



□ Michael Mahlberg

(mm@michaelmahlberg.de)

ist Berater und Geschäftsführer bei der Consulting Guild GmbH in Köln. Abgesehen davon hilft er Unternehmen seit dem letzten Jahrtausend als beratender Softwarearchitekt und Methoden-Coach bei der Nutzung und Verbesserung ihrer Softwareentwicklung. Seit 1985 ununterbrochen selbstständig hat er viele Technologien kommen und gehen sehen, aber noch kein Jahr erlebt, in dem Architekturen und Entwicklungsprozesse nicht zentral in seiner Tätigkeit gewesen wären.

objektspektrum themenspecial: architekturen

Ein bekanntes Land

Am 17. Dezember 1903 hob vor der kleinen Stadt Kitty Hawk im amerikanischen Bundesstaat North Carolina eine 274 kg schwere Ansammlung von Fichtenholz, Klavierdraht und Leinwand – heute bekannt als der „Wright Flyer“ – ab und läutete damit den kommerziellen motorisierten Flug ein. Am 30. September 1968 rollte die erste Boeing 747 aus ihrer Fertigungshalle – ein Flugzeugtyp, der heute noch im Einsatz ist und als „Jumbo-Jet“ jahrzehntelang als Sinnbild für Fernreisen galt –, entworfen und gebaut etwas weniger als 65 Jahre nach „Entstehung“ der Branche an sich. Am 14. Februar 1946 wurde mit dem ENIAC der erste „programmierbare Allzweckrechner“ in Betrieb genommen und läutete damit quasi die Disziplin der Softwareentwicklung ein. Zum Zeitpunkt des Erscheinens dieser Ausgabe des Online-Themenspecials „Architekturen“ ist die Softwareentwicklung also gut und gerne 67 Jahre alt und damit mehr als ein Jahr älter als der motorisierte „schwerer als Luft“-Transport zur Zeit der Entwicklung eines der Standardflugzeuge überhaupt. Die Zeit, in der die Softwareentwicklung ein neues und unbekanntes Land war, sollte also weit hinter uns liegen – und vielfach ist das auch so. Wir sehen heute eine Branche, in der Wissensgrenzen überwunden werden und solide und erprobte Praktiken aus unterschiedlichen Bereichen der System- und Softwareentwicklung in andere Bereiche überführt werden.

Von Bausteinen...

Waren beispielsweise Komponenten noch vor wenigen Jahren hauptsächlich als Bausteine für Benutzungsoberflächen sichtbar, sind inzwischen umfangreiche Infrastrukturen zur Komponentennutzung auf allen Ebenen verfügbar. **Boris Wehrle** zeigt in seinem Beitrag nicht nur die Herausforderungen bei der Entwicklung individualisierter Standardsoftware, sondern vor allem auch die Wege aus dem Verwaltungschaos. Die Mischung von kommerziellen – in diesem Fall Microsoft – und Open Source Produkten, die er beschreibt, ist inzwischen in nahezu allen „Ökosystemen“ der Softwareentwicklung

üblich und zeigt für mich ebenfalls deutlich die Reife der Branche.

Diese Reife zeigt sich auch in der Größe der Arbeitsgruppen: an heutiger Software arbeiten im Allgemeinen dutzende Personen – wenn wir in die Spielebranche gucken, sogar oft hunderte. Um Vorhaben in dieser Größenordnung in den Griff zu bekommen, haben sich auf der Entwicklungsseite Praktiken und Vorgehensmodelle aus den Bereichen lean und agile weitgehend durchgesetzt.

Der Beitrag von **Matthias Bohlen** zeigt, wie sich diese Erfolgsmodelle auch von Softwarearchitekten in ihrer täglichen Arbeiten dazu nutzen lassen, in großen

Projekten mit vielen Anforderungen aus unterschiedlichen Richtungen entspannt den Überblick zu behalten. Mit deren Hilfe können sie widersprüchliche Anforderungen kontrolliert angehen, um große Systeme schlank zu entwerfen.

Ramon Anger und **Matthias Ehlert** werfen ebenfalls einen Blick auf die Lean-Prinzipien in der Architektur. Sie erläutern, wie die Einhaltung dieser Prinzipien zu Architekturmodellen mit nahezu lebenslanger Änderbarkeit führen kann.

