



TURNING VISIONS INTO BUSINESS

OKTOBER 2011

Scrum4Services

MALTE FOEGEN, CAROLINE GANSER, DAVID CROOME, TIMO FOEGEN

Scrum ist nicht nur in der Entwicklung, sondern auch im Service-Umfeld wertvoll. Die Komplexität von Services ist mindestens ebenso groß wie die von Entwicklungen. Deshalb ist auch der Nutzen von Scrum bei Service-Teams ähnlich groß wie bei den Entwicklungs-Teams. Mit unserer Erfahrung haben wir ein Scrum4Services Framework ausgearbeitet, das wir seit Jahren erfolgreich im Einsatz haben und kontinuierlich weiterentwickeln.



wibas als Unternehmen

wibas ist eine Unternehmensberatung, die darauf spezialisiert ist, Organisationen bei der Verbesserung von Geschäfts- und IT-Prozessen zu unterstützen.

Unsere Herausforderung

Als Unternehmen konnten wir signifikante Verbesserungen durch den Einsatz von Scrum in der Entwicklung erreichen. Entwicklung macht bei wibas aber nur einen geringen Teil der Arbeit aus. Ein weitaus größerer Teil sind Servicetätigkeiten. Für Serviceaufgaben gab es bis dato noch kein klar definiertes Scrum-Framework wie für Entwicklungsaufgaben. Wir standen daher vor der Herausforderung, für unsere Service-Teams ein „Scrum4Services“ Framework zu gestalten.

In der Zwischenzeit ist es uns gelungen, unsere gesamte Organisation auf agile Techniken umzustellen. Dabei hat Scrum4Services eine große Rolle gespielt. Wie wir zu einem Scrum Framework für Services gekommen sind und wie dieses Framework aussieht, beschreiben wir in diesem Artikel.

Was ist anders bei Services?

Es gibt zwei wesentliche Unterschiede:

- Quellen von Aufgaben: Entwicklungsteams haben einen über den Sprint planbaren Ablauf und ein festes Sprint-Ziel. Sie haben ein Product Backlog als Input. Serviceteams hingegen haben neben planbaren Aufgaben auch Kundenanfragen (service requests) und Incidents. Serviceteams haben daher zwei Quellen für Aufgaben. Die Balancierung dieser beiden Quellen ist eine der typischen Herausforderungen jedes Service-Teams.
- Art der Komplexität: Serviceaufgaben sind wesentlich repetitiver als Entwicklungsaufgaben. Die Komplexität bei Services liegt in einem schnellen, verlässlichen und wiederholbaren Service – also mehr im System und weniger in der einzelnen Aufgabe.

Unsere Idee

Unser Ansatz war es, auf Basis der Werte von Scrum ein Framework für Scrum4Services zu entwickeln. Wir sind von der Grundannahme ausgegangen, dass die Scrum-Werte, die für die Entwicklung gelten, eine Basis darstellen, die unverändert auch für Services funktioniert. Wohingegen die Scrum-

Techniken für Service-Teams angepasst und teilweise neu gestaltet werden müssen.

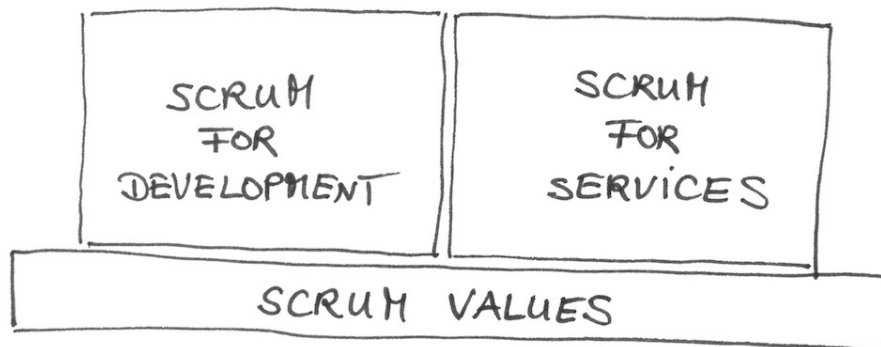


Abbildung 1: Unsere Grundannahme: Scrum4Services und Scrum4Development bauen auf denselben Grundwerten auf, haben aber unterschiedliche Frameworks, die große Gemeinsamkeiten aufweisen.

