Measuring Enterprise Application Software Interoperability Capability

Andrius Valatavičius and Saulius Gudas

Institute of Data Science and Digital Technologies, Vilnius University,
Akademijos 4, Vilnius
{andrius.valatavicius, saulius.gudas}@mii.vu.lt

Abstract. Building automated solutions that ensure enterprise application interoperability requires measuring the capability of the application interoperability. The paper presents an enterprise application software (EAS) interoperability capability evaluation method. The background of the method is a more in-depth look into evaluation potentiality of interoperability by comparing edit distance of web service operations gathered for each enterprise application software. To evaluate the capability of interoperability of few enterprise application software systems (SuiteCRM, ExactOnline, NMBRS, Prestashop) web service operations and objects was compared using edit distance calculations. The edit distances have been calculated to gather data for evaluation potentiality of the interoperability solution.

Keywords: Enterprise application interoperability, Measurement of interoperability capability, Distance calculation, Autonomic interoperability component

1 Introduction

Dynamic nature of the business processes causes many problems with the already developed enterprise architecture and business process models, as well as with implemented (legacy) applications. Most common scenario when changes in business forces to replace outdated legacy software by one or multiple new software designed for some specific business process (i.e., bookkeeping software, enterprise resource planning system or e-commerce software). Changes in legacy software cause the problem of EAS integrity and interoperability. Enterprise application software (EAS) interoperability evaluation methods are highly needed. The value chain can be optimized when software applications are integrated and interoperable, and this reduces data inconsistencies and business process redundancies. There are some theoretical works concerning enterprise application interoperability measurement, but seemingly no deterministic or probabilistic methods are used in the domain. Most approaches use empirical observations, questionnaires, objective information, rather than detailed computational analysis of EAS web service properties.

Lupeikiene A., Matulevičius R., Vasilecas O. (eds.):
Baltic DB&IS 2018 Joint Proceedings of the Conference Forum and Doctoral Consortium.

Some research interoperability evaluation scope is broader and not explained by deterministic evaluation of EAS interoperability cases [1, 10]. The measurement of applications interoperability potentiality should give the essential indicators for improving interoperability. We have experimented with few edit distance formulas (Levenshtein, Jaro-Winkler, Jaccard, and Longest Common Subsequence) for evaluation of the operation names similarity of different applications. In our approach interoperability should be evaluated on the stage of architectural design of the interoperability solution by comparing names of the web service operation using existing edit distance methods.

We propose that interoperability capability evaluation should be carried out at the stage of the web service architecture analysis by comparing names of the web service operations. Applications interoperability capability is measured by comparing the names of the different web service transaction operations of the (integrated) systems: if the Transaction1 identifier is the same as the Transaction2 identifier, then the estimate is 100%. This research is limited to enterprise applications developed using service-oriented architecture and focuses on EAS that use web services over SOAP, and RESTful protocol for data transfer. When REST web service meta-data description is not standardized, it is more complicated to extract meta-data for interoperability evaluation. The interoperability capability of software systems (SuiteCRM, ExactOnline, NMBRS, PrestaShop) has been measured experimentally by comparing web service operations using edit distance calculations. The primary assumption in this research paper, that interoperability should be evaluated by comparing web service meta-data (i.e., operation names, objects, object field names, object types, and finally object values) using edit distance calculations. The EAS interoperability measurement serves as a basis for improving interoperability methods.

This paper is structured as follows. In the second section, we provide the basic concepts of interoperability capability evaluation. In the third section, we present related works and test out provided solutions within our environment setup. In the fourth section, the architecture of interoperability evaluation system is laid out. In the fifth section, the experiment environment is described, and interoperability capability measurement experiment is explained. The interoperability capability evaluation autonomic component is laid out in the seventh section. Finally, conclusions, cover the brief overview of results and summarize the experiment.

2 Basic Concepts

Interoperability is the ability of different computer systems, applications or services to communicate, share and exchange data [9]. Therefore, for EAS to be interoperable, they must be designed using SOA (Service-oriented architecture). The central principle of SOA is to have a system design that it would be internally a black-box providing description about its inputs and outputs so that user of such system would be able to interact with it [12 - 330p]. Such interactive systems can use each other's input and output to become interoperable, but there are several barriers. EAS interoperability barriers are defined in European integration framework [8, 1].

