCcall) make available different attribute of Deployability characteristic (Seffahet *al.*, 2001). Portability, Installability, Configurability, Adapatability, and Distributability are the required attribute of Deployability. The second quality characteristic in the current study is Maintainability. Maintainability may be referred to as capability of the FOSS to be rectified, tailored and enhanced as per the varying requirements of the user and the desired functional stipulations. As per the definition given by Bertoa and Vallecillo (2002), Maintainability is the capability of software to be modified. Different models of software quality (like Boehm, IEEE, ISO 9126, and MCcall) make available different attribute of the characteristic, Maintainability (Seffahet *al.*, 2001). Sub-characteristics of the Maintainability as given by ISO (2001) are Stability, Analyzability, Changeability, Testability, Trackability, Flexibility and Upgradeability (Bertoa and Vallecillo, 2002).

#### 1.1 RESEARCH OBJECTIVES

The objectives of this empirical research can be outlined as follows:

- To determine the status of FOSS in terms of its existing attributes.
- To identify the association between the two prime characteristics of FOSS i.e. Deployability and Maintainability.
- To investigate some important aspects related to FOSS.

Through this research effort it was tried to know the present status of FOSS as per the experiences of the people involved in FOSS in one way or the other in the ongoing scenario as far as the software industry is concerned. The main objective was to describe FOSS as seen through the eyes of its users/advanced users.

## II. FREE AND OPEN SOURCE SOFTWARE

### 2.1 DEFINITION

Free and Open-source software is computer software that is provided by its copyright holder to anyone for any use with a license that entitles its user(s) to read/study, modify and distribute/redistribute the software to others for any purpose (Laurent and Andrew, 2008). Another category of software that we all know about is available in the market as closed source software. It is also called proprietary software that is available in the market at a price without the availability of any source code and with a license for restricted use of software. It is also worth mentioning here that FOSS may not be absolutely free (i.e. without any cost). It is because the FOSS may be accompanied by the proprietary components which may cost you and also services offered with FOSS may be charged for.

### 2.2 LICENSES

A FOSS license makes certain that a particular software is available freely for its users and that one can alter and redistribute it without any restriction. The Table 1 shows the three general categories of FOSS in which its licenses may be put collectively (Alspaugh *et al.*, 2012).

Table 1: Types of FOSS licenses

Type of License	Also known As	Examples
Permissive	Academic	Apache, BSD, MIT
Reciprocal	Copyleft	MPL, LGPL
Propagating	Strong Copyleft	GPL, AGPL

## III. RESEARCH METHODOLOGY

To carry out this empirical research data was collected from the participants existing over the vast swath in the region. Simple random sampling was used to make the sample an appropriate representative of the entire population as far as possible. A sample of two hundred respondents was chosen so as to make it a good fit for this research endeavour. The qualitative and quantitative methods were employed to gather and analyse the required data collected as a result of the survey to carry out this empirical research. Utmost care was taken to get the data with maximum possible inclination towards the reality. The qualitative methods were brought into practice to hold interviews with respondents to capture their views based on their experiences with FOSS and thus recording the observations of the participants. Quantitative methods appeared in the picture because the data was needed to be analysed by employing statistical means to check the credibility of the data and thus making interpretations based on the results obtained as a result of application of methods such as Cronbach's alpha and Fisher's exact test. Before carrying out the actual process of this empirical approach a pilot study was also planned and implemented to have a better insight of the prevailing circumstances in the vicinity of FOSS. Another motive of this pilot study was to consider the various attributes of FOSS agreed upon by the participants involved in this pilot construct.

