

SZCZEGÓŁOWY OPIS PRZEDMIOTU ZAMÓWIENIA

Opis istniejącego środowiska Zamawiającego

I.1. Opis ogólny

Zamawiający posiada środowisko serwerowe oparte na systemach operacyjnych Windows Server oraz środowisko bazodanowe oparte na SQL Server.

I.2. Kontekst projektowy

Zamawiający realizuje projekt Systemu Transformacji Cyfrowej i Procesowej, którego elementem jest hurtownia danych. Podprojekt hurtowni danych przewiduje wykorzystanie jako systemu bazodanowego Microsoft SQL Server, w związku z czym oprogramowanie to jest wskazane w SOPZ jako oprogramowanie referencyjne.

II. Wymagania

Przedstawione poniżej wymagania są domyślnie wymaganiami obligatoryjnymi, tj. muszą być spełnione, aby oferta była zgodna z zamówieniem.

II.1. Dostawy

Dostawa wskazanych w poniższym zestawieniu licencji oprogramowania Microsoft SQL Server Enterprise 2019 na 16 rdzeni CPU wraz z pakietami wsparcia Software Assurance w ramach programu MPSA na podstawie podpisanych przez Zamawiającego umów z firmą Microsoft (Government) lub równoważnych.

Tabela 1: Wymagania techniczne dla dostaw

Produkt	Nazwa ref.	Kod ref.	Ilość
SQL Server Enterprise Edition Core License	SQL Server Ent Core 2 LSA	AAA-03757	8

II.2. Wymagania ogólne

II.2.1. Licencje dla organizacji administracji publicznej.

str. 1



- II.2.2. Licencje muszą być licencjami bezterminowymi, Zamawiający nie dopuszcza dostawy licencji ograniczonych czasowo.
- II.2.3. Licencje muszą pozwalać na swobodne przenoszenie pomiędzy serwerami,
- II.2.4. Licencja muszą zapewniać możliwość korzystania z wcześniejszych wersji zamawianego oprogramowania i korzystania z kopii zamiennych.
- II.2.5. Dostęp do witryny producenta danego oprogramowania umożliwiający pobieranie kodu zamówionego oprogramowania i kluczy licencyjnych.
- II.2.6. Oferowane licencje muszą być zakupione w autoryzowanym kanale dystrybucji producenta.

II.3. Cechy równoważnego Oprogramowania typu Microsoft SQL Server Enterprise Core

II.3.1. Zamawiający dopuszcza zaoferowanie oprogramowania równoważnego obsługującego 16 rdzeni procesora, jeżeli łącznie zostaną spełnione następujące warunki:

- a) rozwiązania alternatywne w żadnym elemencie nie pogorszą wydajności, niezawodności i bezpieczeństwa Systemu HD w odniesieniu do infrastruktury opartej na założeniach Zamawiającego,
- b) zaproponowane rozwiązania alternatywne będą dostępne na rynku, tj. rozwijane i sprzedawane, od co najmniej 5 lat, co potwierdzą referencje wdrożonych systemów odpowiadających zamawianemu systemowi co do zakresu funkcjonalności, skali i kosztów,
- c) zaproponowane rozwiązania równoważne muszą być udostępniane przez producenta w wersji oficjalnej i produkcyjnej. Niedopuszczalne jest wykorzystanie wersji produktów typu Alpha, Beta, Community Preview (CP), prototypowej lub innej, która wyklucza wykorzystanie produkcyjne przez Zamawiającego.

II.3.2. System bazodanowy (SBD) licencjonowany na rdzenie procesora musi spełniać poniższe wymagania poprzez wbudowane mechanizmy:

1. Możliwość wykorzystania SBD, jako silnika relacyjnej bazy danych, analitycznej, wielowymiarowej bazy danych, platformy bazodanowej dla wielu aplikacji. Powinien zawierać serwer raportów, narzędzia do: definiowania raportów, wykonywania analiz biznesowych, tworzenia procesów ETL.

str. 2



2. Zintegrowane narzędzia graficzne do zarządzania systemem – SBD musi dostarczać zintegrowane narzędzia do zarządzania i konfiguracji wszystkich usług wchodzących w skład systemu (baza relacyjna, usługi analityczne, usługi raportowe, usługi transformacji danych). Narzędzia te muszą udostępniać możliwość tworzenia skryptów zarządzających systemem oraz automatyzacji ich wykonywania.
3. Zarządzanie serwerem za pomocą skryptów - SBD musi udostępniać mechanizm zarządzania systemem za pomocą uruchamianych z linii poleceń skryptów administracyjnych, które pozwolą zautomatyzować rutynowe czynności związane z zarządzaniem serwerem.
4. Wykonywanie typowych zadań administracyjnych w trybie on-line - SBD musi umożliwiać wykonywanie typowych zadań administracyjnych (indeksowanie, backup, odtwarzanie danych) bez konieczności przerywania pracy systemu lub przechodzenia w tryb jednonużytkownikowy.
5. Skalowalność systemu - SBD powinien wspierać skalowanie w kontekście wielkości rozwiązania (powinien być dostępny zarówno na platformie wieloserwerowej, jak również średniej wielkości komputerów i urządzeń mobilnych).
6. Możliwość dodawania procesorów wirtualnych bez restartu SBD. SBD powinien umożliwiać dodanie procesora wirtualnego do systemu, bez konieczności restartu silnika bazy danych.
7. SBD powinien umożliwiać dodanie wirtualnej pamięci bez konieczności restartu silnika bazy danych.
8. SBD musi umożliwiać tworzenie klastrów niezawodnościowych. Powinien również umożliwiać tworzenie klastrów niezawodnościowych, których węzły znajdują się w różnych podsięciach komputerowych.
9. Wysoka dostępność - SBD musi posiadać mechanizm pozwalający na replikację bazy danych między wieloma lokalizacjami (podstawowa i zapasowa) przy zachowaniu następujących cech:

str. 3



- 1) bez specjalnego sprzętu (rozwiązanie tylko programowe oparte o sam SBD);
 - 2) niezawodne powielanie danych w czasie rzeczywistym (potwierdzone transakcje bazodanowe);
10. Możliwość zastosowania reguł bezpieczeństwa obowiązujących w przedsiębiorstwie - wsparcie dla zdefiniowanej w przedsiębiorstwie polityki bezpieczeństwa (np. automatyczne wymuszanie zmiany haseł użytkowników, zastosowanie mechanizmu weryfikacji dostatecznego poziomu komplikacji haseł wprowadzanych przez użytkowników), możliwość zintegrowania uwierzytelniania użytkowników z Active Directory.
11. Możliwość definiowania reguł administracyjnych dla serwera lub grupy serwerów - SBD musi mieć możliwość definiowania reguł wymuszanych przez system i zarządzania nimi. Przykładem takiej reguły jest uniemożliwienie użytkownikom tworzenia obiektów baz danych o zdefiniowanych przez administratora szablonach nazw. Dodatkowo wymagana jest możliwość rejestracji i raportowania niezgodności działającego systemu ze wskazanymi regułami, bez wpływu na jego funkcjonalność.
12. Ograniczenie użycia zasobów – SBD powinien posiadać wbudowany mechanizm ograniczający wykorzystanie zasobów systemu operacyjnego (% wykorzystania czasu procesora, % wykorzystania pamięci, liczba operacji wejścia/wyjścia podsystemu dyskowego). Reguły definiujące ograniczenia dla użytkowników lub grup użytkowników dotyczące wykorzystania zasobów powinny mieć możliwość użycia w nich logiki zaimplementowanej za pomocą języka programowania (np. używanego w danym SBD języka SQL).
13. W celu zwiększenia wydajności przetwarzania system bazy danych musi posiadać wbudowaną funkcjonalność pozwalającą na rozszerzenie cache przetwarzania w pamięci RAM o dodatkową przestrzeń na dysku SSD.
14. System bazy danych, w celu zwiększenia wydajności, musi zapewniać możliwość asynchronicznego zatwierdzania transakcji

str. 4



bazodanowych (lazy commit). Włączenie asynchronicznego zatwierdzania transakcji powinno być dostępne zarówno na poziomie wybranej bazy danych, jak również z poziomu kodu pojedynczych procedur/zapytań.

15. W celu zwiększenia bezpieczeństwa i niezawodności system bazy danych powinien udostępniać komendę pozwalającą użytkownikowi na utrwalenie na dysku wszystkich zatwierdzonych asynchronicznych transakcji (lazy commit).

16. System SDB musi łączyć w sobie cechy bazy przechowywanej w pamięci RAM (IMDB) oraz tradycyjnej bazy danych (RDBMS) przechowywanej na dyskach.

17. System SDB musi zapewniać w ramach tej samej bazy danych możliwość umieszczenia wybranych tabel w pamięci RAM serwera, a pozostałych tabel w tradycyjnej postaci (na dysku).

18. SDB musi posiadać możliwość korzystania w procedurach jednocześnie z tabel przechowywanych w pamięci RAM oraz tabel przechowywanych na dyskach.

19. System SDB musi zapewniać wersjonowanie wierszy w tabelach przechowywanych w pamięci RAM.

20. W celu zwiększenia wydajności SDB musi posiadać możliwość tworzenia procedur składowanych w kodzie natywnym, to znaczy takich procedur, które są automatycznie kompilowane do kodu natywnego podczas ich tworzenia oraz składają się z instrukcji procesora, które nie wymagają dalszych kompilacji lub interpretacji.

21. Rejestrowanie zdarzeń silnika bazy danych w czasie rzeczywistym - SDB musi posiadać możliwość rejestracji zdarzeń na poziomie silnika bazy danych w czasie rzeczywistym w celach diagnostycznych, bez ujemnego wpływu na wydajność rozwiązania, pozwalać na selektywne wybieranie rejestrowanych zdarzeń (rejestrowanie tylko zdarzeń spełniających zdefiniowane warunki filtrujące, np. dotyczących tylko wskazanego obiektu). Wymagana jest rejestracja zdarzeń:

str. 5



- a. odczyt/zapis danych na dysku dla zapytań wykonywanych do baz danych (w celu wychwytywania zapytań znacząco obciążających system);
 - b. wykonanie zapytania lub procedury trwające dłużej niż zdefiniowany czas (wychwytywanie długo trwających zapytań lub procedur);
 - c. para zdarzeń zablokowanie/zwolnienie blokady na obiekcie bazy (w celu wychwytywania długotrwałych blokad obiektów bazy).
22. Możliwość rejestrowania bardzo dużej liczby zdarzeń i analizowania ich z minimalnym opóźnieniem – SBD powinien dostarczać wbudowaną platformę do tworzenia aplikacji typu CEP (Complex Event Processing). Aplikacje takie umożliwiają rejestrowanie bardzo dużej liczby zdarzeń (np. odczytów liczników lub z innych urządzeń pomiarowych, dowolnych zdarzeń występujących z dużą częstotliwością) i reagowanie na nie z minimalnym opóźnieniem. System powinien również udostępniać mechanizmy wysokiej dostępności dla tej usługi.
23. Zarządzanie pustymi wartościami w bazie danych - SBD musi efektywnie zarządzać pustymi wartościami przechowywanymi w bazie danych (NULL). W szczególności puste wartości wprowadzone do bazy danych powinny zajmować minimalny obszar pamięci.
24. Definiowanie nowych typów danych - SBD musi umożliwiać definiowanie nowych typów danych wraz z definicją specyficzną dla tych typów danych logiki operacji. Jeśli np. zdefiniujemy typ do przechowywania danych hierarchicznych, to obiekty tego typu powinny udostępnić operacje dostępu do „potomków” obiektu, „rodzica” itp. Logika operacji nowego typu danych powinna być implementowana w zaproponowanym przez Dostawcę języku programowania. Nowe typy danych nie mogą być ograniczone wyłącznie do okrojenia typów wbudowanych lub ich kombinacji.
25. Wsparcie dla technologii XML - SBD musi udostępniać mechanizmy składowania i obróbki danych w postaci struktur XML. W szczególności musi:

str. 6



- a. udostępniać typ danych do przechowywania kompletnych dokumentów XML w jednym polu tabeli;
 - b. udostępniać mechanizm walidacji struktur XML-owych względem jednego lub wielu szablonów XSD;
 - c. udostępniać język zapytań do struktur XML;
 - d. udostępniać język modyfikacji danych (DML) w strukturach XML (dodawanie, usuwanie i modyfikację zawartości struktur XML);
 - e. udostępniać możliwość indeksowania struktur XML-owych w celu optymalizacji wykonywania zapytań.
26. Wsparcie dla danych przestrzennych - SBD musi zapewniać wsparcie dla geometrycznych i geograficznych typów danych pozwalających w prosty sposób przechowywać i analizować informacje o lokalizacji obiektów, dróg i innych punktów orientacyjnych zlokalizowanych na kuli ziemskiej, a w szczególności:
- a. zapewniać możliwość wykorzystywania szerokości i długości geograficznej do opisu lokalizacji obiektów;
 - b. oferować wiele metod, które pozwalają na łatwe operowanie kształtami czy bryłami, testowanie ich wzajemnego ułożenia w układach współrzędnych oraz dokonywanie obliczeń takich wielkości, jak pola figur, odległości do punktu na linii, itp.;
 - c. obsługa geometrycznych i geograficznych typów danych powinna być dostępna z poziomu języka zapytań do systemu SBD;
 - d. typy danych geograficznych powinny być konstruowane na podstawie obiektów wektorowych, określonych w formacie Well-Known Text (WKT) lub Well-Known Binary (WKB), (powinny być to m.in. takie typy obiektów jak: lokalizacja (punkt), seria punktów, seria punktów połączonych linią, zestaw wielokątów, itp.).
27. Możliwość efektywnego przechowywania dużych obiektów binarnych większych niż 20 MB- SBD powinien umożliwiać przechowywanie i efektywne zarządzanie dużymi obiektami binarnymi (pliki graficzne,

str. 7



multimedialne, dokumenty, itp.). Obiekty te nie powinny być przechowywane w plikach bazy danych, ale w systemie plików. Jednocześnie pliki te powinny być zarządzane przez SBD (kontrola dostępu na podstawie uprawnień nadanych w SBD). Dodatkowo dane binarne powinny być dostępne dla użytkowników bazy danych jako standardowa kolumna tabeli (dostęp z poziomu zapytań języka SQL obsługiwanego przez SBD).

28. Możliwość kompresji przechowywanych danych - SBD powinien udostępniać wbudowany mechanizm kompresji zgromadzonych danych w celu osiągnięcia lepszej wydajności przy niezmienionej konfiguracji sprzętowej. SBD powinien umożliwiać kompresję UNICODE systemem UCS-2.

29. Możliwość rejestracji zmiany w rekordzie danych – SBD powinien pozwalać na rejestrację zmian w danych włącznie z zapamiętaniem stanu pojedynczego rekordu danych sprzed modyfikacji. Rozwiązanie nie powinno ujemnie wpływać na wydajność systemu i powinno być konfigurowalne bez wpływu na istniejące aplikacje korzystające z danych. Rozwiązanie powinno rejestrować także zmiany w definicji struktur danych.

30. Audyt dostępu do danych - SBD powinien pozwalać na rejestrację operacji takich jak: logowanie, wylogowanie użytkownika, zmiany w definicji obiektów bazy danych (tabele, procedury), wykonywanie przez wskazanego użytkownika operacji takich jak SELECT, INSERT, UPDATE, DELETE. Rozwiązanie powinno być niezależne od aplikacji, wbudowane w SBD.

31. Partycjonowanie danych (tabel i indeksów) - SBD powinien pozwalać na podział danych w jednej tabeli między różne fizyczne pamięci masowe zgodnie ze zdefiniowanymi warunkami podziału. Powinien udostępniać mechanizm równoległego (wielowątkowego) dostępu do danych umieszczonych w różnych partycjach. Dodatkowo powinna być dostępna możliwość szybkiego przesyłania dużych zbiorów danych poprzez mechanizm przełączania partycji (czyli dane przenoszone są z jednej tabeli do drugiej za pomocą operacji na metadanych, a nie przez fizyczne kopiowanie rekordów). Dzięki takiej funkcjonalności możliwe jest

str. 8



przeniesienie dużej liczby rekordów w bardzo krótkim czasie (rzędu sekund). Dodatkowo minimalizowane jest odczuwanie wpływu tej operacji przez użytkowników (minimalny wpływ przenoszenia danych na obciążenie systemu).

