Metodyka Rodan Nowe Technologie

Bartosz Nowicki
Rodan Systems S.A.
bart@rodan.pl

Streszczenie

Stwierdzenie, że podstawowy kapitał firmy tworzącej oprogramowanie stanowią ludzie jest prawdziwe w odniesieniu do takich wartości jak motywacja lub kreatywność. Czy jednak opieranie zdolności produkcyjnych wyłącznie na wiedzy pracowników jest właściwe? Rodan Systems zabezpiecza swój kapitał intelektualny dotyczący inżynierii oprogramowania w metodyce Rodan Nowe Technologie. Ten zestaw pojęć, notacji, modeli, języków i procedur postępowania pozawala na świadomą i kontrolowaną produkcję oprogramowania. Niniejszy artykuł prezentuje konkretną metodykę, jej ukierunkowanie, podstawowe elementy, architekturę dokumentacyjną oraz informacje o sposobie jej tworzenia i konserwacji.

1. Metodyka

Metodyka to zestaw pojęć, notacji, modeli, języków i procedur postępowania służący osiągnięciu określonego celu. Metodyka odpowiada na następujące pytania:

- po co? – cele, ukierunkowanie, strategia,
- jakie rezultaty? – dokumenty, oczekiwania, kryteria dla kontroli jakości,
- kiedy? – terminy, zobowiązania, powiązania i zależności,
- co robić? – etapy, zadania, działania,

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

- na bazie czego? – dane wejściowe,
- w jaki sposób? – notacje, szablony, narzędzia, metody, dobre praktyki,
- kto może to zrobić? – kompetencje.

Metodyka wytwarzania oprogramowania jest zazwyczaj zorganizowana wokół cyklu życia systemu informatycznego, który jest strukturą obrazującą podstawowe działania wytwarzania, wdrażania i użytkowania systemu wraz z wzajemnymi relacjami pomiędzy tymi działaniami, przepływem danych pomiędzy nimi oraz ich zależnościami w czasie.

Stosowanie spójnej metodyki ma następujące cele:

- nadanie cechy trwałości wiedzy posiadanej przez Firmę i uniezależnienie się od zmiany pracowników,
- zwiększenie szans na sukces realizowanych projektów przez wykorzystanie praktyk, które udowodniły swoją skuteczność w innych projektach,
- pozyskiwanie, zabezpieczanie i upowszechnienie dobrych praktyk,
- ujednolicenie produktów i procesów wszystkich projektów Firmy,
- zwiększenie efektywności wytwarzania przez ponowne użycie,
- zwiększenie efektywności szkoleń,
- lepsze wykorzystanie metod i narzędzi,
- zobiektywizowanie rezultatów kontroli jakości,
- ulepszenie komunikacji w ramach Firmy, z odbiorcami i poddostawcami,
- ustalenie bazy dla poprawy wytwarzania oprogramowania i pomiarów.

2. Przeszłość i przyszłość

Rodan Systems docenia wagę problemów jakościowych i aktywnie działa na rzecz jakości świadczonych usług oraz dostarczanych produktów. Przykładem tych działań jest metodyka Rodan Nowe Technologie (RNT) – jej pierwsze wydanie z roku 1994 oraz wydanie drugie z roku 2000 [1]. Metodyka została opracowana w ramach działań koordynowanych przez Pion Zarządzania Jakością [2] i łącznie nazwanych Programem Poprawy Jakości Rodan.

Chociaż prace nad drugą wersją metodyki są zamknięte, to metodyka jest stale konserwowana. Wynika to zarówno ze zmian w technologii jak i zmian dotyczących strategii biznesowej Firmy. Dodatkowo metodyka powinna doskonalić się w kolejnych iteracjach pętli jakości polegającej na obserwacji skuteczności i efektywności zestandaryzowanego procesu, zbieraniu i zabezpieczaniu dobrych praktyk oraz uwzględnianiu propozycji poprawy, które są identyfikowane w ramach auditów. Tego rodzaju podejście odpowiada podstawowym założeniom systemów jakości [3,4].