## IV. RESULTS AND FINDINGS

### 4.1 SOURCES OF SUPPORT FOR FOSS

The proprietary software users take support from the proprietor itself (online/offline) whereas such support is not available for FOSS users. In the absence of such support, use and growth of FOSS may be in danger. But, it is also true that FOSS is not a myth in fact it is a reality that has proved its existence and usefulness. Therefore, some way-out must be there to support FOSS. The knowledge of this source (to support FOSS) is very important for the potential FOSS professionals. To answer this question, the authors inquired the FOSS professionals regarding their source of support for FOSS. The information given in Figure 1 highlights that such software systems are not devoid of support. The support is available in one way or the other. The need of the hour is to recognize the best possible option of getting help as per your requirements. A look at the data in Figure 1 reveals the fact that majority of the FOSS people are capable enough to handle such issues themselves. Although, at the first look it does not appear to be appealing but if we look at the present scenario we find that the people and time are changing at a very fast speed. Today, the common man has become more technology oriented. Moreover, especially the software users have become smarter because of internet and other electronic

media which are acting as chief source of knowledge for all kinds of information. Apart from this the people who are also concerned with other issues related to FOSS such as its development (developer users), are more likely to face lesser number of problems as far as the technicalities related to FOSS are concerned.

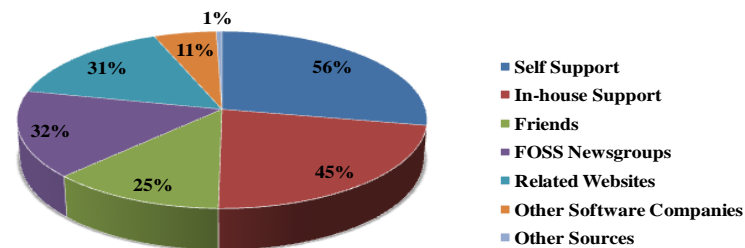


Figure 1: Various Sources of help and support regarding FOSS

Yet another potential source of help is there which provides support through in-house staff. The statistics (Figure 1) point out that it is the second most important source of support as substantial number of respondents agree that they rely on this medium of support. This finding does not need any further justification or clarification as it is quite obvious that when you get help from the in-house team it comes absolutely free provided if we do not take into account the time spent on troubleshooting the problem for which the help is called upon.

The other two main sources of service and support related to FOSS are: FOSS newsgroups and concerned websites. These platforms which hold a significant position (as shown in Figure 1) among the other sources of help and support are very important from the point of view of technological advancements in the country as far as the information and communication technology is concerned. It is apparent that these two options are equally popular among FOSS people for approaching help. The figures also confirm that a fairly good number of people are dependent on these options.

Getting help from friends also seems to be a dependable option (from the survey data) as the figures show that people also follow this practice. The other sources of help shown in Figure 1 appear to contribute feebly towards resolving the issues related to FOSS as they may involve certain expenses to a considerable amount which may not appear to be justifying from point of view of some people (certainly you may not be willing to spend directly or indirectly for something that is available in the market free of cost).

So, the survey reveals that FOSS users are smart enough that by applying their mind they become self-supportive. They also get support from in-house staff. FOSS newsgroups and relevant websites are also supportive. The wonderful outcome of this empirical survey also establishes that people spend little money on support that is they barely depend upon paid supply.

#### 4.2 TRAINING AND HELP REQUIRED

Although FOSS are freely available systems, yet training regarding them is available to users, paid or free. It is expected that a person who is trained would be more confident and his further online queries will be trifling. To answer this question, respondents were asked about their training regarding FOSS. Out of 200 respondents 25 percent responded positively regarding some training acquired. Rest of them (75 percent) didn't take any training (Figure 2). If we carefully examine the results of survey we find that the results are on expected lines and do not need any supportive base as they appear to be justifying themselves and it will become clear from the discussion coming up.