2.1 Interoperability Barriers and Areas

The problem of interoperability solutions is divided into barriers and areas. European integration framework (EIF) identifies interoperability barriers (technical, semantical, organizational and legal) [8]. Interoperability areas [1]: data, services, processes, and business.

We focus mainly on evaluating interoperability capability of EAS in areas of services and data by tackling semantical barriers. The Interoperability area of data: covers different issues of the heterogeneous data integration from diverse sources with different schemas. The Interoperability area of services: covers different issues of the heterogeneous data encapsulated by web-services of applications that designed and implemented independently.

2.2 Other Interoperability Problems

Multiple problems arise when trying to achieve EAS interoperability in a dynamic business environment. Most of EAS are also dynamic – their schema changes over time. The schema is a formal data structure description in a language understandable by database management system or the application using it. Structural changes in EAS impact business process and previous business process models become invalid. There are no methods to autonomically evaluate the potential of interoperability between EAS over the period.

To ensure EAS can be interoperable integration expert needs to perform schema alignment [7, 15, 20, 21, 23]. In the next step, the expert must ensure record linkage and data fusion [3, 11]. The expert then orchestrates jobs – the timing of each data migration component and ensures the choreography of application services and data objects – sequence and order in which applications would exchange data.

2.3 Edit Distance to Evaluate EAS Object Similarity

Interoperability potential should be evaluated using EAS web service architectural design by comparing web service operations and objects and other meta-data. We used four edit distance [17] formulas for object comparison: Levenshtein, Jaro-Winkler, Jaccard and Longest Common Subsequence for the similarity of operations evaluation. Using these calculations, we estimate interoperability capabilities of multiple EAS.

Levenshtein edit distance. Calculates edit distance by a minimum number of single character edits required to change one word into the other. Levenshtein algorithm was the first known method developed to compare string distances in 1965 [13]. For each character pair from two strings take the minimum amount of changes required to make the strings identical.

Jaro - Winkler edit distance. Calculates how many transpositions in a string required to make strings similar. A transposition is when characters of two strings are exchanged until strings become similar.

Longest common subsequence edit distance. Takes the sum of characters by calculating some subsequences that are matched and are longest in the other string.

Jaccard edit distance. For a given character of each string, a character matrix is formed where characters for each set represent the total number of characters have the same value.

3 Related Works

Various application interoperability methods are applied to maintain interoperability of enterprise applications. Most researchers of integration subject use advanced methods such as agent technologies [18], and ontology-based technologies [14, 22]. However sophisticated methods of the process integration already exist [2], just not being applied in the application area. In dynamic environment business processes often needs optimizing, similar as to [2] examples of business process integration [2, 19].

Table 1. Selected system interoperability capability measure by LISI method [10]

a) Technical view, Technical interoperability scorecard.		b) Systems view, Systems interoperability scorecard			
Source	Standards	S1	S2	S3	S4
S1	Y		Y	Y	G
S2	Y	Y		G	Y
S3	Y	Y	G		Y
S4	G	G	Y	Y	

Some researchers underlie the guidelines of measurements and give propositions of what methods should be used for interoperability capability evaluation. One of the favorite inspirers for this research Kasunic [10] proposed to evaluate systems interoperability using three views: Technical, Operational and Systems. A similar approach to the business and information systems alignment measurement introduced in [16]. Codes in Table 1 represent the usage of standards above inadequate (R), marginal (Y), or adequate (G), for the EAS (S1 – ExactOnline, S2 – PrestaShop, S3 – SuiteCRM, S4 NMBRS). Technical view table (see Table 1, a) indicates that chosen EAS are not using strong standards. Such method requires a lot of investigation and manual input, also understanding the technical aspects required for interoperability.

The enterprise application software (EAS) interoperability measurement (between services) is the basis for improving interoperability methods. Some known interoperability evaluation methods are described by these researchers: Scorecard – DoD in [10], I – Score in [5], and Comparison by functionality in [4].

These EAS interoperability evaluation methods are not sufficient because the assessments obtained through questionnaires and expert judgment. We strive to develop a method that evaluates the characteristics of the systems that is integrated - without using human input like tests, questionnaires, and experiences. The aim is to use only characteristics of software: metadata and systems network service architectures.

