Tematyka zarządzania projektem metodą Łańcucha Krytycznego (Critical Chain Project Management – CCPM) nieprzypadkowo została w naszym kursie umieszczona pomiędzy modułami poświęconymi EVM oraz Scrum. Przemawia za tym kilka powodów. Mianowicie, CCPM jest rozwiązaniem, które łączy w sobie składniki „algorytmiczne” (tj. rozwiązania dotyczące metod planowania, kalkulowania, harmonogramowania, raportowania itp.) oraz „behawioralne” (filozofia postępowania, styl współpracy).

Bardziej fundamentalna i kluczowa dla sukcesu zastosowania metody jest naturalnie ta druga sfera. Na tym tle CCPM plasuje się gdzieś pomiędzy mocniej sformalizowanymi metodykami opartymi na procesach oraz indywidualnym rozliczaniu wykonawców a zakładającymi wyższą elastyczność oraz zespołowy styl działania podejściami zwinnymi. Szerzej aspektom behawioralnym oraz zagadnieniom czynnika ludzkiego w CCPM przyjrzymy się w kolejnym wykładzie.

Z punktu widzenia ćwiczeń, powinniśmy przede wszystkim zapoznać się z CCPM w jej warstwie „algorytmicznej”. Tutaj punktem wyjścia dla nas są obserwacje poczynione w EVM. Otóż, o ile wskaźniki oparte na wartości wypracowanej istotnie pozwalały wykrywać opóźnienia w realizacji projektu, to stwierdziliśmy, że nie zapewniają one wiarygodnej prognozy dotyczącej terminu ukończenia projektu. Problemem było to, że choć mogliśmy wyrazić w wartościach pieniężnych czy nawet czasowych rozmiar powstałego opóźnienia, to liczby te nie mówiły niczego o strukturze harmonogramu, a więc o łatwości czy trudności nadrobienia opóźnień. Jednym z tradycyjnych narzędzi, za pomocą którego możemy przezwyciężyć tę słabość jest analiza ścieżki krytycznej. Na tym etapie, z ćwiczeń z budowania i aktualizowania harmonogramu powinniśmy znać już pojęcia ścieżki krytycznej, zapasu swobodnego zadania (*free slack*) oraz zapasu całkowitego zadania (*total slack*). W tym miejscu warto wyjaśnić, że pojęcie łańcucha krytycznego różni się od pojęcia ścieżki krytycznej tym, że obok zależności mertytorycznych/technologicznych warunkujących kolejność realizacji zadań bierze też pod uwagę kwestię dostępności zasobów. Zatem, ilekroć w narzędziu do harmonogramowania uwzględniamy ilość dostępnych zasobów i dokonujemy wyrównywania zasobów (*resource* *leveling*), ustalamy niejako łańcuch krytyczny naszego projektu. To spostrzeżenie domyka, nazwijmy to, harmonogramowo-konwencjonalną część podejścia CCPM.

Pozostałe elementy algorytmiczne CCPM są już unikalnym wkładem tego podejścia, bo definiują specyficzny dlań sposób budowania i wykorzystania rezerw oraz określania kondycji trwającego projektu.

Wiemy, że każde opóźnienie w harmonogramie projektu stanowi potencjalnie trudny do skompensowania problem, toteż nie jest zaskoczeniem, że wielu kierowników dąży do takiego zaplanowania oraz egzekwowania realizacji prac, by każde z zadań zostało zrealizowane o czasie. Tymczasem jednak terminowa realizacja pojedynczych zadań nie przynosi korzyści organizacji realizującej projekt. Korzyść taką stanowi jedynie ukończenie całości projektu o czasie. Bazując na tym spostrzeżeniu (poszukiwanie globalnego optimum, a nie lokalnych optimów) CCPM proponuje, by rezerwy na nieprzewidywalność oraz na zdarzenia losowe zawarte dotychczas implicite w oszacowaniu każdego z zadań z osobna, „wyciągnąć przed nawias” i uczynić z nich scentralizowany bufor. Tym samym nominalne czasy realizacji przypisane zadaniom w harmonogramie stają się krótsze, i chociaż wiele z nich zostanie w trakcie realizacji przekroczonych, zużywając tym samym bufor projektu, to i tak całość przedsięwzięcia okazuje się krótsza niż w przypadku tradycyjnego harmonogramowania. Często pojawiającą się w literaturze formułą jest tutaj:

1. Skrócić tradycyjnie oszacowane „bezpieczne” czasy trwania zadań o połowę (harmonogram skraca się do 50% pierwotnego).
2. Dołączyć do harmonogramu bufor projektu w wysokości 50% zabranych rezerw (a więc ostatecznie harmonogram, wraz z przeznaczonym do zużycia buforem, będzie stanowił 75% pierwotnego).

Naturalnie, dedykowane CCPM narzędzia pozwalają na bardziej precyzyjne odzwierciedlenie specyfiki poszczególnych zadań i ich probabilistyki, toteż zetkniemy się z możliwością wpisania dla każdego z zadań dwóch czasów trwania: agresywnego (focus duration) i bezpiecznego (low risk duration).

W rozpatrywanym w naszym ćwiczeniu prostym scenariuszu pojedynczego projektu musimy zaznajomić się z dwoma typami buforów: bufor projektu (zwany także buforem ukończenia – completion buffer) oraz bufory dopływu (feeding buffer). Te ostatnie dotyczą niekrytycznych fragmentów harmonogramu, stanowiących „dopływy” do łańcucha krytycznego. Dla tego typu „dopływów” należy dopilnować, aby:

1. Zaczynały się na tyle wcześnie, by uniknąć sytuacji, w której to łańcuch krytyczny czeka na ukończenie dopływu.
2. Zaczynały się na tyle późno, aby nadmiernie wcześnie nie ponosić kosztów.

Rozstrzygnięciu ww. kwestii służy właśnie algorytm kalkulacji długości bufora dopływu. Zauważmy, że o łańcuchu niekrytycznym zakończonym chroniącym jego terminowość buforem dopływu możemy myśleć jak o mini-projekcie posiadającym własny bufor.

Poniższy rysunek ilustruje ideę konwersji harmonogramu „bezpiecznego” na harmonogram CCPM. Zwróćmy uwagę na skrócenie całości nawet po wkalkulowaniu bufora projektu.

1. Stan przed konwersją. Łańcuch krytyczny złożony z 3 zadań oraz jego dopływ złożony 2 zadań, posiadający zapas.



1. Stan po konwersji: obok skrócenia czasów zadań wprowadzono bufor projektu oraz bufor dopływu.



Uwagi dotyczące narzędzia ProChain:

* Po przełączeniu na widoki specyficzne dla ProChain, domyślna kolumna zawierająca czas realizacji zadań zostanie potraktowana jako oszacowanie „agresywne” (focus duration). Stąd też, jeśli startujemy od harmonogramu skalkulowanego tradycyjnie, wartości te będziemy skracać, zaś pierwotnie wielkości umieszczać w osobnej, specyficznej dla ProChain kolumnie (low risk duration). Jeśli tej drugiej nie wypełnimy pozostawiając w niej zero program nie zgłosi błędu, tylko zinterpretuje to jako wolę przyjęcia wartości domyślnej, która jest następująca: Low-risk duration = Focus duration x 2
* W domyślnym trybie działania ProChain w obecnej wersji nie zobaczymy jawnych buforów dopływów (zastąpiono je pojęciem „integration risk”). Jest to może mniej intuicyjne, ale racjonalne, ponieważ nie zawsze łańcuch niekrytyczny ma dostatecznie dużo luzu aby zaalokować w nim niezbędnej długości bufor dopływu. W takiej sytuacji dodatkowa rezerwa trafia do bufora projektu.

Dalsze informacje (dla zainteresowanych) – zob. np. dostępny w tym samym katalogu materiał: Critical Chain Scheduling and Buffer Management.pdf