As, we have established in Figure 1 that people have become smarter than ever in meeting their problems successfully in the present times as far as their software issues are concerned (FOSS related). The data in Figure 2 is just an analogy to that. As we can clearly see that only one forth people have gone any sort of training in FOSS technologies and rest of them are on their own. Two possibilities can be there in that they didnot have any training. The first likely reason that seems to be there is that they were competent enough in this regard to handle any kind of issue. So, they did not opt for any training sort of assistance when they started their work in this field. The other reason that may be there is that the groups/companies for which they are workingwere fully confident about their professional competency of whatever task they were/are assigned. So,

they havenot undergone any sort of training till date to work on FOSS platforms.A quite natural finding can also bededuced from the statistics available with us (Figure 2) that as anyone would expect the people having greater

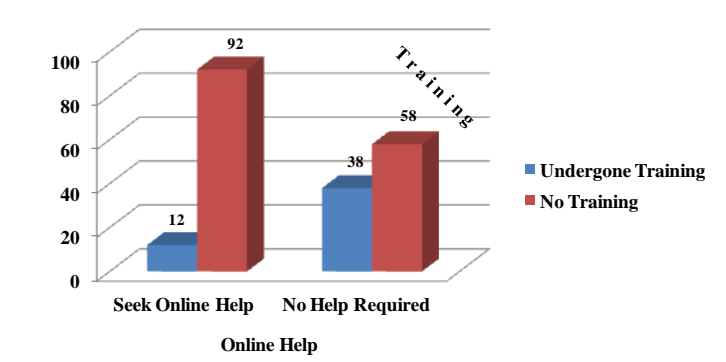


Figure 2: Users involved in Training and Queries

proficiency in their work/field (attained either through training or through any other means) are less likely to expect any sort of assistance from any probable source. This likely outcome from the obtained data is further strengthened by the respondents’ response because only 6 percent of the total number of respondents who have undergone any sort of FOSS training said that they occasionally need online assistance to sort out their FOSS related problems. Whereas one fifth of total number of respondents say that they have attained such level of expertise in their work through training that they seldom need any sort of support (online or offline).

The number of the respondents portrayed in Figure 2 who have not undergone any FOSS training admit that sometime they may need some sort of assistance to sort out some typical technicalities which are beyond the scope of their work profile. At this point they seek for online help/queries in a view to complete their ongoing task. This activity does take noticeable time but at the sametime, to complete the desired task it is very necessary also. The length of the response time and probability of response may vary according to the complexity of the query posed. On the contrary a sufficiently large number of respondents confirm that they manage to resolve the matters without any external advice. They try doing it themselves and remain successful most of the times.

Hence it can be concluded here that trained person need less further online help. Though, sometimes even those who got training are also involved in further online queries to resolve certain issues.

#### 4.3 RELIABILITY AND VALIDITY OF DATA

In order to ensure the reliability and validity of the data collected as a result of the undertaken survey to carry out this empirical research the data was subjected to Cronbach’s alpha test. This test is a measure of internal consistency of the data.

Table 2: Cronbach’s alpha for Deployability Data

Reliability Statistics		
Cronbach's Alpha	N of Items	VARIABLES
.981	5	Portability, Installability, Configurability, Adaptability, Distributability

The lower and upper bounds of the value of alpha are 0 and 1 respectively. The application of this method/test on the primary database is significant from the point of view of checking its suitability for hypothesis testing. The outcome of the application of Cronbach’s alpha on the primary data is depicted in Table

Table 3: Cronbach’s alpha for Maintainability Data

Reliability Statistics		
Cronbach's Alpha	N of Items	VARIABLES
.707	7	Stability, Analyzability, Changeability, Testability, Trackability, Flexibility, Upgradeability.

2 and Table 3. As a general convention a value 0.7 or above is considered as a satisfactory value for further processing of data. It can be judged from the two values of Cronbach’s alpha ( i.e. .981 and .707 from Table 2

and Table 3 respectively) that obtained data is sufficiently reliable and can be considered valid for carrying out hypothesis testing for determining association between Deployability and Maintainability (if it exists).