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

Proces opracowania metodyki przypomina proces wytwarzania systemu informatycznego:

- audit wskazuje na fragment procesu wytwarzania, który wymaga polepszenia,
- opracowanie architektury metodyki ma wskazać na podstawowe elementy składowe oraz powiązania pomiędzy nimi (patrz rozdział Architektura),
- opracowanie dokumentu składowego metodyki wymaga sprecyzowania celów, budowy zespołu autorskiego, określenia struktury, nadzorowania postępów (zastosowanie metod zarządzania projektem),
- kontrola jakości dokumentu składowego metodyki ma potwierdzić przydatność dokumentu (zastosowanie metod recenzji, inspekcji),
- wdrażanie ma na celu zapewnienie dostępu, znajomości oraz stosowalności metodyki (szkolenia, pamięć korporacyjna),
- zarządzanie zmianami jest niezbędne dla każdego systemu, w którym mogą wystąpić zmiany (rejestr problemów procesów Firmy).

Rodan Systems rozważa możliwość dostosowywania systemu jakości do wymogów norm serii ISO 9000 [4]. Pierwsze doświadczenia pokazują jednak, że połączenie klasycznej inżynierii oprogramowania i wyrosłego z czystych systemów jakości ISO nie będzie łatwe. Na drodze wdrażania ISO Firma będzie przykładała dużą wagę do harmonijnego rozwoju składowej zarządzania (jakością) oraz merytorycznej. Wynika to ze zrozumienia, że oczekiwaną efektywność i skuteczność uzyskuje się dopiero poprzez synergię tych dwóch aspektów. Z tego powodu metodyka zostanie ulokowana w warstwie instrukcji roboczych systemu jakości ISO.

3. Ukierunkowanie metodyki

Przed przystąpieniem do tworzenia metodyki zdefiniowano podstawowe cechy, które metodyka powinna posiadać. Zaprezentowano je poniżej.

Zagwarantowanie sukcesu projektu. Przez sukces projektu rozumie się wytworzenie i wdrożenie systemu o odpowiednim zakresie funkcjonalnym i parametrach jakościowych wywodzących się z celów biznesowych odbiorcy w określonym czasie i budżecie. Kontrolę nad czasem i budżetem zapewnia się na drodze zarządzania projektem i współpracy z odbiorcą, a ich efektywne wykorzystanie przez odpowiednio dobraną i wdrożoną technologię wytwarzania systemu. Zarządzanie ryzykiem pozwala podjąć odpowiednie działania, które zminimalizują negatywne konsekwencje lub wyeliminują potencjalne problemy.

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

Włączenie odbiorcy w proces wytwarzania i wdrażania systemu. Reprezentanci odbiorcy biorą udział w definiowaniu systemu, w ocenie zebranych wymagań i prototypu systemu oraz współorganizują wdrożenie systemu. Zakres angażowanych reprezentantów odbiorcy rozciąga się od użytkowników pośrednich, którzy nie są zainteresowani samą funkcjonalnością systemu, ale rezultatami jego działania np. członków zarządu i właścicieli procesów biznesowych aż po administratorów i użytkowników końcowych.

Ukierunkowanie na jakość zapewnianą od samego początku. Przejawia się to oceną produktów i procesów w całym cyklu życia systemu, iteracjami po napotkaniu na problemy, przechodzeniem z problemami całego cyklu życia. Podejście przyrostowe daje możliwość potwierdzenia właściwości użytkowych części systemu lub ich doskonalenie w ramach realizacji wdrożeń kolejnych fragmentów systemu (przyrosty).