Nicht nur die Nutzung gemeinsamer Prinzipien in den Software-Umsetzungsmodellen und der Architekturarbeit oder die Selbstverständlichkeit von Komponen-

ten in immer mehr Bereichen zeigen die wachsende Professionalität der Softwareentwicklung in den letzten 67 Jahren.

Auch die Zusammenarbeit unterschiedlicher Unternehmen hat sich weit von der „Wir bauen sogar unseren Motor selber“-Philosophie der Gebrüder Wright entfernt und ist einer manchmal kompetitiven, manchmal kooperativen, auf jeden Fall aber unverzichtbaren Zusammenarbeit Vieler geworden.

Mit den richtigen Rahmenbedingungen – und unter anderem mit dem Aufgreifen des Service-Bus-Gedankens auch für Softwareentwicklungswerkzeuge – ist bei einer modernen Werkzeugintegration sogar ein automatisierter Datenaustausch über Unternehmensgrenzen hinweg möglich, wie **Ralf Klimpke** in seinem Artikel deutlich macht.

...und Systemen

Auch die Hardware hat sich in den letzten 67 Jahren erheblich gewandelt. Während Thomas Watson, damals Chef der IBM, im Jahre 1943 noch meinte, es gäbe weltweit einen Markt für „ungefähr fünf Computer“, haben viele von uns alleine diese Anzahl im Haushalt – besonders, wenn man Smartphones, computergesteuerte Waschmaschinen und ähnliches mitzählt.

Die meisten Systeme, die im kommerziellen Umfeld entwickelt werden, laufen dementsprechend auch nicht mehr auf ei-

nem Rechner, sondern benötigen als Lebensraum meistens eine größere Anzahl spezialisierter Rechner. Eine solche Umgebung für den gesamten Lebenszyklus einer Anwendung aufrechtzuerhalten, stellt uns vor ganz neue Herausforderungen bei der Entwicklung skalierbarer Softwaresysteme.

Wie man diese Welten vom Entwickler-Notebook bis zur Serverfarm konsistent hält, zeigt **Oliver Tigges** in seinem Artikel, in dem er uns zudem noch mit einer Marionette, einem Koch und einem Landstreicher bekannt macht, die sich dieser Herausforderung besonders gerne stellen.

Damit bei den vielen unterschiedlichen Anforderungen und den vielen Änderungen, die in heutigen Prozessen erlaubt und sogar erwünscht sind, die Softwarearchitekturen in agilen Projekten auch das Tragwerk, die Bogengänge und sogar die Fenster berücksichtigen können, zeigt **Dr. Markus Voß** die Vergleichbarkeit und Nichtvergleichbarkeit der Architekturen von Kathedralen und IT-Systemen auf.

Den Bogen schließt **Andreas Schlapsi** mit seinem Beitrag zu technischen Schulden, in dem er einen sehr umfassenden Überblick über die aktuelle Diskussion zur Fragestellung „wie viel Abkürzungen kann man nehmen, bevor man sich im Unterholz verläuft“ gibt – natürlich übertragen auf die Softwareentwicklung. Hier zeigt sich, dass auch bei Detailschei-


dungen in der Softwareentwicklung inzwischen nicht mehr nur Bauchgefühl und Intuition ausschlaggebend sind, sondern solide Entscheidungsgrundlagen.

Alles in allem scheint mir die Softwareentwicklung tatsächlich nicht wirklich langsamer in der Entwicklung zu sein als der Flugzeugbau – das wird nicht verhindern, dass wir auch in den nächsten Jahren die Softwareanalogien zur VFW 614 (einem Flugzeug, das zu teuer für den Zielmarkt und sehr unzuverlässig war) oder dem Eurofighter (der erst fertig wurde, als der kalte Krieg schon vorbei war, sein eigentlicher Zweck also gar nicht mehr bestand) erleben werden.

Aber mit dem heute verfügbaren Handwerkszeug werden wir auch die Softwareanalogien zur Erfolgsgeschichte des Airbus oder zum ersten privat finanzierten Weltraumflug mit dem SpaceShipOne erleben und vielleicht zum Teil sogar mitgestalten. ■

In diesem Sinne wünsche ich Ihnen wieder viel Vergnügen beim Studium der Anregungen in diesem Online-Themenspecial „Architekturen“ und viele neue Ideen und Erkenntnisse.

Ihr


Michael Mahlberg