4 Experiment Results

This research is limited to enterprise applications developed using service-oriented architecture and mostly focus on software that uses web services and SOAP and RESTful protocol for data transfer which meta-data is usually described using standardized documents. Web service operations are compared from four software system applications for the enterprise: PrestaShop, ExactOnline, NMBRS, and SuiteCRM. Each application has some distinct roles and aspects in an enterprise:

1. **PrestaShop.** E-Commerce software system – provides a platform to create a website to sell products, also deals with the warehouse management by tracking a remaining number of products.
2. **ExactOnline.** ERP software system. Accounting and industry software – has more than one integrated tool such as enterprise resource management ERP, CRM, accounting.
3. **NMBRS.** HR-Payroll software system – helps manage and calculate payrolls and debts.
4. **SuiteCRM. It is a c**ustomer relationship management software that helps manage customer relationships by allowing plan meetings look for opportunities, deal with customers.

Some meta-data were automatically extracted from these services (therefore can be automated), other EAS require more efforts to do the extraction, but with careful re-thinking, the meta-data extraction can be automated as well. Using the meta-data of web services, we counted for each system how many distinct objects are covered by operations of web services (Fig. 1).

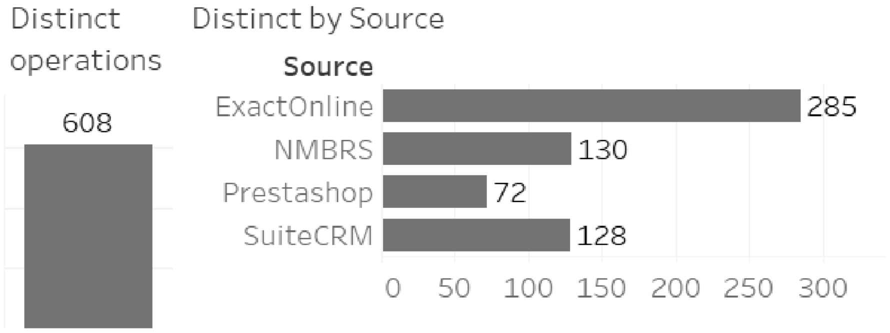


Fig. 1. The number of distinct operation objects in EAS packages

There are 608 distinct objects in considered EAS used in the experiment. On average EAS has 153 operation entities per system provided by their web service. The experiment results are the analysis of similarity for each operation name in each EAS system. If the edit distance for each operation name is high enough, this indicates that majority of operations are similar in that pair of EAS packages. The Results evaluated by the outcomes of the edit distance calculations and presented in the form of matrix M1 – M6 of the using similarity percentage for each EAS object in comparison to other EAS object. The heatmap of possible interoperability (Fig. 2) shows the edit distance of operations. The matrixes are repeated multiple times in Fig. 2 because it represents the same data combination, say Source1 X Source2 = Source2 X Source1. Consider the matrix M1 of the ExactOnline to NMBRS interoperability evaluation. Dark gray spots indicate > 85 % operation similarity compared to other operations (light gray). Dark gray area in matrix also indicate higher probability of operations being similar (above 50%), (Fig. 2). For example ExactOnline web service object „AbsenceRegistrations“ matches NMBRS web service object „Absence“ by 60% using ensemble of edit distance calculation.