32. Wsparcie dla Indeksów kolumnowych - SBD powinien umożliwiać tworzenie indeksów przechowujących dane osobno dla każdej z kolumn tabeli łącząc je następnie w całość. Indeks powinien również wykorzystywać mechanizm kompresji oraz pozwalać na modyfikowanie danych w tabeli, dla której taki indeks utworzono. Dodatkowo tworzenie indeksu powinno być możliwe w trybie online czyli w trakcie wprowadzania modyfikacji indeksowanych danych.

33. Indeksowanie podzbioru danych w tabeli - SBD powinien umożliwiać tworzenie indeksów na podziorze danych z tabeli określonym poprzez wyrażenie filtrujące.

34. Możliwość tworzenia funkcji i procedur w innych językach programowania - SBD musi umożliwiać tworzenie procedur i funkcji z wykorzystaniem innych języków programowania, niż standardowo obsługiwany język zapytań danego SBD. System powinien umożliwiać tworzenie w tych językach m.in. agregujących funkcji użytkownika oraz wyzwalaczy. Dodatkowo powinien udostępniać środowisko do debuggowania.

35. Możliwość tworzenia rekursywnych zapytań do bazy danych - SBD musi udostępniać wbudowany mechanizm umożliwiający tworzenie rekursywnych zapytań do bazy danych bez potrzeby pisania specjalnych procedur i wywoływania ich w sposób rekurencyjny.

36. Obsługa błędów w kodzie zapytań - język zapytań i procedur w SBD musi umożliwiać zastosowanie mechanizmu przechwytywania błędów wykonania procedury (na zasadzie bloku instrukcji TRY/CATCH) – tak jak w klasycznych językach programowania.

37. Raportowanie zależności między obiektami - SBD musi udostępniać informacje o wzajemnych zależnościach między obiektami bazy danych.

38. Mechanizm zamrażania planów wykonania zapytań do bazy danych - SBD musi udostępniać mechanizm pozwalający na zamrożenie planu wykonania zapytania przez silnik bazy danych (w wyniku takiej operacji zapytanie jest zawsze wykonywane przez silnik bazy danych w ten sam sposób). Mechanizm ten daje możliwość zapewnienia przewidywalnego czasu odpowiedzi na zapytanie po przeniesieniu systemu na inny serwer (środowisko testowe i produkcyjne), migracji do innych wersji SBD, wprowadzeniu zmian sprzętowych serwera.

39. System transformacji danych - SBD musi posiadać narzędzie do graficznego projektowania transformacji danych. Narzędzie to powinno pozwalać na przygotowanie definicji transformacji w postaci pliku, które potem mogą być wykonywane automatycznie lub z asystą operatora. Transformacje powinny posiadać możliwość graficznego definiowania zarówno przepływu sterowania (program i warunki logiczne) jak i przepływu strumienia rekordów poddawanych transformacjom. Zestaw standardowych dostępnych transformacji powinien obejmować takie transformacje jak: sortowanie, wyszukiwanie wartości według klucza w tabelach słownikowych, automatyczna obsługa SCD (*Slowly Changing Dimension*) w zasilaniu hurtowni danych, pobranie danych z serwera FTP, wysłanie e-maila, łączenie danych z wykorzystaniem logiki rozmytej, poprawa jakości danych wykorzystująca integrację z dedykowanym systemem zarządzania jakością danych oraz jego bazą wiedzy i reguł walidujących. Powinna być także zapewniona możliwość tworzenia własnych transformacji. Środowisko tworzenia transformacji danych powinno udostępniać m.in.:

- a. mechanizm debuggowania tworzonego rozwiązania;
- b. mechanizm stawiania „pułapek” (breakpoints);
- c. mechanizm logowania do pliku wykonywanych przez transformację operacji;
- d. możliwość wznowienia wykonania transformacji od punktu, w którym przerwano jej wykonanie (np. w wyniku pojawienia się błędu);

str. 10



- e. możliwość cofania i ponawiania wprowadzonych przez użytkownika zmian podczas edycji transformacji (funkcja undo/redo);
 - f. mechanizm analizy przetwarzanych danych (możliwość podglądu rekordów przetwarzanych w strumieniu danych oraz tworzenia statystyk, np. histogram wartości w przetwarzanych kolumnach tabeli);
 - g. mechanizm automatyzacji publikowania utworzonych transformacji na serwerze bazy danych (w szczególności tworzenia wersji instalacyjnej pozwalającej automatyzować proces publikacji na wielu serwerach);
 - h. mechanizm tworzenia parametrów zarówno na poziomie poszczególnych pakietów, jak też na poziomie całego projektu, parametry powinny umożliwiać uruchamianie pakietów podrzędnych i przesyłanie do nich wartości parametrów z pakietu nadrzędnego;
 - i. mechanizm mapowania kolumn wykorzystujący ich nazwę i typ danych do automatycznego przemapowania kolumn w sytuacji podmiany źródła danych;
 - j. możliwość integracji z transakcjami bazy danych SBD, także rozproszonymi bez potrzeby pisania kodu.
40. Wbudowany system analityczny - SBD musi posiadać moduł pozwalający na tworzenie rozwiązań służących do analizy danych wielowymiarowych (hurtownia danych). System powinien umożliwiać pracę w dwóch trybach: wielowymiarowym (tworzenie kostek wielowymiarowych), tabelarycznym (wykorzystującym technologię in-memory BI). Powinno być możliwe tworzenie: wymiarów, miar. Wymiary powinny mieć możliwość określania dodatkowych atrybutów będących dodatkowymi poziomami agregacji. Powinna być możliwość definiowania hierarchii w obrębie wymiaru. Przykład: wymiar Lokalizacja Geograficzna. Atrybuty: miasto, gmina, województwo. Hierarchia: Województwo->Gmina.
41. Wbudowany system analityczny musi mieć możliwość wyliczania agregacji wartości miar dla zmieniających się elementów (członków)