#### 4.4 ASSOCIATION BETWEEN DEPLOYABILITY AND MAINTAINABILITY

As stated earlier Maintainability (see Table 5) of FOSS can be expressed in terms of Stability, Analyzability, Changeability, Testability, Trackability, Flexibility and Upgradeability and different sub-characteristics of Deployability (refer Table 4) characteristic are Portability, Installability, Configurability,

Table 4: Depolybilityof FOSS as seen through its Sub-Characteristics and Attributes

SN	Characteristic-SubCharacteristic-Attribute(s)
	<b>Deployability</b>
	<i>Portability and its Attribute(s)</i>
1	The FOSS system is not dependant on the machine architecture (Machine independence).
	<i>Installability and its Attribute(s)</i>
2	FOSS can be installed without any difficulty (Easy installation).
	<i>Configurability and its Attribute(s)</i>
3	The technical information about the software provided through documentation helps to configure FOSS (Technical documentation).
	<i>Adaptability and its Attribute(s)</i>
4	FOSS supports such features which make personalization possible (Appropriate for personalization).
5	The FOSS is developed in such way that it goes well with the requirements of variety of users (Suitability).
	<i>Distributability and its Attribute(s)</i>
6	FOSS system can be set up as a distributed environment (Diversified system).

Adapatability, and Distributability. So, looking at the relationship (if it exists) between Deployability and Maintainability, is also a different but important FOSS related concept. It is quite easy to assume that the

Table 5: Maintainability of FOSS as seen through its Sub-Characteristics and Attributes

SN	Characteristic-SubCharacteristic-Attribute(s)
	<b>Maintainability</b>
	<i>Stability and its Attribute(s)</i>
1	FOSS is unlikely to face unexpected errors and so the chances of the system getting stuck amid a process are minimal (No system crash/hang).
	<i>Analyzability and its Attribute(s)</i>
2	FOSS is capable of identify the root cause of bugs/errors or failures (Error detection and analysis).
	<i>Changeability and its Attribute(s)</i>
3	FOSS is suitably flexible to carry out its customization (Tailorability).
4	FOSS is capable of extending its functionality without much problems to fulfil varying user requirements (Extensible).
	<i>Testability and its Attribute(s)</i>
5	FOSS is capable to test itself before installation in the new environment for its efficient and able working (Self-testing).
6	Test suite can be used to check FOSS capability and its features (Manoeuvre control).
	<i>Trackability and its Attribute(s)</i>
7	The upgraded FOSS in terms of optimal use of available resources and throughput is better than its earlier version but does not appear to be a stranger in context of appearance when compared with it (Familiar looks).
	<i>Flexibility and its Attribute(s)</i>
8	FOSS is capable of adapting changes such as increase in the number of users of a resource (Scalability).
	<i>Upgradeability and its Attribute(s)</i>
9	FOSS can easily be upgraded to the newer version (Easily updateable).

Deployability and Maintainability are two unrelated issues. There does not seem to be any sort of correspondence between them. But, things may be different when seen from different perspectives. This is because the same perception may yield different results when subjected to different situations/conditions.

The basic idea here was to determine if there is any association between these two prime characteristics of FOSS (i.e. Deployability and Maintainability of FOSS). It was needed to know that if such relationship exists between these two entities because it seems quite reasonable to think that Deployability factor of FOSS has some influence on its Maintainability. The reason behind this thinking is that if a software (FOSS) is easily deployable in a new environment then the possibility that we can easily configure it, adapt it and distribute it, increases drastically. If so then it ought to be analyzable, stable and flexible. This reason seems to be a sufficient ground to give consideration to this thinking the authors are trying to develop and sustain.

Figure 3 portrays the association between Deployability and Maintainability. In order to prove it statistically, the two variables (i.e. Deployability and Maintainability) were subjected to statistical analysis using Fisher's Exact Test (Table 6 & 7). The null hypothesis we assume here is that Problems with Deployability of FOSS does not have any negative effect on its Maintainability. So, the quite obvious alternate hypothesis that comes into existence is poor Deployability is indicator of problematic Maintainability of FOSS.