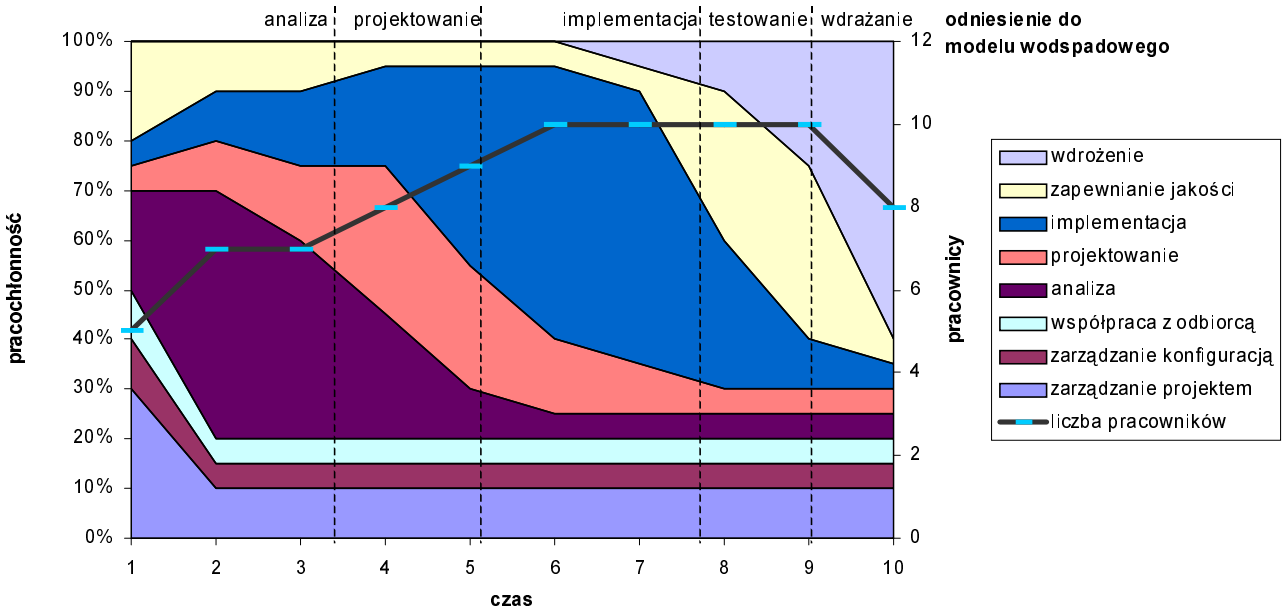
Podejście obiektowe. Metodyka RNT opiera się na popularnych metodykach obiektowych (OMT i UML [5, 6]), które uznając kluczową rolę danych systemu doceniają również jego dynamiczne aspekty. Metodyka jest wspomagana przez środki (modelowanie) wzmacniające ludzką pamięć i wyobraźnię oraz umożliwiają i ułatwiają komunikację pomiędzy odbiorcą i dostawcą oraz w ramach zespołu projektowego.

4. Podstawowe elementy metodyki

Metodyka RNT jest zbudowana wokół kilku głównych pojęć: działanie, etap, zmiana, iteracja, kamień milowy i przyrost.

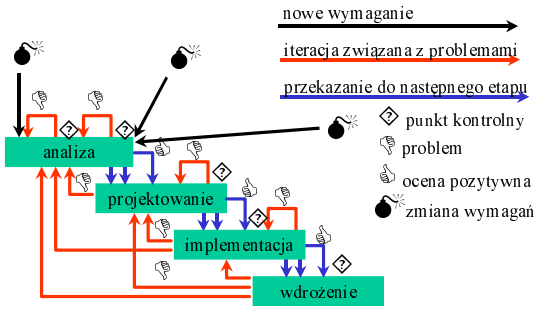
Proces wytwarzania oprogramowania składa się ze zbioru **działań** ukierunkowanych na osiągnięcie różnych celów. Na poziomie całego projektu (skala makro) działania nie następują ściśle po sobie (tak jak w tradycyjnym modelu wodospadowym), lecz nakładają się. W danym okresie czasu widać jednak znaczącą dominację działań jednego typu (Rysunek 1. Procentowy podział pracochłonności na działania.) zgodnie z kolejnością podawaną w modelu wodospadowym. Okres, w którym dominują działania danego typu nazywamy **etapem** i nazywamy zgodnie z nazwą dominującego działania.

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001



Rysunek 1. Procentowy podział pracochłonności na działania.

