

Witold Marciszewski

Monadologia Leibniza

w porównaniu z obiektywnym paradygmatem programowania

Niniejszy szkic powstał dzięki abstraktowi do odczytu dra inż. Jarosława Strzeleckiego pt. "Czy monada może być szczegółowym przypadkiem obiektu OOP?" przewidzianego na seminarium w PW 25.X.2018. Zob. link do abstraktu w zawiadomieniu o referacie: http://marciszewski.eu/?page_id=8381.

Konfrontacje z tekstem "Monadologii"

Zadaniem tych uwag jest skonfrontowanie danych przez Autora interpretacji "Monadologii" z odpowiednimi tekstami tego utworu. W tabeli zestawiającej tezy monadologii z zasadami projektowania obiektowego Autor zawarł próbę definicji monady, rozpisując ją na 12 numerowanych wierszy. Cytuję te elementy definicji wg ich numerów, poprzedzając je literą R (od "Referat"). Odpowiadające im ustępy z tekstu "Monadologii" oznaczam numerami ustępów z tekstu Leibniza, poprzedzonymi literą M.

R.1: "Monada jest substancją prostą, czyli bez części." To wyjściowe stwierdzenie ma pokrycie w zdaniu:

M.1: "Monada jest substancją prostą tzn. pozbawioną części."

R.2: "Monada jest elementem agregatu."

M.66: "W najdrobniejszej części materii [ożywionej] istnieje świat ... entelechii" (tj. monad, por. M.18).

Wg Leibniza agregaty to obiekty złożone. Są wśród nich ciała organiczne, które L. określa jako automaty boskie, tj. nie będące artefaktami człowieka (nazywa je też boskimi maszynami). Monady są częściami takich agregatów (zob. M.66: "w najdrobniejszej części materii istnieje ... świat entelechii i dusz [tj. monad]"). Nie są to części w sensie mereologicznym. Są częściami w sensie, który L. określa w jednym z listów następująco (cytuję za <https://plato.stanford.edu/entries/leibniz-exoteric/>, odc.6): *properly speaking, matter isn't composed of constitutive unities, but results from them ... Substantial unities aren't really parts, but the foundations of phenomena*".

Nasuwa się tu interpretacja informatyczna, że monada jest częścią organizmu jako agregatu w takim sensie, jak np. system operacyjny jest częścią komputera. Przy takiej jednak interpretacji załamię się postulowana w Referacie analogia z OOP, gdzie w prawym wierszu R.2 mówi się, że OOP może być agregatem. Monada zaś nie może być agregatem, bo wtedy byłaby złożona, wbrew R.1 oraz M.1. Wiersz więc R2 wskazywałby na różnicę, a nie analogię.

Jest też dwoistość w używaniu przez Leibniza słowa "automat". W sensie M.64 automaty są agregatami,

a więc są materialne, monady zaś nie, ale w M.18 pojawia się pojęcie monady jako automatu bezcielesnego w sensie bytu mającego "własne czynności wewnętrzne". Czy nie należałoby więc w Referacie odróżnić automaty bezcielesne (software?) od cielesnych (hardware?).

R.4: "Monada powstaje lub ginie za jednym zamachem". To jest sprzeczne z następującą tezą Leibniza.

M.73: "Nie ma ani całkowitych narodzin ani śmierci zupełnej, ściśle pojętej, polegającej na odosobnieniu duszy [monady]. A to, co my zwiemy narodzinami jest tylko rozwojem i wzrostem, podobnie jak to, co nazywamy śmiercią jest tylko zwinięciem się i zmniejszeniem."

M.77: "Dusza jest niezniszczalna." Jak to się ma do R.4? Jest niezniszczalna w tym sensie, że nie rozpadnie się sama z siebie, ale może być unicestwiona przez Stwórcę (por. M.6). O który sens chodzi w R.4?

R.10: "Monada jest bezcielesnym automatem."