Table 6: SPSS Output for Deployability & Maintainability Crosstabulation

Deployability * Maintainability Crosstabulation						
		Deployability				Total
		1	2	3	4	
Maintainability	1	92	33	5	1	131
	2	25	30	3	0	58
	3	8	1	2	0	11
Total		125	64	10	1	200

When Fisher's Exact Test was used we found that the results were significant (Table 7). The significant value that was obtained from this test comes out to be .001 (< 0.05), well within the allowed range. So, we do not hold the null hypothesis. In other words the alternate hypothesis is accepted and we conclude that the Deployability has (more or less) some influence on Maintainability no matter how much the two are associated with each other.

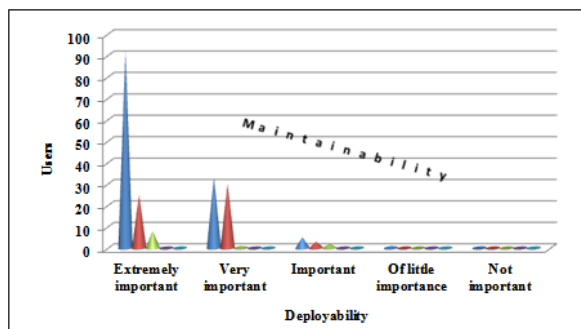


Figure 3: Association between Deployability and Maintainability

So, our thinking that we were trying to develop earlier about the association between the two variables comes out to be well justifiable as seen through the results of statistical analysis. Therefore, now we are in a position to hold this observation through a hypothesized view. The direct conclusion we draw here is that the

Table 7: SPSS output for Fisher's Exact Test

Chi-Square Tests					
	Value	Df	Asymptotic Significance(2-sided)	Exact Sig.(2-sided)	Exact Sig.(1-sided)
Pearson Chi-Square	20.401	6	.002	.029	
Likelihood Ratio	19.317	6	.004	.002	
<b>Fisher's Exact Test</b>	20.471			<b>.001</b>	
Linear-by-Linear Association	4.745	1	.029	.032	.021
N of Valid Cases	200				



more easy it is to deploy the software in the new setting, the easier it is to maintain it from different viewpoints in terms of the attributes of Maintainability.

#### 4.5 DISCUSSION

We opt for a particular software as per our convenience, there is no doubt about it (whether it is Proprietary or FOSS). But, what if we start facing problems related to different issues? It may be the software functionality, technicalities involved or the software's user interface that can be the matter of concern for someone working with the software as a normal user. One may face challenges as an advanced user also when he/she is involved in the development of the software. In other words when we deal with a software we may find our self in a problematic state when software encounters some problem or if it does not act accordingly as per the expectations. FOSS users also face such problems. In fact these moments are harder for FOSS users as compared to proprietary software users (in case of proprietary software the help is just a phone call away) because there is no immediate help around. But, it has been found that FOSS users are smart enough to tackle these problems mostly through self support and in-house staff. FOSS communities, concerned websites and friends are also a good source of help as reported by respondents. Expertise in a particular area is an imperative element that should be embedded in an individual seeking work in that area. The software field is no exception. In fact this field demands more consideration because of having highly technical temperament due to the modern high tech technologies involved. So, if a person going for some serious business related to some particular technology in this field acquires some knowledge or gets some technical lead beforehand, will definitely be able to proceed comfortably comparatively. In this regard it has been found that those professionals who undergo training are less likely to seek further assistance (online help/queries) as far as FOSS related issues are concerned. This may be because it broadens their vision regarding tackling problems. But, such people do take help occasionally as and when required depending upon the complexity of the problem encountered. So, it can be said with sufficient confidence that training certainly gives you an edge and helps in developing better insight to handle the problems. At this moment of time it is worth mentioning that training/assistance is not the necessary condition to work on FOSS platforms because it has been established through this study that people who do not undergo any formal training are also capable of handling FOSS on their own depending upon their talent, technical knowhow and skill set. The outcome of this empirical research is quite interesting showing that there holds a bond between the two basic characteristics of FOSS (i.e. Deployability and Maintainability). This association or dependability among the two underlying characteristics is vital too. This is because this relationship shows that if FOSS can be suited to a new environment in which it is being installed only then it is useful from usage point of view thereby making sense for its maintenance or Maintainability, to be more precise.