Działania nakładają się na siebie za względu na **zmiany**, które pojawiają się w czasie całego projektu: zmieniają się wymagania oraz pojawiają się błędy. Na poziomie pojedynczego wymagania lub błędu (skala mikro) działania następują po sobie. Jest to wynikiem przyjętego założenia o przechodzeniu z problemami przez pełen zakres działań oraz **iteracyjnego** cyklu życia (Rysunek 2. Iteracje w cyklu życia.). Jeżeli więc testowanie doprowadzi do identyfikacji błędu, to poprawa błędu polega na analizie tego problemu z punktu widzenia wymagań na system, zaprojektowaniu rozwiązania i jego usunięciu w kodzie. Prawidłowe uwzględnienie rezultatów zapewniania jakości również wymaga iteracyjnego cyklu życia.



Rysunek 2. Iteracje w cyklu życia.

Ważne momenty w projekcie (**kamienie milowe**) wynikają ze zobowiązań podjętych w stosunku do odbiorcy lub z działań zapewniania jakości. Kamienie milowe kończą poszczególne etapy projektu.

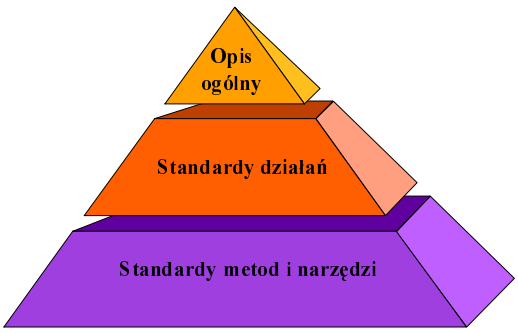
Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

Średnie i duże projekty są realizowane w kilku **przyrostach**. Przyrost to fragment systemu, który jest wdrażany niezależnie i samodzielnie może świadczyć określony zakres usług. Podejście przyrostowe pozwala odbiorcy na szybsze produkcyjne uruchomienie wybranych usług oraz ostateczne i pełne (bo wynikające z rzeczywistego wykorzystania) potwierdzenie właściwości użytkowych części systemu lub ich doskonalenie w ramach dostaw związanych z kolejnymi przyrostami (w odróżnieniu od podejścia big–bang). Jeżeli projekt trwa długo i w tym czasie zmieniają się cele biznesowe odbiorcy, to podejście przyrostowe umożliwia zmiany dotyczące celów i zakresu kolejnych przyrostów zwiększając tym samym szansę projektu na sukces.

Każdy z przyrostów jest niezależnym projektem, jednak skuteczna realizacja podejścia przyrostowego wymaga planowania na poziomie całego systemu. Przyrosty są definiowane w oparciu o całościową (ogólną) analizę systemu. Istotne jest również wstępne opracowanie całościowej i uniwersalnej architektury systemu, która „pomieści" wszystkie przyrosty.

5. Architektura metodyki

Architektura metodyki RNT jest trójpoziomowa (Rysunek 3. Architektura metodyki RNT.). Na najwyższym poziomie znajduje się dokument całościowo prezentujący metodykę RNT na wysokim poziomie ogólności. Dokument prezentuje cel metodyki, jej ukierunkowanie, przyjęty cykl życia oraz główne działania. Metodyka wskazuje na odrębne standardy definiujące poszczególne działania (poziom drugi). Przykładem standardu działania są standardy analizy, projektowania, zapewniania jakości. Standardy działań odwołują się z kolei do standardów metod i narzędzi (poziom trzeci). Przykładem standardu metod i narzędzi są standardy programowania, listy kontrolnej, testowania, kopii zapasowych, obsługi rejestru problemów.



Rysunek 3. Architektura metodyki RNT.