str. 11



wymiarów i ich atrybutów. Agregacje powinny być składowane w jednym z wybranych modeli (MOLAP – wyliczone gotowe agregacje rozłącznie w stosunku do danych źródłowych, ROLAP – agregacje wyliczane w trakcie zapytania z danych źródłowych). Pojedyncza baza analityczna musi mieć możliwość mieszania modeli składowania, np. dane bieżące ROLAP, historyczne – MOLAP w sposób przezroczysty dla wykonywanych zapytań. System powinien pozwalać na integrację z relacyjną bazą danych – wymagana jest możliwość uruchomienia procesu wyliczenia agregacji zainicjowana poprzez dodanie rekordu do tabeli w relacyjnej bazie danych. Dodatkowo powinna być dostępna możliwość drążenia danych z kostki do poziomu rekordów szczegółowych z bazy relacyjnych (drill to detail).

42. Wbudowany system analityczny musi pozwalać na dodanie akcji przypisanych do elementów kostek wielowymiarowych (np. pozwalających na przejście użytkownika do raportów kontekstowych lub stron www powiązanych z przeglądaniem obszarem kostki).

43. Narzędzia do zarządzania jakością danych - SBD powinien mieć wbudowane mechanizmy do zarządzania jakością danych w organizacji. W ramach tych funkcji powinien:

- 1) udostępniać funkcje do profilowania danych (analiza i raporty dotyczące jakości danych);
- 2) udostępniać funkcje do deduplikacji danych;
- 3) określać stopień poprawności wartości atrybutu i w przypadku błędnej wartości sugerować wartość poprawną do akceptacji przez użytkownika;
- 4) umożliwiać definiowanie osobnych reguł czyszczenia dla wybranych domen (typów atrybutów);
- 5) umożliwiać definiowanie złożonych domen (zestawu kilku atrybutów) oraz ocenę jakości danych na podstawie powiązań między tymi atrybutami (np. weryfikację poprawności danych adresowych złożonych z kodu pocztowego, miasta i ulicy);

- 6) pozwalać na ręczną korektę nieprawidłowych danych w dedykowanej aplikacji (bez konieczności programowania);
 - 7) umożliwiać eksport wyników badania (poprawnych i sugerowanych wartości) do pliku tekstowego lub bazy relacyjnej, eksport powinien obejmować wartości po korekcie oraz ewentualnie te przed korektą;
 - 8) przechowywać reguły walidujące i oceniające jakość danych w dedykowanej bazie danych (bazie wiedzy);
 - 9) umożliwiać uzupełnianie i rozszerzanie bazy wiedzy o dane referencyjne pochodzące z systemów zewnętrznych;
 - 10) zapewniać mechanizmy „uczenia się” bazy wiedzy, czyli w miarę realizacji kolejnych procesów ręcznego czyszczenia danych baza wiedzy powinna umożliwiać gromadzenie tych informacji na potrzeby kolejnych procesów;
 - 11) umożliwiać wykorzystanie bazy wiedzy w automatycznym procesie czyszczenia danych (powinien integrować się z narzędziami do ekstrakcji, transformacji i ładowania danych, dzięki czemu będzie można wykorzystać te mechanizmy w automatycznym procesie ładowania danych).
44. Wbudowany system analityczny powinien posiadać narzędzie do rejestracji i śledzenia zapytań wykonywanych do baz analitycznych.
45. Wbudowany system analityczny musi umożliwiać rejestrowanie zapytań wykonywanych przez użytkowników, a następnie umożliwiać na podstawie zgromadzonych informacji na automatyczną optymalizację wydajności systemu (np. automatyczne projektowanie agregacji pozwalające na przyspieszenie wykonywania najczęściej wykonywanych zapytań do bazy danych).
46. Wbudowany system analityczny powinien obsługiwać wielojęzyczność (tworzenie obiektów wielowymiarowych w wielu językach – w zależności od ustawień na komputerze klienta).
47. Wbudowany system analityczny powinien udostępniać mechanizm zapisu danych przez użytkownika do kostek wielowymiarowych.

str. 13



48. Wbudowany system analityczny powinien umożliwiać tworzenie perspektyw na bazie wielowymiarowej pozwalających ograniczyć widok dla użytkownika tylko do pewnego podzbioru obiektów dostępnych w całej bazie danych.
49. Wbudowany system analityczny powinien umożliwiać użytkownikom tworzenie analiz InMemory, czyli przetwarzanie dużej liczby rekordów skompresowanych w pamięci RAM. Powinien umożliwiać tworzenie modeli wykorzystujących tabele pochodzące z wielu niezależnych źródeł danych i łączone między sobą relacjami.
50. Wbudowany system analityczny powinien udostępniać dedykowany język do tworzenia logiki biznesowej w modelu. Język ten powinien m.in. obsługiwać relacje utworzone między tabelami, mechanizmy time intelligence (operacje na datach i okresach) oraz zapewniać mechanizmy kontroli bezpieczeństwa i dostępu do danych na poziomie poszczególnych wierszy.
51. Wsparcie dla optymalizacji zapytań z modelu gwiazdy (fakty-wymiary) - SBD powinien udostępniać mechanizmy optymalizacji zapytań w modelu gwiazdy (tabela faktów łączona z tabelami wymiarów). Zapytania te często wykorzystywane są w hurtowniach danych i analizach wielowymiarowych. Ze względu na dużą liczbę danych wykorzystywanych w tego typu zapytaniach metody optymalizacji tego typu zapytań pozwalają znacząco zwiększyć wydajność przy tworzeniu rozwiązań hurtowni danych i wielowymiarowych struktur analitycznych (OLAP).
52. Wsparcie dla zapytań aktualizujących tabele faktów w modelach wielowymiarowych - SBD powinien udostępniać wbudowane mechanizmy pozwalające w łatwy i szybki sposób aktualizować zawartość tabel faktów (wykorzystywanych w modelach wielowymiarowych). Mechanizm ten powinien być dostępny z poziomu zapytań języka SQL obsługiwanego przez silnik bazy danych.
53. Aktywne buforowanie danych Proactive caching - SBD powinien udostępniać mechanizm odświeżania danych w strukturach wielowymiarowych, który wykrywa zmiany w systemach źródłowych i na bieżąco aktualizuje bazę wielowymiarową.

