Product Line Architecture for Automatic Evolution of Multi-Tenant Applications

Jose-Miguel Horcas, Mónica Pinto, and Lidia Fuentes
Universidad de Málaga, Andalucía Tech, Spain
Email: {horcas,pinto,lff}@lcc.uma.es

***Abstract*—Cloud computing is becoming the predominant mechanism to seamlessly deploy applications with special requirements such as massive storage sharing or load balancing, usually provided as services by cloud platforms. A developer can improve the application's delivery and productivity by following a multi tenancy approach, where variants of the same application can be quickly customized to the necessities of each tenant. However, managing the inherent variability existing in multi-tenant applications and, even more importantly, managing the evolution of a multi-tenant application with hundreds of tenants and thousands of different valid architectural configurations can become intractable if performed manually. In this paper we propose a product line architecture approach in which: (1) we use cardinality-based variability models to model each tenant as a clonable feature, (2) we automate the process of evolving the multi-tenant application architecture, and (3) we demonstrate that the implemented process is correct and efficient for a high number of tenants in a reasonable time. We use a running case study in the domain of medical software.**

ACKNOWLEDGMENT

This work is supported by the project Magic P12-TIC1814 and by the project HADAS TIN2015-64841-R (co-financed by FEDER funds), and Universidad de Málaga.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[2] R. Krebs, C. Momm, and S. Kounev, "Architectural concerns in multi-tenant saas applications." *CLOSER*, vol. 12, pp. 426–431, 2012.

[3] K. Pohl, G. Böckle, and F. J. v. d. Linden, *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag New York, Inc., 2005.

[4] E. Cavalcante, A. Almeida, T. Batista, N. Cacho, F. Lopes, F. C. Delicato, T. Sena, and P. F. Pires, "Exploiting software product lines to develop cloud computing applications," in *Software Product Line Conference*, ser. SPLC, 2012, pp. 179–187.

[5] I. Kumara, J. Han, A. Colman, T. Nguyen, and M. Kapuruge, "Sharing with a difference: Realizing service-based saas applications with runtime sharing and variation in dynamic software product lines," in *Conference on Services Computing (SCC)*, 2013, pp. 567–574.

[6] L. Baresi, S. Guinea, and L. Pasquale, "Service-oriented dynamic software product lines," *Computer*, vol. 45, no. 10, pp. 42–48, 2012.

[7] K. Czarnecki, S. Helsen, and U. Eisenecker, "Formalizing cardinality-based feature models and their specialization," *SP: Improvement and Practice*, vol. 10, no. 1, pp. 7–29, 2005.

[8] R. Mietzner, A. Metzger, F. Leymann, and K. Pohl, "Variability modeling to support customization and deployment of multi-tenant-aware software as a service applications," in *Principles of Engineering Service Oriented Systems*, 2009, pp. 18–25.

[9] H. Yang, S. Zheng, W.-C. Chu, and C.-T. Tsai, "Linking functions and quality attributes for software evolution," in *APSEC*, 2012, pp. 250–259.

[10] T. Nguyen, A. Colman, and J. Han, "Modeling and managing variability in process-based service compositions," in *International Conference on Service-Oriented Computing*, ser. ICSOC, 2011, pp. 404–420.

[11] J. Bosch, *Design and use of software architectures: adopting and evolving a product-line approach*. Pearson Education, 2000.

[12] O. Haugen, B. Moller-Pedersen, J. Oldevik, G. Olsen, and A. Svendsen, "Adding standardized variability to domain specific languages," in *SPLC*, 2008.

[13] D. Betts, A. Homer, A. Jezierski, M. Narumoto, and H. Zhang, *Moving Applications to the Cloud on Windows Azure*, 3rd ed. Microsoft patterns & practices, 2013.

[14] N. Gamez and L. Fuentes, "Architectural evolution of famiware using cardinality-based feature models," *Information and Software Technology*, vol. 55, no. 3, pp. 563–580, 2013.

[15] M. Abu Matar, R. Mizouni, and S. Alzahmi, "Towards software product lines based cloud architectures," in *IEEE IC2E*, 2014, pp. 117–126.

[16] CVL Submission Team, "Common Variability Language (CVL), OMG revised submission," http://www.omgwiki.org/variability/, 2012.

[17] F. Jouault, F. Allilaire, J. Bézivin, and I. Kurtev, "ATL: A model transformation tool," *Sci. Comput. Program.*, vol. 72, no. 1–2, pp. 31–39, 2008.

[18] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[19] E. Tsang, *Foundations of constraint satisfaction*. Academic press London, 1993, vol. 289.

[20] Y. Wu, X. Peng, and W. Zhao, "Architecture evolution in software product line: An industrial case study," in *Top Productivity through Software Reuse*, 2011.

[21] N. Loughran, P. Sánchez, A. Garcia, and L. Fuentes, "Language support for managing variability in architectural models," in *Software Composition*, 2008.

[22] T. Nguyen, A. Colman, and J. Han, "Enabling the delivery of customizable web services," in *International Conference on Web Services (ICWS)*, 2012, pp. 138–145.

[23] I. Kumara, J. Han, A. Colman, and M. Kapuruge, "Runtime evolution of service-based multi-tenant SaaS applications," in *Service-Oriented Computing*. Springer, 2013, pp. 192–206.