Die Grundwerte von Scrum sind:

- Früh und regelmäßig liefern
- Transparenz
- Inspect & Adapt
- „Timeboxing“
- Selbstorganisation

Unser Lösungsansatz

Im Sinne von Inspect & Adapt haben wir mit einem ersten Ansatz für ein „Scrum4Services“ Framework begonnen. Uns war bewusst, dass viele Schleifen notwendig sein werden, bis die Grundstruktur „stabil“ ist.

Um zu einem ersten Ansatz bei Scrum4Services zu kommen waren mehrere Schritte notwendig:

- Um einen ersten Ansatz zu finden haben wir die typischen Praktiken einer Serviceorganisation und der Scrum-Werte als eine kleine Tabelle erfasst (vgl. Abbildung 2). Für jede der Praktiken haben wir dann konkrete Umsetzungs-Techniken identifiziert, die zu den Scrum-Werten passen. Diese Techniken stammen aus Scrum, Kanban oder sind teilweise neue Ideen.
- Das so entworfene anfängliche Framework haben wir dann in 12 Sprints (mit je 30 Tagen) immer weiter angepasst, bis wir ein stabiles Scrum4Services Framework hatten.

Das Framework, das sich aus dieser ersten Konzeption und vielen Inspect & Adapt Schleifen ergeben hat, stellen wir im Folgenden dar.

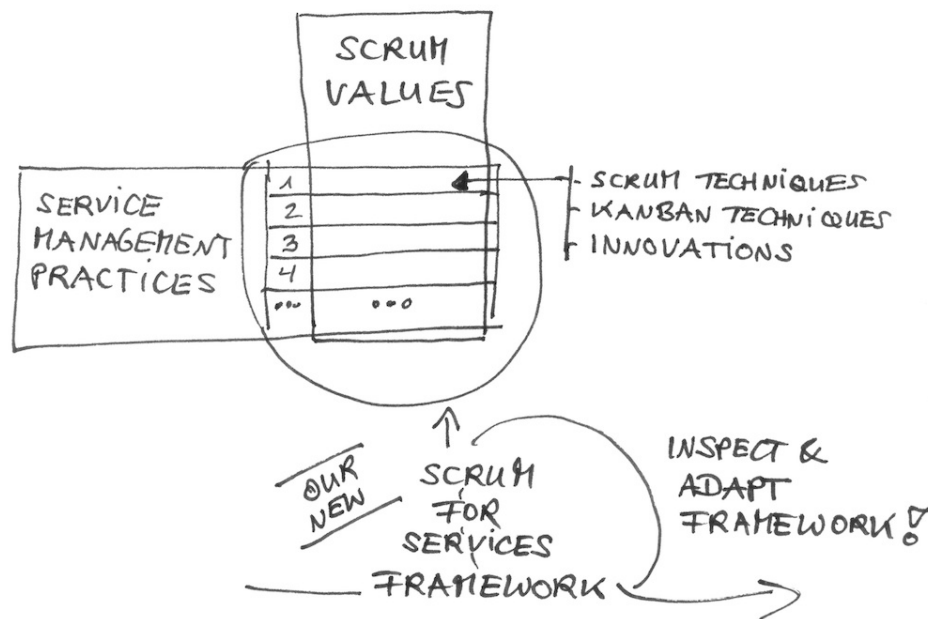


Abbildung 2: Der Weg zum ersten Scrum4Services Framework: Ideen für die Umsetzung der Service-Praktiken auf Basis der Scrum-Werte.

Von der Idee zur Umsetzung

Bevor wir das Framework näher erklären, gehen wir kurz auf den Veränderungsprozess in unserem Unternehmen ein. Wie sind wir von dem Konzept zu einer Umsetzung in der Praxis gekommen? Wir haben das Framework von einem Team zum nächsten getragen:

1. Zuerst erfolgte die Konzepterstellung mit dem ersten Service-Team und dem Führungsverantwortlichen zusammen.
2. Dann wurde das Konzept in der Praxis getestet und durch mehrere Inspect & Adapt Schleifen optimiert.
3. Anschließend haben wir es mit den Fürsprechern und Erfahrungen aus dem 1. Team auf ein weiteres Team übertragen. Dadurch sind weitere Erfahrungen in das Framework eingeflossen.
4. Nachdem das erfolgreich war, haben wir weitere Teams Stück für Stück hinzugenommen. Dabei hat sich auch die Management-Tätigkeit geändert, die heute ebenso den Scrum-Regeln folgt.
5. Alle Teammitglieder – und heute jeder Mitarbeiter in der Organisation – haben eine Ausbildung in Scrum erhalten.