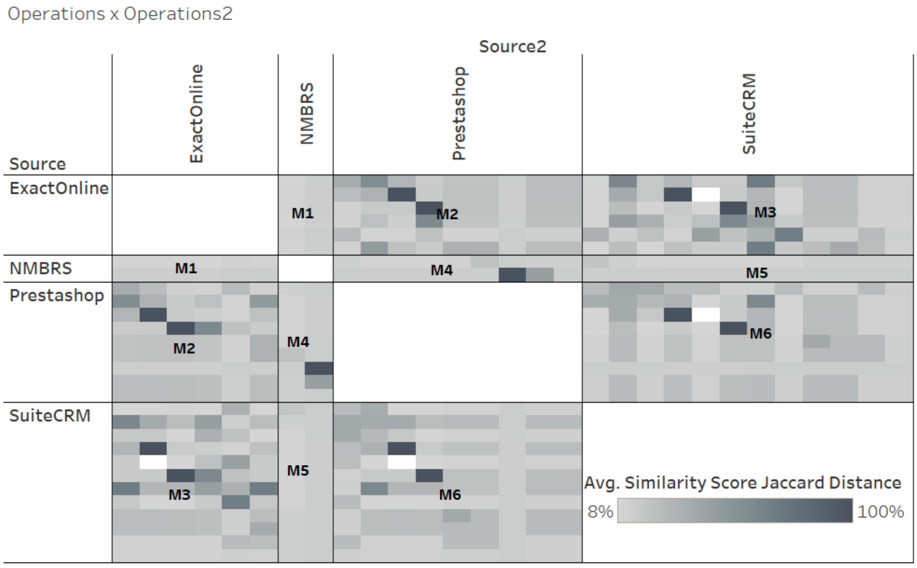


Fig. 2. Operation interoperability scoring heat map using Levenshtein edit distance algorithm

In Fig. 2 visible calculations only from one method (Levenshtein), but similar calculations were carried out for other methods as well (Jaccard, Jaro-Winkler, Longest common subsequence). We evaluate each of (M1- M6) using the ensemble edit distance – a combination of all four edit distance calculations, the separate test shows their similarity by Source X Source2. Light gray cells represent the pairs of objects that are not similar (values < 50%), Darker gray cells represent more similar pairs (values >= 50 %). In the visible figure (Fig. 2) web service operations are limited by top 20 records of Levenshtein distance and merely represent partial scope of the research done. By comparing results from each edit distance calculations, we can draw some conclusions: Jaro – Winkler and Longest common subsequence algorithms tend to evaluate more similar objects around 50 percent; Levenshtein (a) separates more but does not tend to give very high scores for seemingly similar operations. Jaccard

(c) can separate very distinct operations (much more green area) from very closely similar operations. Though for similar operations scores are not so high as described in further examination of the methods.

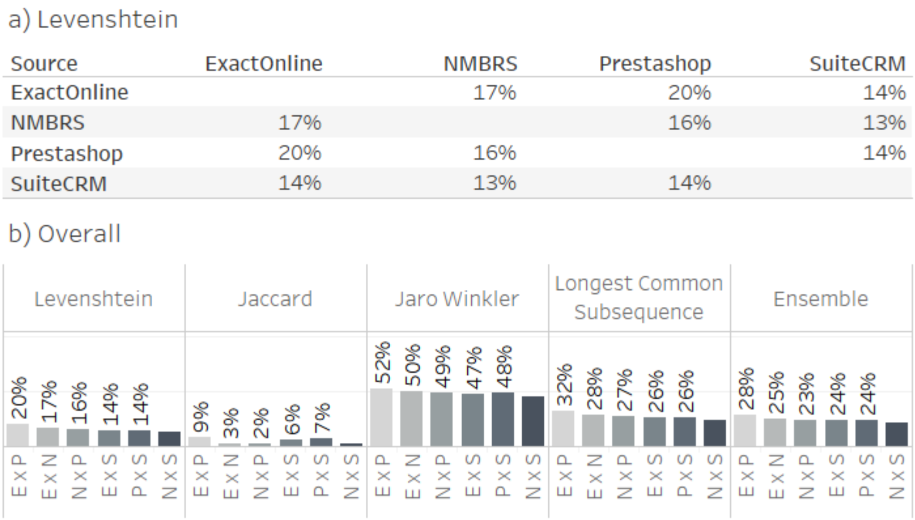
For results ensemble method (average of all similarity scores from edit distance algorithms) was selected to evaluate overall results. Assuming that objects by their same name are semantically similar, the results of the operations interoperability show that in ExactOnline (E) and NMBRS (N) there exist operation objects that are similar. Here is a brief list of example of similarity evaluation: E Addresses – N Address (85%), E BankAccounts – N BankAccount (91%), E CostCenters – N CostCenter (90%), E Costunits – N CostUnit (88%), E Departments – N Department (90%), E Employees – N Employee (88%) and E Schedules – N Schedule (88 %). But there also operation objects that are confused: E Contacts – N Contract (76%), E Contacts – N ContractPerson (72%) and E Contacts – N ContractV2 (70%) – these might actually share some similar data (as names or pointers to the right object), but need to evaluate from data structure perspective for this operation. Exact online with NMBRS has 24 operations with result higher than 65%. We could improve by determining thresholds by enriching objects with schema data and semantic meaning evaluation trying to avoid mismatching. As can be seen from all objects in ExactOnline (285) and in NMBRS (130) has only 24 operation objects with possible interoperability application with similarity score > 65%. Further, compared Exact Online (E) and PrestaShop (P) where similarity results are above or equal to 70 %. We can see that full similarity (100%) between few objects is achieved: Addresses; Contacts; Currencies; Employees; Warehouses. However, one confusion is found at (74%): E Projects – P products (74%).