### V. CONCLUSION

The basic idea behind carrying out this empirical research was to put forward the present status pertaining to certain prime issues concerning FOSS. These issues can be considered vital because they are related to the basic functioning of FOSS in terms of its normal and advanced users. So, in order to have better understanding of these concepts an empirical study was carried out through a survey. From the survey results it can be concluded that people/professionals involved in FOSS are quite self-sufficient. They are capable of handling the normal/routine problems themselves. Occasionally they do seek help if the things go beyond their control. This help mainly comes from the people working with them or associated with them in one way or the other. As this is a very high tech age involving computers, internet and other ultra modern techniques of information exchange so, they also get some help from other sources like FOSS communities and associated web content. It has also been established through this empiricalswot up that people who undergo some sort of assistance/training sort of activity rarely need further help to carry out their routine tasks. But, at the same time it has also been established that not many people feel the need to undergo any specialised exercise related to FOSS. Yet, they manage to handle the minor problems themselves. For any other need/hitch they do seek help from the various sources of information dissemination related to FOSS like FOSS newsgroups/communities and internet in general. It can also be concluded from the statistical analysis of data that the bonding between the primary characteristics of FOSS (i.e. Deployability and Maintainability) does exist and is effective.

#### 5.1 LIMITATIONS

Apart from the characteristics considered in the present study the other prominent characteristics of FOSS like Efficiency and Reliability could also be taken into account. Moreover, the various attributes of Deployability and Maintainability indentified and considered in the present study could also be modified

orextended if required. In an empirical research following survey approach, an important but undesired element that comes into play while collecting the data is biasness in the respondents' response while taking down responses/observations. It is an inevitable factor that needs to be dealt as it does influence the outcome of the research. Another important and closely associated element that goes side by side with biasness is fuzziness. An algorithm/method could be used to handle these problems. This study lacks the implementation of any such setup to handle these issues. The exercise of such a technique increases the reliability of the final results of the study undertaken. The increased consistency as a result helps in better interpretation of the facts and figures thereby improving the accuracy of generalization of results to include the entire population.

## 5.2 SCOPE FOR FUTURE RESEARCH

Two areas are there in which progress can be made in the present study. The very first is the selection of characteristics. Other fundamental characteristics of FOSS like Efficiency, Reliability etc. can also be considered to extend this empirical research to have broader view of FOSS concepts. Further, the new attributes of the various characteristics of FOSS (apart from the existing ones) can also be identified to explore new FOSS concepts. The second area concerning the future enhancements in the study carried out can be the involvement of a suitable algorithm(s) capable of handling fuzziness and biasness. This may result in the emergence of a better picture pertaining to FOSS.

## REFERENCES

- [1] Alspaugh, T. A., Asuncion, H. U. and Scacchi, W. (2012). Software licenses, open source components, and open architectures. In: *Aligning Enterprise, System, and Software Architectures* (Eds. I. Mistr'ik, A. Tang, et al.) IGI Global, pp 58–79.
- [2] Bertoa, M. and Vallecillo, A. (2002). Quality attributes for COTS components. In: *Proceedings of the Sixth ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering*, Malaga, Spain. pp. 54-66.
- [3] ISO (2001). *Software Engineering - Product Quality - Part 1: Quality model*. ISO/IEC 9126-1. Geneva: International Organization for Standardization.
- [4] Laurent, S. and Andrew, M. (2008). *Understanding Open Source and Free Software Licensing*, O'Reilly Press, Sebastopol, CA.
- [5] Sanga C. (2010). *A technique for the evaluation of free and open source e-earning systems*. Ph.D. Thesis, University of the Western Cape.
- [6] Seffah, A., Kececi, N. and Donyaee, M. (2001). QUIM: A Framework for Quantifying Usability Metrics in Software Quality Models. *Second Asia-Pacific Conference on Quality Software (APAQS'01)*. p. 0311.