M.64: Monada nie może być automatem, bo automaty czyli maszyny są złożone z części fizycznych. Punkt M.64 porównuje automaty wytworzone przez człowieka z "divina automata", jakimi są żywe istoty. Pierwsze są złożone skończenie; np. trybik w zegarku jest częścią w danej strukturze zegarka, ale zawarte w nim atomy nie należą do tej własnej struktury, będąc bez ograniczeń wymienne. Organizmy natomiast są podzielne w nieskończoność (nawet jeśli wiedza o nich wiedza za tym podziałem nie nadąza, kończąc się po pewnej liczbie kroków).

Powyższy tekst jest wpisem do blogu "Polemiki i Rozmówki w Cafe Aleph", gdzie można z nim dyskutować w formie komentarzy pod adresem: <http://marciszewski.eu/?p=9900>.

24.X.2018

Dziękuję za obie odpowiedzi na mój krótki komentarz z 23.X. Naprowadzają one na temat, którego nie poruszyłem w swoim wyjściowym wpisie, nie chcąc kumulować zbyt wielu kwestii. Teraz pojawił się odpowiedni kontekst

Z pozycji historyka logiki zastanawiam się nad dwoma możliwymi kierunkami interpretacji jakichś dawnych i nowych koncepcji: interpretacja nowej za pomocą dawnej, lub odwrotnie. Niech pierwsza nazywa się regresywną (cofamy się do przeszłości), a druga progresywną (interpretujemy to co przeszłe

w świetle obecnego). Przykładem progresywnej może być interpretacja prób algebraizacji logiki, jakie podejmowali matematycy 17-go i 18-go wieku, czyniona w świetle algebry Boole'a; a także interpretacja sylogistyki w kategoriach monadycznego rachunku predykatów.

Pewien rodzaj interpretacji regresywnej ma miejsce, gdy się rozważa ciąg ewolucyjny; stadium późniejsze może dalece się różnić od wcześniejszego, ale zachodzi zwykle jakieś między nimi podobieństwo. Ich porównywanie ma sens, gdy dzięki znajomości stanu wcześniejszego lepiej rozumiemy późniejszy; np. początki monarchii konstytucyjnej w Anglii (Wilhelm Orański, 1689) rzucają światło na jej późniejsze losy. Trzeba jednak wyśledzić w tym celu jakiś związek dziedziczenia czy inspiracji. Np. Leibniza projekt "Characteristica universalis" był inspiracją dla Fregego "Begriffsschrift"; wiedza o tym pozwala lepiej uchwycić doniosłość dzieła Fregego.

Teraz pytania do Prelegenta. Czy relację OOP do monadologii należy rozumieć na sposób którejś z tych interpretacji. Regresywna by znaczyła, że Autor spodziewa się dla OOP jakichś pożytków paraleli z monadologią. Progresywna – że od OOP spodziewa się

wkładu w rozjaśnienie monadologii, co istotnie by się przydało, bo po czterech wiekach pozostaje ona zagadkowa. A może moja dychotomia nie chwyta intencji Autora?

Jaki jest zysk poznawczy w porównywaniu monadologii z OOP? Tabelka w Abstrakcie zestawia podobieństwa. Komentarz Jarka wskazuje, jak rozumiem, na ciekawą różnicę, która komuś kto coś niecoś wie o Leibnizu, a jest profanem w sprawach OOP, przybliża – na zasadzie kontrastu – pewną własność tych drugich. Czy dla profana byłby to przypadek interpretacji regresywnej? Czy sięgając wstecz, po wiedzę o Leibnizu, może on lepiej zrozumieć teorię OOP?

Uwaga Jarosława Strzeleckiego, że monady nie mają zamkniętych okien nie wydaje mi się zgodna z treścią monadologii. Aby okna były zamknięte lub otwarte lub "przymknięte", trzeba żeby istniały, a wg Leibniza one w ogóle nie istnieją. Jeśli takie stany mogą mieć obiekty OOP, to rysuje się w tym punkcie różnica między monadologią i OOP; też może owocna dla zrozumienia tych drugich?

Krótko mówiąc, ciekawi mnie jaki pożytek poznawczy (teoretyczny lub praktyczny) może mieć podjęte przez Autora porównanie?