Każdy ze standardów jest zorganizowany wg następującego szablonu:

1. cel – określa dlaczego działanie jest wykonywane i jakie są jego rezultaty,
2. działania – określają podstawowe czynności,
3. wymagania wstępne – określają warunki, które muszą być spełnione, by działanie było wykonywane efektywnie i skutecznie,
4. role – podają listę funkcji zaangażowanych osób, ich uprawnienia, odpowiedzialności oraz umiejętności,
5. wejście – określa dokumenty, które stanowią podstawę dla realizacji działania,
6. opis procesu – prezentuje całościowo sposób realizacji,
7. wyjście – specyfikuje dokumenty, które są rezultatami,
8. metody – określają i prezentują wykorzystywane metody,
9. zalecenia praktyczne – podają zbiór dobrych praktyk,
10. dokumenty wzorcowe – wskazują na wzorce i szablony,
11. dokumenty przykładowe – wskazują na rezultaty konkretnych projektów, które mogą służyć jako punkt wyjściowy, źródło powtórnego wykorzystania lub interpretację danego standardu.

Architektura metodyki jest kluczowa dla planowego tworzenia i konserwacji. Pozwala lokować prace w obszarach, w których poprawa potrzebna jest najbardziej.

6. Prezentacja działań

Metodyka pokrywa bardzo szeroki zakres działań związanych z wytwarzaniem oprogramowania. Wyróżnia się 3 rodzaje działań:

- przygotowawcze, które następują przed rzeczywistym wytwarzaniem systemu informatycznego (strategiczny plan informatyzacji, reorganizacja procesów biznesowych, analiza wstępna, przygotowanie oferty i umowy),
- wytwarzania, które koncentrują się na wytworzeniu elementów systemu ich wdrożeniu i utrzymaniu (analiza, projektowanie, implementacja, wdrożenie, konserwacja),
- wspomagające, które powodują, że pozostałe działania są realizowane skutecznie i efektywnie (zarządzanie projektem, zarządzanie konfiguracją, zapewnianie jakości i współpraca z odbiorcą).

Strategiczny plan informatyzacji ma na celu wyznaczenie kierunków rozwoju zastosowań informatyki w organizacji odbiorcy, w przyjętym horyzoncie czasowym z uwzględnieniem celów biznesowych tej organizacji oraz możliwości technicznych zastosowań informatyki.

Reorganizacja procesów biznesowych ma na celu zmianę sposobu funkcjonowania organizacji odbiorcy w odpowiedzi na występujące problemy,

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

ograniczenia i wyznaczone cele biznesowe oraz wprowadzenie usprawnień wynikających z zastosowań informatyki i wspierających tą zmianę. Celem jest optymalizacja (z punktu widzenia kosztu, jakości, czasu) procesów realizowanych przez odbiorcę również na drodze wdrażania systemu informatycznego.

Analiza wstępna ma na celu zebranie danych będących podstawą decyzji o podjęciu dalszych działań związanych z realizacją systemu oraz opracowaniu oferty i umowy dotyczących systemu w ramach zadanych ograniczeń czasowych i dostępności potencjalnego odbiorcy.

Przygotowanie oferty i umowy ma na celu przedstawienie zobowiązań, które podejmuje dostawca oraz wymagań, które są stawiane wobec odbiorcy i gwarantujących sukces projektu oraz ustanowienie płaszczyzny, na której dochodzi do rozstrzygnięć potencjalnych konfliktów.

Analiza określa czym ma być system: jakie oraz o jakiej jakości usługi ma świadczyć system. Analiza określa co ma być zrobione (wymagania) w ramach jakich ograniczeń (ograniczenia). Analiza daje opis rzeczywistości po wprowadzeniu systemu (z uwzględnieniem reorganizacji procesów biznesowych) wraz z kontekstem dziedziny aplikacyjnej. Analiza NIE określa jak, od strony technicznej, system ma być zrealizowany. Analiza NIE ogranicza sposobów realizacji systemu (chyba, że ograniczenie jest jawne).

Projektowanie określa przy pomocy jakich rozwiązań technicznych system wyspecyfikowany na etapie analizy ma być zrealizowany. Projekt stanowi podstawę przemyślanej implementacji systemu. Projektowanie architektury systemu określa w jaki sposób system ma być zrealizowany jako całość (na wysokim poziomie abstrakcji). Projekt architektury nie dotyczy elementów drobniejszych niż pojedynczy moduł programowy. Projektowanie szczegółowe określa w jaki sposób poszczególne elementy systemu (zidentyfikowane na drodze projektowania architektury) mają zostać zrealizowane (na poziomie abstrakcji wyższym niż język programowania). Projekt szczegółowy odwzorowuje i optymalizuje obiekty rzeczywistości zdefiniowane w analizie.