Wie gehen wir mit den verschiedenen Aufgabenquellen um?

Der größte Unterschied bei Scrum4Services sind die unterschiedlichen Quellen der Aufgaben: die geplanten und die ungeplanten Aufgaben, die Kundenanfragen (service requests) und Incidents. Die tägliche Arbeit hat daher zwei Ursprünge: das Sprint Backlog und die Request-Queue.

Zu den geplanten Aufgaben gehören:

- die zur Wahrung der Service Levels wiederkehrenden Aufgaben und
- die Tätigkeiten zur Weiterentwicklung der Services.

Zu den Kundenanfragen gehören:

- alle Serviceanfragen der Kunden und
- die Incidents.

Um die Stabilität von Scrum sicherzustellen haben wir klare Kriterien definiert:

Kundenanfragen und Incidents können nur durch unsere Kunden gestellt werden, und nur der Product Owner kann die Product Backlog Items (inklusive der Service Level Items) ordnen.

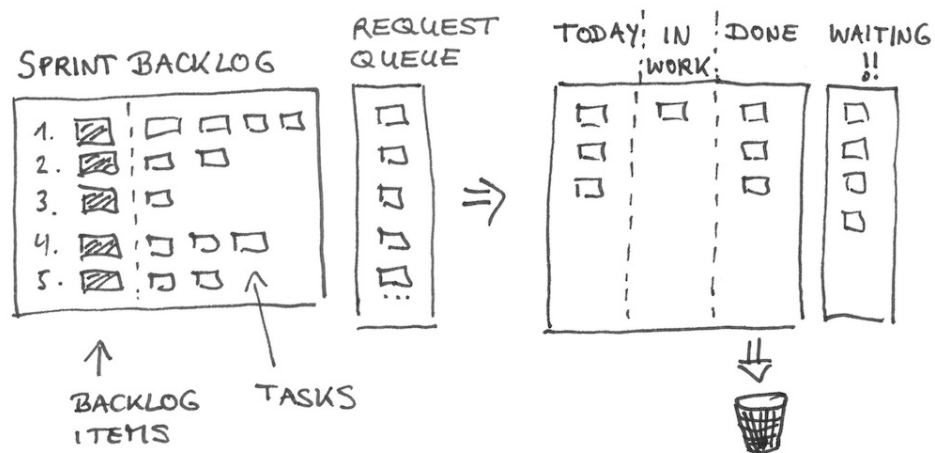


Abbildung 3: Skizze eines Scrum4Services Boards. Links stehen die beiden Input-Quellen für die Aufgaben: das Sprint Backlog und die Request-Queue. Rechts das Kanban-Board zur Bearbeitung der Aufgaben.

Zur Bearbeitung der Aufgaben haben wir das klassische Scrum Board erweitert und nutzen Kanban-Techniken. In der „Today“ Spalte stehen alle Aufgaben, die sich das Team für den Tag aus dem Sprint Backlog und der Request-Queue vorgenommen hat. In der „in Work“ Spalte stehen die Aufgaben, die gerade bearbeitet werden. In der „Done“ Spalte stehen die Aufgaben, die bereits abgearbeitet wurden. In der „Waiting“ Spalte stehen Aufgaben, bei denen das Team auf Antworten anderer (z.B. Kunden) wartet. Am Ende jedes Daily Scrums entfernen wir die Karten in der „Done“ Spalte aus dem Board.



Abbildung 4: Bild eines Scrum4Services Boards. Orange sind die Product Backlog Items, grün sind Aufgaben des Sprint Backlogs, weiß sind Kundenanfragen und Incidents und gelb sind die für die Service Levels notwendigen wiederkehrenden Aufgaben.

Events bei Scrum4Services

Der Zyklus und die Events sind bei Scrum4Services dieselben wie bei Scrum4Development.

Im Sprint Planning wird ein Forecast über die möglichen zu liefernden Product Backlog Items erstellt und die Tasks zu den Product Backlog Items geplant. Die Velocity des Teams, also die Anzahl der Storypoints, die ein Team in einem Sprint erledigt, wird inklusive der Menge der Kundenanfragen/Incidents berechnet. Auf diese Weise berücksichtigt der Forecast auch die wahrscheinliche Menge der im Sprint zu bearbeitenden Kundenanfragen.