54. Wbudowany system analityczny powinien zapewniać mechanizmy dynamicznego security (każdy z użytkowników modelu powinien widzieć tylko swoje dane).
55. Wbudowany system analityczny powinien umożliwiać zasilanie modelu tabelarycznego m.in. z następujących systemów źródłowych: bazy relacyjne, bazy wielowymiarowe, modele tabelaryczne, zbiory danych przechowywane w usługach chmury publicznej, pliki płaskie.
56. Wbudowany system analityczny powinien umożliwiać działanie modelu tabelarycznego w dwóch trybach – z użyciem buforowania (możliwe opóźnienie, ale większa wydajność) oraz bez użycia buforowania (zapytania użytkowników końcowych korzystających z modelu są przesyłane bezpośrednio do źródłowej bazy relacyjnej i zwracają najbardziej aktualną wersję danych).
57. Wbudowany system analityczny musi udostępniać rozwiązania Data Mining, m.in.: algorytmy reguł związków (Association Rules), szeregów czasowych (Time Series), drzew regresji (Regression Trees), sieci neuronowych (Neural Nets oraz Naive Bayes). Dodatkowo system powinien udostępniać narzędzia do wizualizacji danych z modelu Data Mining oraz język zapytań do odpytywania tych modeli.
58. System analityczny powinien pozwalać na dodawanie własnych algorytmów oraz modułów wizualizacji modeli Data Mining.
59. Tworzenie głównych wskaźników wydajności KPI (Key Performance Indicators) - SBD musi udostępniać użytkownikom możliwość tworzenia wskaźników KPI (Key Performance Indicators) na podstawie danych zgromadzonych w strukturach wielowymiarowych. W szczególności powinien pozwalać na zdefiniowanie takich elementów, jak: wartość aktualną, cel, trend, symbol graficzny wskaźnika w zależności od stosunku wartości aktualnej do celu. System powinien umożliwiać tworzenie takich wskaźników również w modelach danych wykorzystujących technologię in-memory BI.
60. System raportowania - SBD musi posiadać możliwość definiowania i generowania raportów. Narzędzie do tworzenia raportów powinno pozwalać na ich graficzną definicję. Raporty powinny być udostępniane

str. 15



przez system protokołem HTTP (dostęp klienta za pomocą przeglądarki), bez konieczności stosowania dodatkowego oprogramowania po stronie serwera. Dodatkowo system raportowania powinien obsługiwać:

- a. raporty parametryzowane;
 - b. cache raportów (generacja raportów bez dostępu do źródła danych);
 - c. cache raportów parametryzowanych (generacja raportów bez dostępu do źródła danych, z różnymi wartościami parametrów);
 - d. współdzielenie predefiniowanych zapytań do źródeł danych;
 - e. wizualizację danych analitycznych na mapach geograficznych;
 - f. możliwość opublikowania elementu raportu (wykresu, tabeli) we współdzielonej bibliotece, z której mogą korzystać inni użytkownicy tworzący nowy raport;
 - g. możliwość wizualizacji wskaźników KPI;
 - h. możliwość wizualizacji danych w postaci obiektów sparkline.
61. Wymagane jest generowanie raportów w formatach: XML, PDF, Microsoft Excel, Microsoft Word, HTML, TIFF, PowerPoint.
62. SBD musi umożliwiać rozbudowę mechanizmów raportowania m.in. o dodatkowe formaty eksportu danych, obsługę nowych źródeł danych dla raportów, funkcje i algorytmy wykorzystywane podczas generowania raportu (np. nowe funkcje agregujące), mechanizmy zabezpieczeń dostępu do raportów.
63. SBD musi umożliwiać wysyłkę raportów drogą mailową w wybranym formacie (subskrypcja) do dynamicznej listy odbiorców (pobieranej z bazy danych np. zapytaniem SQL).
64. Wbudowany system raportowania powinien posiadać rozszerzalną architekturę oraz otwarte interfejsy do osadzania raportów oraz do integrowania rozwiązania z różnorodnymi środowiskami IT.

str. 16



65. Narzędzia do tworzenia raportów ad-hoc - SBD powinien udostępniać narzędzia do tworzenia raportów ad-hoc przez niezalogowanych użytkowników. Tworzenie raportów powinno odbywać się w środowisku graficznym. Użytkownicy powinni mieć możliwość publikowania stworzonych raportów na serwerze w celu udostępnienia ich szerszemu gronu osób.
66. SBD musi mieć wbudowane mechanizmy przetwarzania w sposób równoległy skryptów analitycznych w językach R i Python.
67. SBD musi mieć możliwość tworzenia i trenowania modeli predykcyjnych w języku R w oparciu o dane z poszczególnych partycji w bazie danych.
68. SBD musi posiadać pełne wsparcie kodowania znaków UTF-8.
69. SBD musi umożliwiać wdrażanie i zarządzanie certyfikatami SSL/TLS.
70. SDB musi umożliwiać nadawanie granularnych ról i uprawnień w zakresie całej instancji serwera bazodanowego, pojedynczych baz danych, pojedynczych obiektów w bazie danych (np. tabel, widoków) oraz pojedynczych wierszy (tzw. Row-Level Security).
71. SBD musi mieć możliwość tworzenia wyzwalaczy, procedur i funkcji składowanych w bazie oraz musi mieć możliwość kreowania konstrukcji typu zmienne, pętle i instrukcje warunkowe w oparciu o język w pełni współpracujący ze standardem Transact-SQL.
72. SBD musi umożliwiać przebudowę indeksów i wprowadzanie zmian w schemacie bazodanowym w trybie online, umożliwiając innym użytkownikom na dostęp odczytu danych modyfikowanych obiektów.