Implementacja ma na celu utworzenie kodu źródłowego i wykonywalnego systemu w wybranym środowisku realizacyjnym, zgodnie z decyzjami projektowymi i wymaganiami analizy.

Wdrożenie przygotowuje organizację odbiorcy oraz system do efektywnej i skutecznej współpracy. Wdrażanie angażuje całą lub znaczącą część organizacji odbiorcy, w szczególności wiele osób i zasobów, które w normalnych warunkach zajmują się realizacją misji organizacji. Ze względu na skalę procesu wdrażania, zaangażowane zasoby i umiejscowienie działań na terenie organizacji odbiorcy, rola dostawcy sprowadza się do wspierania odbiorcy we wdrażaniu. Dostawca może uczestniczyć we wdrożeniu zarówno w aspektach technicznych jak i zarządzania, jednak odpowiedzialność za wdrożenie leży po stronie odbiorcy [7].

Konserwacja ma na celu utrzymanie i rozwój systemu, który jest wdrożony i użytkowany. Konserwacja jest wykonywana w różnym zakresie (korekcyjna, dostosowawcza, optymalizacyjna), w zależności od rodzaju umowy.

Zarządzanie projektem dotyczy aspektów organizacyjnych i ma na celu osiągnięcie przez projekt sukcesu – osiągnięcie, zarówno przez dostawcę jak i odbiorcę, celów (zakresu funkcji i jakości) w ramach określonych czasu i budżetu. Celem zarządzania jest znalezienie właściwego kompromisu pomiędzy wzajemnie kłócącymi się wymaganiami czasu (który określa terminy wykonania poszczególnych zadań), budżetu (które określają dostępność zasobów) i jakości (która określa jakie funkcje i jak dobrze zrealizowane ma posiadać system).

Zapewnienie jakości ma na celu zapewnienie by rezultaty jakościowe projektu przyczyniały się do spełnienia ogólnofirmowych celów jakościowych. Jest to osiągane przez dostosowanie technologii wytwarzania oprogramowania do specyfiki konkretnego projektu. Dobra technologia (proces, metody, szablony itd.) powoduje, że od samego początku jakość jest wbudowywana w produkty (końcowe i częściowe) wytwarzania oprogramowania. Dostosowanie technologii polega na zadecydowaniu które standardy, instrukcje robocze i w jakim zakresie będą obowiązywały w danym projekcie. Kontrola produktów i procesów jest ukierunkowana na potwierdzenie, że technologia jest stosowana prawidłowo i daje w rezultacie produkty o pożądanej jakości oraz na identyfikację problemów. Działania kontrolne stanowią dodatkowe źródło informacji o rzeczywistym (biorącym pod uwagę jakość) postępie projektu i zmniejszą niepewność związaną z gotowością oprogramowania do wykorzystania.

Zarządzanie konfiguracją ma na celu utrzymanie aktualności i spójności wszystkich elementów konfiguracji systemu informatycznego w sytuacji zmian zachodzących w projekcie. Zmiana jest podstawowym, choć niepożądanym, zjawiskiem zachodzącym w projektach informatycznych. Zmiany wynikają ze zmiany rzeczywistości, w której system ma funkcjonować lub zmiany sposobu postrzegania tej rzeczywistości (zmiany wymagań) lub są odpowiedzią na problemy, które pojawiają się w projekcie.

Współpraca z odbiorcą definiuje prawa i obowiązki odbiorcy i dostawcy w wszystkich sytuacjach, w których wymagana jest współpraca obu stron, w stopniu bardziej szczegółowym niż ma to miejsce w umowie.

