

Crystal

Crystal ist keine einzelne Methode zur Softwareentwicklung, sondern eine Sammlung bewährter Vorgehensweisen in Softwareentwicklungsprojekten. Crystal-Entwickler Alistair Cockburn sollte Anfang der 1990er Jahre für die IBM Consulting Group eine Methode für objektorientierte Entwicklung ausarbeiten. Allerdings existierten zu dieser Zeit keine Vergleichsmöglichkeiten, an denen sich das Resultat orientieren konnte. Cockburn analysierte deshalb so viele Projekte wie nur möglich und befragte die jeweiligen Projektteams, was für den Erfolg oder Misserfolg entscheidend war.

Er kam zu dem überraschenden Ergebnis dass Projekte dann erfolgreich abgeschlossen wurden, wenn das Projektteam keine festgelegte Vorgehensweise befolgte, sondern lediglich regelmässig miteinander kommunizierte. Die Ergebnisse blieben dabei von 1991 bis 1999, international wie auch über verschiedene Entwicklungssprachen hinweg, einheitlich. Cockburn kam somit zu dem Entschluss, dass sowohl jedes Projekt als auch jedes Team einzigartig ist und keine allgemeine Vorgehensweise existiert. Aufgrund dessen erfolgt in Crystal eine grobe Abgrenzung von Softwareentwicklungsprojekten durch folgende Grössen:

- Mitarbeiteranzahl
- Kritikalität (Auswirkungen beim Scheitern)
- Prioritäten des Projektes (zum Beispiel frühzeitige Auslieferung, gesetzliche Auflagen)

Mitarbeiteranzahl steht dabei für die Anzahl der Projektmitarbeiter, die koordiniert werden sollen. Die Kritikalität stuft ein, ob sich Fehler in der Software nur auf den Komfort des Anwenders (niedrigste Stufe), den finanziellen Mehraufwand für den Auftraggeber, den Fortbestand des Unternehmens oder sogar auf Menschenleben (höchste Stufe) auswirken. Prioritäten können individuell und nach Bedarf bestimmt werden. Durch diese Dimensionen kann vor Projektbeginn die passendste Methode gewählt und bei Bedarf während des Projektes angepasst werden.

Einzelbausteine, die individuell ausgewählt und zusammengesetzt werden können, unterstützen dabei die Flexibilität dieser Methodenfamilie. Jede einzelne Methode innerhalb dieser Methodenfamilie ist nach einer Farbe (Crystal Clear, Crystal Yellow, Crystal Orange, ...) benannt. Dazu definiert Crystal wenig allgemeine Vorgaben und zwingt den Anwender zu keiner speziellen Vorgehensweise. Crystal zeigt lediglich Erfahrungen aus unterschiedlichen Projekten auf, welche Teams in welchen Situationen welche Erfahrungen gemacht haben mit den daraus resultierenden Konsequenzen.

Die verschiedenen Verfahren innerhalb von Crystal können sich voneinander stark unterscheiden, weisen allerdings folgende, gemeinsame Merkmale auf:

- Der Kunde erhält regelmässig lauffähige Zwischenversionen der Software (min. jedes Quartal)
- Probleme und Meinungsverschiedenheiten (sowohl innerhalb des Teams als auch zu Vorgesetzten) werden offen angesprochen
- Permanente Suche, Sammlung und Priorisierung von Verbesserungsvorschlägen
- Enge Kommunikation (innerhalb des Teams und zum Kunden)
- Auf Kundenseite muss stets ein erfahrener Mitarbeiter als Benutzer des zukünftigen Produktes erreichbar sein
- Mitarbeiter arbeiten zielorientiert
- Einsatz von Versionsverwaltung (besser: Konfigurations-Management)

- Häufige (automatisierte) Tests von Programmcode sowie regelmässige Erstellung einer lauffähigen Testversion.

Bewertung BQI Research

Crystal hat seine Stärke im Bereich Test (T), auch das Qualitäts-Management (QM) und die Implementierung (IMP) werden gut abgedeckt. Mittel bis schwach ausgeprägt sind dagegen Projekt-Management (PM), Requirement-Management (RM), Integration/Einführung (INT) und Wartung (W). Keine Unterstützung gibt es für Betrieb (B) und Systemdesign/technische Konzeption (SD).

- Bekanntheitsgrad und Verbreitung sind insgesamt hoch.
- Es existiert spezielle Tool-Unterstützung.
- Die Methode ist nicht normiert/standardisiert, aber zertifiziert.
- Keine Lizenzierung erforderlich.
- Support ist vorhanden.

[zurück zur Studie](#)