In der produktiven Phase des Sprints stimmt sich das Team im Daily Scrum ab, indem jeder die drei Fragen beantwortet:

- Was hab ich gestern erledigt?
- Was erledige ich heute?
- Habe ich Impediments?

Jeder Sprint schließt mit einem Sprint Review ab. Wie beim Scrum4Development ist dies ein Review der konkreten Produkte. Hierbei ist es wichtig, die Services als Produkte zu verstehen. Die Product Backlog Items sind die erfahrbaren Eigenschaften („Features“) dieser Produkte. Im Service-Umfeld sind diese häufig breiter gefächert als in der Entwicklung. Jedes Product Backlog Item hat seine eigene „Definition of Done“. Zusätzlich gibt es die allgemeine Definition of Done, die den Service Level definiert. Das Sprint Review hat sich bewährt, um gemeinsam im Team die Weiterentwicklung der Service-Produkte zu besprechen. Dies ist ein unschätzbare Wert für die strategische Entwicklung der Services.

Nach dem Sprint Review folgt die Sprint Retrospektive, bei der das Scrum-Team seine Arbeitsweise betrachtet und Verbesserungen initiiert.

Sprints bei Scrum4Services sind tendenziell länger als bei Scrum4Development, da nur ein Teil der Arbeitszeit für Product Backlog Items zur Verfügung steht. Die Sprint Länge beträgt aber nicht mehr als 30 Tage.

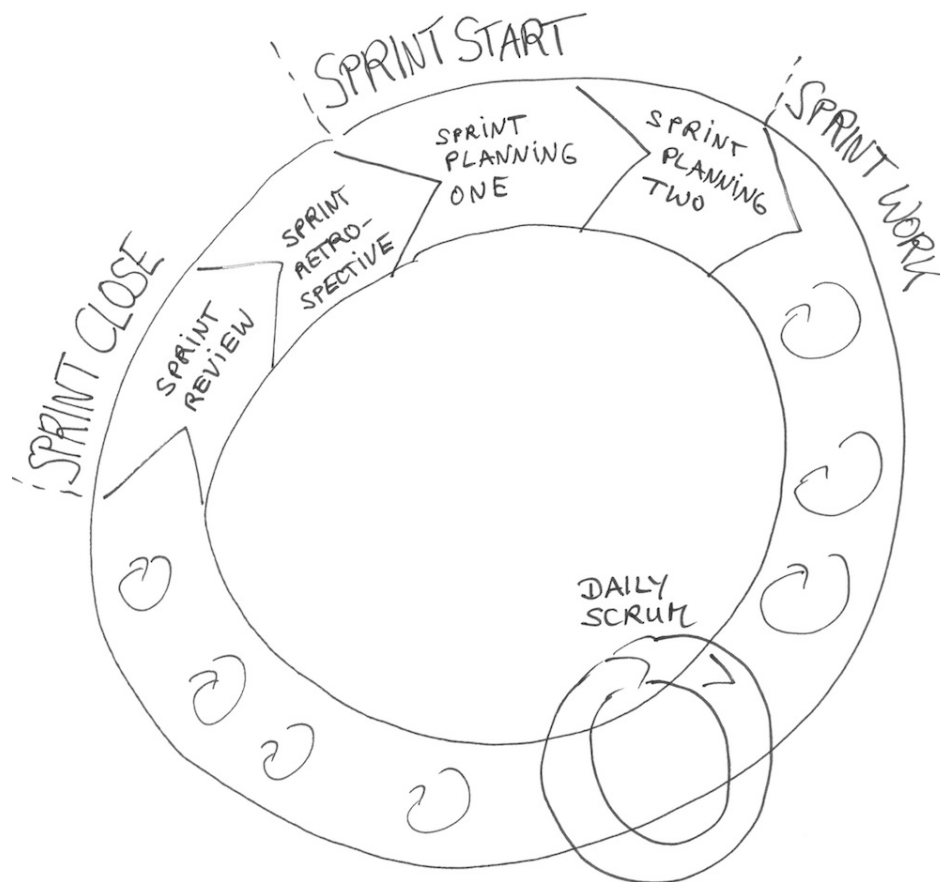


Abbildung 5: Der Scrum Flow und die Events sind bei Scrum4Services identisch zu Scrum4Development.