Exact online with PrestaShop has 18 operations with result higher than 70 %. As can be seen, ExactOnline 285 PrestaShop 72 operations has only 18 operations possible interoperability with score > 70 % (see Table 2.). Other results are overviewed as follows and presented in Table 2. The experiment confirms that it is possible to evaluate the interoperability capability, i.e., identify the pairs of specific operations that potentially can be interoperable.

Table 2. Count of operations with a given score for each software interoperability combination

	Similarity >=		**100 %**				
	60%	**70%**	**Enseble**	**Levenshtein**	**Jaro-Winkler**	**Jaccard**	**Longest Common Subsequence**
ExactOnline x NMBRS	40	20	-	-	-	-	-
ExactOnline x Prestashop	54	18	5	5	5	5	5
ExactOnline x SuiteCRM	48	12	-	-	-	8	-
NMBRS x Prestashop	11	6	1	1	1	1	1
MMBRS x SuiteCRM	7	-	-	-	-	-	-
SuiteCRM x Prestashop	13	6	1	1	1	5	1

In Fig. 3 the similarity of sources using different edit distance calculations is depicted, where combinations of each EAS (EAS1 x EAS2) represented in letters E (ExactOnline), N (NMBRS), P (PrestaShop), S (SuiteCRM) see Fig. 3. Almost all edit distance algorithms determine the same similarity between the EAS (Fig. 3), except Jaccard method found PrestaShop and SuiteCRM more similar than ExactOnline than NMBRS.



a) Levenshtein

Source	ExactOnline	NMBRS	Prestashop	SuiteCRM
ExactOnline		17%	20%	14%
NMBRS	17%		16%	13%
Prestashop	20%	16%		14%
SuiteCRM	14%	13%	14%	

b) Overall

Fig. 3. The similarity of sources using edit distance calculations a) Levenshtein and b) overall

The scoring amplitudes are different for each edit distance method because of the difference of the edit distance calculations implemented by these methods. The lower the percentage - the more procedures tried to compare. Ultimately the score is lower because of the different amounts of procedures can be identified as similar by each edit distance method.

5 Further Work

This research is an experimental part of an investigation on autonomic solutions for application integration in the dynamic business environment using in-depth domain knowledge. Comprehensive research is still in progress, and this experimental part reveals essential knowledge on how autonomic component can evaluate whether its managed application systems are interoperable. What is more, this research provides the basis for supporting Business Process alignment to Application Processes and may impact the quality of application interoperability when using business process models. The idea is that after measuring whether software systems are interoperable, we can, in theory, measure the alignment to business processes and see which operation fall outside of business process model.

6 Conclusions

The goal of this research was a preliminary evaluation of the interoperability capability of different EAS. The lack of automated and deterministic models in the EAS interoperability capability evaluation inspired to look for interoperability measurements that can be calculated and not impacted by human input such as surveys. An attempt to compare the software systems was implemented using extracted meta-data from API interfaces. This meta-data consisted of operations from which 608 distinct objects per all EAS were identified. On average 153 objects per single EAS package.

The measurements of the capability of interoperability were implemented using the edit distance calculation methods: Jaccard, Jaro-Winkler, Levenshtein, and Longest Common Subsequence. Methods have a different level of precision estimating not such similar strings (below 60%).

The outcome suggests drilling down to characteristics of EAS web-service can be helpful for determining similar objects which could be integrated. However, this approach does not include analysis for data structures which could provide even better results and help evaluate the possible schema – matching issues.

Other methods could be used for analyzing the potential of interoperability such as text data clustering, NLP methods and Latent Dirichlet allocation [24]. These and other methods could add up to the total evaluation score.

The obtained data and use this meta-data for further research in automation and evaluation of interoperability solutions. This goal was achieved successfully and can be applied in control loop or as knowledge for autonomic interoperability component.