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

7. Podsumowanie

W artykule zaprezentowano metodykę Rodan Nowe Technologie. Opracowane rozwiązanie nie pretenduje do miana ogólnoświatowego standardu. Wręcz przeciwnie, metodyka jest skierowana do wnętrza Firmy. Metodyka jest dostosowana do specyfiki prowadzonych projektów oraz dziedzin aplikacyjnych. W wielu zakończonych sukcesem projektach udowodniła swoją skuteczność. Odgrywa ogromną rolę jako podstawowy przedmiot ogólnofirmowego procesu poprawy jakości i zwiększa znacząco dojrzałość technologiczną Firmy. Jej szczególną wartością jest to, że jest znana i rozumiana oraz, że może być dowolnie dostosowywana do zmieniających się wymagań.

Metodyka czerpie z ogólnodostępnych metodyk i standardów, wchłania ich najlepsze rozwiązania i pomysły, uzupełnia je oraz dostosowuje do potrzeb Firmy. W wielu istotnych dla Firmy miejscach metodyka jest bardziej szczegółowa i precyzyjna niż inne, ogólnodostępne opracowania. Z drugiej strony, dzięki przejrzystej architekturze, RNT może odwoływać się do wartościowych fragmentów innych metodyk (np. modelowanie obiektowe).

Metodyka, chociaż ogranicza dowolność postępowania, została pozytywnie przyjęta przez pracowników. Nawet najwięksi indywidualiści przyznają, że „porządkuje prace", stawia „jasne wymagania", zawiera „wartościowe informacje", „skraca czas przygotowania dokumentów" i „jest pod ręką". Towarzyszący wdrażaniu metodyki program szkoleniowy znacząco wyrównał poziom przygotowania pracowników i usprawnił ich komunikację. Paradoksalnie, zbyt sztywne przywiązanie do zaleceń i szablonów może okazać się niekorzystne dla specyficznych projektów. To niekorzystne zjawisko jest niwelowane przez podkreślaną konieczność każdorazowego dostosowywania technologii.

Nieformalne odniesienie do modelu CMM [3] wskazuje, że prezentowana metodyka, w przypadku jej konsekwentnego stosowania, lokuje Firmę na poziomie 2 (wyjątkiem jest obszar zarządzania podwykonawcami) lub 3.

Dalsze prace pro–jakościowe w Rodan Systems są ukierunkowane na uporządkowanie i (być może) zautomatyzowanie w systemie zarządzania procesami pracy organizacji Firmy na najwyższym poziomie (wytwarzanie oprogramowania jest tylko jednym z procesów Firmy, a metodyka dotyczy głównie aspektów technologicznych tego procesu). Z drugiej strony Firma prowadzi ciągłe prace nad rozpoznaniem technologii na najniższym poziomie (np. wyłaniające się standardy rozwiązań internetowych, narzędzia). Rozszerzenie systemu pomiarów będzie pozwalało na ilościową ocenę skuteczności metodyki.

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001

Szkolenia z zakresu metodyki oraz mentorstwo w zastosowaniu przez poszczególne projekty są stałymi działaniami Firmy.

Bibliografia

[1]. Rodan Systems, Metodyka Rodan Nowe Technologie, wrzesień 2000

[2]. Nowicki B., Pion Zarządzania Jakością Rodan System. Rezultaty roku działalności, KKIO'99, Górski J. (ed), Kazimierz Dolny, październik 1999

[3]. Software Engineering Institute, The Capability Maturity Model: Guidelines for Improving the Software Process, Addison–Wesley, 1995

[4]. Polski Komitet Normalizacyjny, Systemy zarządzania jakością – Wymagania, PN–ISO 9001: 2000

[5]. Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorensen W., Object–Oriented Modelling and Design, Prentice Hall Int., 1991

[6]. Rumbaugh J., Jacobson I., Booch G., The Unified Modeling Language Reference Manual, Addision Wesley, 1999

[7]. Nowicki B., Dobre praktyki w trudnych wdrożeniach, II Krajowa Konferencja Inżynierii Oprogramowania KKIO'00, Zakopane 2000

Artykuł zgłoszony na III Krajową Konferencję Inżynierii Oprogramowania
Otwock, 7-20 października 2001