Artefakte bei Scrum4Services

Scrum4Services hat vier Artefakte:

- Product Backlog: eine sortierte Liste aller Features, die bei den Service-Produkten zu liefern oder neu zu entwickeln sind
- Service Request Queue: die sortierte Liste aller Service Requests und Incidents, die der Kunde gestellt hat.
- Sprint Backlog: prognostizierte Product Backlog Items und die Aufgaben zur Umsetzung
- Increment: das den Kunden bedienende Service-System (mit Menschen, Technologien und Abläufen).

Zusätzlich ist beim Scrum4Services die Service Request Queue, das alle Kundenanfragen und Incidents enthält, die vom Kunden gestellt werden.

Sprint Burndown und Velocity

Da Scrum4Services zwei Aufgabenquellen hat, nutzen wir beim Sprint-Burndown zwei Quadranten. Im oberen Quadranten tragen wir die Anzahl der verbleibenden Aufgaben im Sprint Backlog (zum Zeitpunkt des Daily Scrums) ein. Im unteren Quadranten tragen wir die Anzahl der Kundenanfragen und Incidents (zum Zeitpunkt des Daily Scrums) ein.

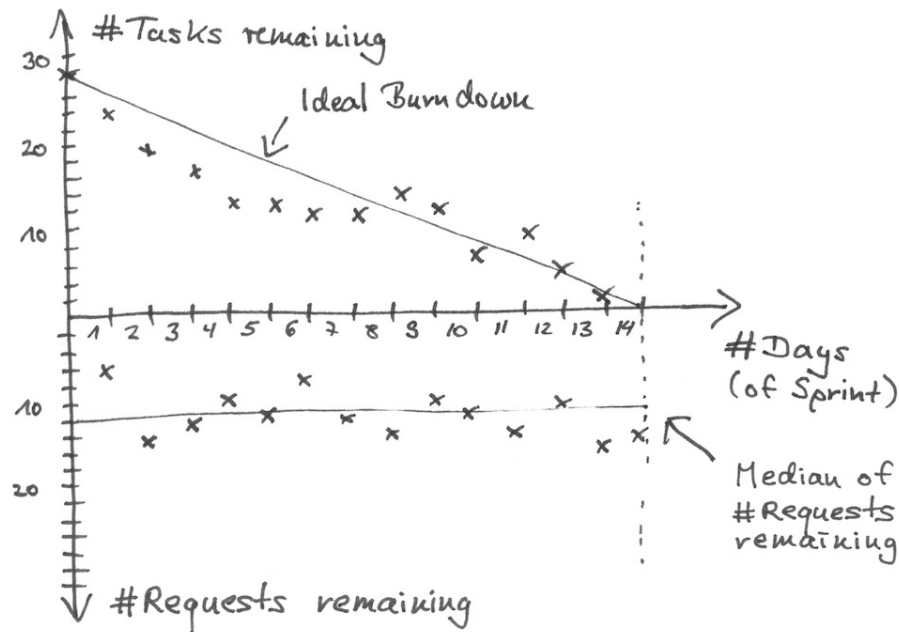


Abbildung 6: Beispiel für einen Burndown Graph bei Scrum4Services

Bei den Aufgaben des Sprint Backlogs handelt es sich um einen typischen Burndown. Der „Ideal Burndown“ zeigt die gleichmäßige Erledigung aller Aufgaben. Bei den Kundenanfragen/Incidents gibt es eine horizontale Linie, die den „Ideal Service Request and Incident Level“ zeigt, also die Anzahl an Kundenanfragen/Incidents, die wir auf Grund des Service Levels in der Request-Queue haben sollten.

Der Service Velocity Graph setzt sich ebenfalls aus zwei Quadranten zusammen. Im oberen Quadranten zeichnen wir für den jeden Sprint die Velocity im Sprint Backlog ein. Diese Velocity nutzen wir, um den Forecast der Product Backlog Items beim nächsten Sprint Planning mit Fakten zu unterstützen. Im unteren Quadranten zeichnen wir die Velocity bei der Bearbeitung der Kundenanfragen/Incidents ein. Zusätzlich zeichnen wir auch die gemeinsame Velocity ein, die uns zeigt, wie sich die Effizienz des Teams entwickelt, d.h. ob z.B. Maßnahmen aus den Retrospektiven greifen.