References

1. Chen, D., Doumeingts, G., Vernadat, F.: Architectures for enterprise integration and interoperability: Past, present and future. Computers in industry 59.7, 647-659 (2008)
2. El-Halwagi, M. M.: Process Integration. Academic Press (2016)
3. Dong, X.L., Srivastava, D.: Big data integration. In: Data Engineering (ICDE), IEEE 29th International Conference, 2013. pp. 1245-1248.
4. Dzemydienė, D., Naujikienė, R.: Elektroninių viešųjų paslaugų naudojimo ir informacinių sistemų sąveikumo vertinimas. Informacijos mokslai 50 (2009) (in Lithuanian)
5. Ford, R., Colombi, J., Graham S., Jacques, D.: Measuring system interoperability. In: Proceedings CSER 2008 (2008)
6. Heylighen, F., Joslyn, C.: Cybernetics, and second-order cybernetics. In: Encyclopedia of Physical Science & Technology 4, pp. 155-170 (2001)
7. Hohpe, G., Woolf B.: Enterprise integration patterns. In: 9th Conference on Pattern Language of Programs, p. 1-9 (2002)
8. Idabc, E., Industry, D. G.: European interoperability framework for pan-European e-government services. European Communities, (2004). Internet access: <http://ec.europa.eu/idabc/servlets/Docd552.pdf>. ISBN 92-894-8389-X.
9. Anistyasari, Y., Sarno, R., Rochmawati, N.: Designing learning management system interoperability in semantic web. In: IOP Conference Series: Materials Science and Engineering. IOP Publishing, pp. 012034 (2018)

10. Kasunic, M.: Measuring Systems Interoperability: Challenges and Opportunities. Carnegie-Mellon Univ., Software Engineering Inst. (2001)
11. Kutsche, R., Milanovic N.: Model-Based Software and Data Integration: First International Workshop. Proceedings. Vol. 8. Springer Science & Business Media. MBSDI (2008)
12. Krafzig, D., Banke K., Slama, D.: Enterprise SOA: Service-Oriented Architecture Best Practices. Prentice Hall Professional (2005)
13. Levenshtein, V.I.: Binary codes with correction of fallouts, inserts and substitutions of symbols. In: Reports of the Academy of Sciences. The Russian Academy of Sciences, 1965. p. 845-848.
14. Li, L., Wu, B., Yang, Y.: Agent-based ontology integration for ontology-based applications. In: Proceedings of the 2005 Australasian Ontology Workshop-Volume 58, Australian Computer Society, Inc., pp. 53-59 (2005)
15. McCann, R., AlShebli, B., Le, Q., Nguyen, H., Vu, L., Doan, A.: Mapping maintenance for data integration systems. In: Proceedings of the 31st International Conference on Very Large Data Bases. VLDB Endowment, pp. 1018-1029 (2005)
16. Morkevičius, A.: Business and information systems alignment method based on enterprise architecture models. Doctoral dissertation, Kaunas (2014)
17. Navarro, G.: A guided tour to approximate string matching. ACM Computing Surveys (CSUR) 33.1, 31-88 (2001)
18. Overeinder, B. J., Verkaik, P. D., Brazier, F. M. T.: Web service access management for integration with agent systems. In: Proceedings of the 2008 ACM symposium on Applied Computing. ACM, pp. 1854-1860 (2008)
19. Pavlin, G., Kamermans, M., Scafes, M.: Dynamic process integration framework: Toward efficient information processing in complex distributed systems. Informatica 34(4) (2010)
20. Peukert, E., Eberius J., Rahm E.: A self-configuring schema matching system. In: 2012 IEEE 28th International Conference on Data Engineering, pp. 306-317, IEEE (2012)
21. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. The VLDB Journal, 334-350 (2001)
22. Shvaiko, P., Euzenat J.: Ontology matching: state of the art and future challenges. IEEE Transactions on Knowledge and Data Engineering 25(1), 158-176 (2013)
23. Silverston, L., Inmon, W. H., Graziano K.: The Data Model Resource Book: A Library of Logical Data Models and Data Warehouse Designs. John Wiley & Sons, Inc. (1997)
24. Blei, D. M., N. G, Andrew. Y. Ng., Jordan, M.: Latent dirichlet allocation. Journal of Machine Learning Research 3, 993-1022 (2003)