II.3.3. Wbudowane w SBD narzędzie ETL musi dodatkowo spełniać poniższe wymagania poprzez wbudowane mechanizmy:

1. Narzędzie ETL musi mieć możliwość ładowania danych z użyciem bezpośrednich mechanizmów dostępu, co najmniej do następujących platform baz danych: MS SQL, MySQL, Oracle SQL, PostgreSQL oraz źródeł danych udostępniających dane w standardzie ODBC.

str. 17



2. Narzędzie ETL musi posiadać funkcjonalność pobierania danych z użyciem protokołu FTP.
3. Narzędzie ETL musi posiadać funkcjonalność pobierania danych z udostępnionych w sieci lokalnej zbiorów danych w postaci plików płaskich w formatach (".xls(x)", "csv", "txt", "xml", "json")
4. Narzędzie ETL musi mieć możliwość weryfikacji sum kontrolnych CRC lub MD5 przetwarzanych plików danych.
5. Dostarczone narzędzia ETL muszą posiadać interfejs graficzny, umożliwiający projektowanie procesów czyszczenia danych bez konieczności pisania kodu.
6. W zakresie procesów czyszczenia i przetwarzania danych narzędzia ETL muszą umożliwiać wykorzystanie języka czwartej generacji oraz muszą zapewniać interfejs programistyczny do pisania kodów czwartej generacji, podpowiadający składnię języka.
7. Narzędzia ETL muszą zawierać algorytmy do parsowania, standaryzacji, deduplikacji danych. Wspomniane reguły czyszczenia muszą być możliwe do wykorzystania przez:
 - a. narzędzia do zarządzania jakością danych,
 - b. procesy przetwarzania i integracji danych.
8. Narzędzia ETL muszą posiadać mechanizm umożliwiający raportowanie z procesów czyszczenia danych (m.in. liczba duplikatów, procent wartości brakujących, dedykowane wskaźniki jakości).
9. Narzędzie ETL musi zawierać zestaw gotowych transformacji. Transformacje muszą być dostępne dla użytkownika, jako komponenty graficzne. Parametryzacja transformacji ma odbywać się w sposób graficzny lub poprzez wypełnianie odpowiednich kreatorów. Użytkownik ma mieć możliwość budowy procesu przetwarzania poprzez łączenie transformacji w procesy (bez konieczności pisania kodu).
10. Narzędzie ETL musi zawierać gotowe, parametryzowane transformacje umożliwiające, co najmniej:

str. 18



- a. import i eksport plików płaskich w formatach (".xls(x)", ".csv", ".txt", ".xml", ".json" itp.);
 - b. zdefiniowanie połączenia do źródła danych (np. do bazy danych);
 - c. pobranie danych z baz źródłowych do bazy danych rozwiązania;
 - d. łączenie zbiorów;
 - e. agregowanie danych w tym wyznaczanie statystyk dla kolumn analizowanych;
 - f. tworzenie kolumn wyliczanych;
 - g. zmianę formatu danych (numeryczne, tekstowe, daty);
 - h. walidację danych z wykorzystaniem tabel słownikowych;
 - i. filtrowanie i sortowanie zbiorów;
 - j. dołączenie wielu
11. Narzędzie ETL musi umożliwiać tworzenie przez użytkownika własnych transformacji, które będą mogły być wykorzystywane przez projektantów procesów ETL, na tych samych zasadach, co transformacje standardowe.
12. Narzędzie ETL musi umożliwiać pracę w formie zintegrowanego, graficznego środowiska, które nie wymaga przełączania się pomiędzy różnymi aplikacjami.
13. Narzędzie ETL musi umożliwiać:
- a. tworzenie parametryzowanych procesów ETL
 - b. wykorzystanie istniejącego (wbudowanego lub utworzonego) procesu, jako szablonu bezpośrednio w narzędziu ETL, co oznacza, że musi istnieć możliwość prostej podmiany danych źródłowych, struktur docelowych i zmiany reguł przetwarzania dla procesu bez konieczności tworzenia procesu od nowa.

str. 19



14. Narzędzie ETL musi umożliwiać kolejgowanie procesów i uruchamianie ich zarówno w trybie interaktywnym jak i w trybie wsadowym. Tryb wsadowy musi umożliwiać uruchamianie procesu (również całej kolejki procesów) zgodnie z ustalonym harmonogramem oraz musi umożliwiać uzależnienie uruchamiania od wystąpienia konkretnego zdarzenia (uruchamianie warunkowe).
15. Narzędzie ETL musi umożliwiać sygnalizację wystąpienia błędów w procesach ETL.
16. Narzędzie ETL musi posiadać mechanizm umożliwiający raportowanie stanów i wyników procesów ETL ze szczególnym uwzględnieniem ewentualnych błędów w procesach. Tworzony w narzędziu raport musi zawierać, co najmniej:
 - a. czas wykonania poszczególnych zadań z możliwością rozbicia na poszczególne kroki procesów ETL;
 - b. informacje o błędach i ostrzeżeniach w poszczególnych zadaniach.
17. Narzędzie ETL musi umożliwiać wersjonowanie procesów przetwarzania.