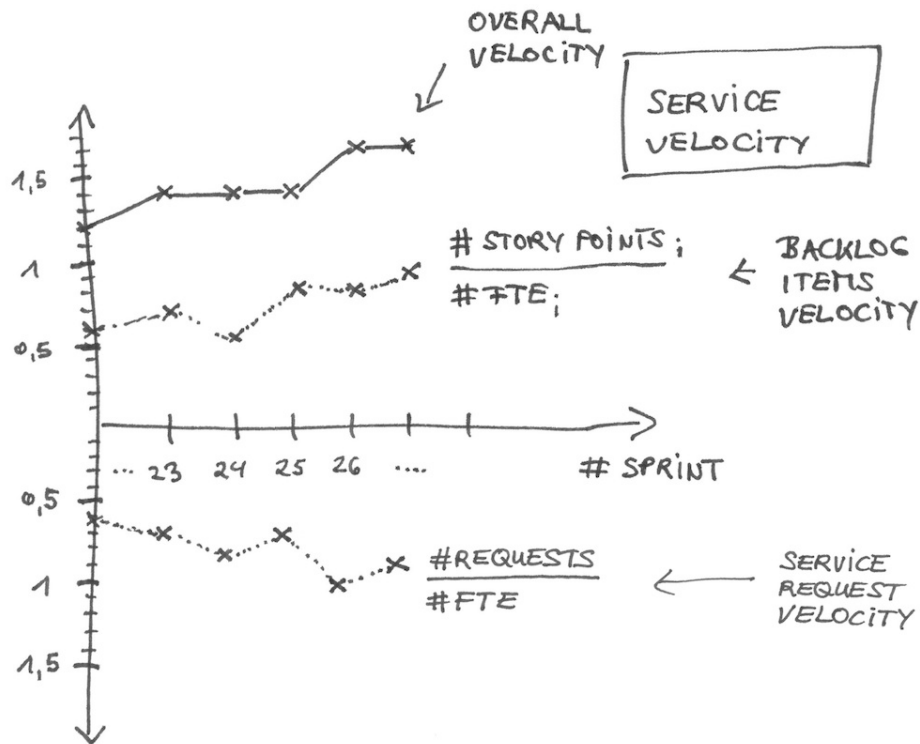


Abbildung 7: Beispiel für einen Velocity Graph bei Scrum4Services

Rollen bei Scrum4Services

Scrum4Services hat vier Rollen. Drei Rollen entsprechen dem Scrum4Development Framework. Hinzu kommt die Rolle des Kunden.

- Product Owner: pflegt das Product Backlog, ist für die strategische Weiterentwicklung der Services verantwortlich und priorisiert die Product Backlog Items
- Kunde (und nur er): stellt Kundenanfragen und Incidents
- Service Team: erbringt die Services
- Scrum Master: ist für den Prozess verantwortlich, moderiert den Prozess, schützt das Team und kümmert sich um die Impediments.

Fazit

Scrum4Services stellt ein Scrum Framework für Service-Teams dar, dass sich seit mehreren Jahren bei uns bewährt hat. Für uns ist Scrum4Services ein erfahrbarer Wettbewerbsvorteil.

Scrum4Services:

- erhöht Effektivität und Effizienz,
- setzt schnell strategische Entscheidungen um,
- ermöglicht schnell Änderungen an Services umzusetzen,
- passt zu einer zeitgemäßen Arbeitskultur.

Scrum ist nicht nur in der Entwicklung, sondern auch im Service-Umfeld wertvoll, da die Komplexität von Services mindestens ebenso groß ist wie die von Entwicklungen. Daher ist auch der Nutzen bei Service-Teams ähnlich groß wie bei Entwicklungs-Teams.

Scrum4Services zeigt uns, dass Werte und das Framework von Scrum auch auf Service-Teams übertragbar sind. Allerdings sind Techniken und Inhalte an vielen Stellen anders als beim Scrum4Development. Mit unserer Erfahrung haben wir ein Scrum4Services Framework ausgearbeitet, das wir seit Jahren erfolgreich im Einsatz haben und kontinuierlich weiterentwickeln. So haben wir z.B. in diesem Jahr unsere Strategie mit unseren Backlogs und diesem Scrum Ansatz verbunden.

Mit unser breiten und langjährigen Scrum Erfahrung gehören wir zu den Pionieren von Scrum. Gerne geben wir diese Erfahrung an Sie weiter.



TURNING VISIONS INTO BUSINESS

Copyright © 2011

wibas GmbH
Management Consultants

Otto-Hesse-Str. 19 b
64293 Darmstadt
Germany

Tel: +49 - (0)6151 - 50 33 49 - 0
Fax: +49 - (0)6151 - 50 33 49 - 33

www.wibas.de