II.3.4. Wbudowane w SBD narzędzie raportowo-analityczne musi dodatkowo spełniać poniższe wymagania poprzez wbudowane mechanizmy:

1. Narzędzie raportowo-analityczne musi zapewniać możliwość automatycznej generacji raportów predefiniowanych oraz tworzonych / zamawianych przez Użytkowników na podstawie określonego harmonogramu.
2. Narzędzie raportowo-analityczne, w ramach standardowej pracy użytkownika z raportem, musi umożliwiać wykonanie, co najmniej następujących czynności:
 - a. sortowanie danych (z uwzględnieniem polskich znaków),
 - b. filtrowanie danych,

str. 20



- c. formatowanie warunkowe w oparciu o wartości danych,
 - d. dodawanie własnych wielkości wyliczanych na bazie operacji arytmetycznych wykonywanych na danych zawartych w raporcie,
 - e. dodawanie i usuwanie podsumowań w raportach,
 - f. synchronizacja obiektów na raporcie (np. powiązanie poziomu szczegółowości prezentowanych danych w tabeli i na wykresie w obrębie jednego raportu).
3. Narzędzie raportowo-analityczne, w ramach standardowej pracy użytkownika z raportem, powinno umożliwiać wykonanie następujących czynności:
- a. rozwijanie/drażenie danych według zdefiniowanych hierarchii (drill down),
 - b. rozwijanie/drażenie danych według ścieżki wybranej przez użytkownika (drill across),
 - c. zmiana wyglądu m.in. zmiana kolorów, zmiana czcionki, zmiana pozycji nagłówków,
 - d. transpozycja raportu tabelarycznego (zamiana wierszy z kolumnami).
4. Narzędzie raportowo-analityczne powinno zapewniać możliwość tworzenia powiązanych raportów. Wybranie konkretnej wartości w raporcie powinno dawać możliwość przekierowania do innego raportu, prezentującego wybrane dane w innym ujęciu (np. pierwszy raport prezentuje dane zagregowane, a drugi raport powiązany prezentuje dane detaliczne dla wybranego przecięcia).
5. Narzędzie raportowo-analityczne powinno umożliwiać tworzenie kokpitów managerskich pozwalających na graficzną i tabelaryczną prezentację wielu wskaźników na jednym ekranie użytkownika, z możliwością drażenia do danych szczegółowych i na odwrót (drill down i drill up) . Drażenie może również polegać na uruchamianiu szczegółowych raportów po

str. 21



kliknięciu na wartość wybranego wskaźnika lub być zrealizowane bezpośrednio w ramach kokpitu.

6. Narzędzie raportowo-analityczne musi zapewniać formatowanie raportów do wydruku w formacie A3 i A4 oraz podgląd wydruku. Narzędzie musi poprawnie wyświetlać polskie znaki diakrytyczne oraz musi zapewniać prawidłowy wydruk raportów zawierających polskie znaki diakrytyczne.
7. Narzędzie raportowo-analityczne musi zapewniać wydruk raportów oraz ich podgląd zgodnie z wymaganiami WYSIWYG.
8. Narzędzie raportowo-analityczne musi zapewniać możliwości tworzenia raportów przy pomocy narzędzi dostępnych z poziomu przeglądarki internetowej.
9. Narzędzie raportowo-analityczne musi umożliwiać tworzenie raportów przy wykorzystaniu interfejsu programistycznego. Dostęp do danych w Hurtowni Danych musi odbywać się, co najmniej za pomocą języka SQL.
10. Narzędzie raportowo-analityczne musi umożliwiać tworzenie i wykorzystywanie szablonów raportów.
11. Narzędzie raportowo-analityczne musi zapewniać możliwość tworzenia raportów i zestawień w oparciu o warstwę metadanych biznesowych (wykorzystanie semantyki biznesowej bez konieczności odwoływania się do fizycznych struktur danych).
12. Narzędzie raportowo-analityczne musi umożliwiać tworzenie raportów poprzez łączenie danych przechowywanych lokalnie (np. w arkuszu MS Excel, pliku tekstowym) z danymi dostępnymi w Hurtowni Danych.
13. Narzędzie raportowo-analityczne musi zapewniać możliwość definiowania kolumn wyliczanych przy pomocy kreatora wyrażeń.
14. Narzędzie raportowo-analityczne musi zapewniać możliwość wzbogacania raportów o dodatkowe elementy takie jak stopka, nagłówek, dodatkowe elementy tekstowe (opisy) i graficzne (np. logo). Dodatkowo rozwiązanie musi pozwalać na dodawanie do stworzonego raportu

str. 22



automatycznych pól opisu, zawierających m.in. autora, datę utworzenia / modyfikacji raportu.

15. Narzędzie raportowo-analityczne musi zapewnić możliwość parametryzowania raportów. Dla raportów z parametrami musi istnieć możliwość wyboru elementu z listy wartości. Lista wartości może być predefiniowana lub budowana dynamicznie na podstawie tabeli z danymi. W przypadku zastosowania kilku zależnych od siebie parametrów, lista dostępnych wartości dla pozostałych (zależnych) parametrów musi być automatycznie zawężana na podstawie wcześniej wybranych parametrów.
16. Narzędzie raportowo-analityczne musi zapewniać możliwość prezentacji raportów w postaci tabel, wykresów i wskaźników (KPI). Musi istnieć możliwość umieszczania wielu takich elementów na jednym raporcie na jednej lub wielu stronach / zakładkach.
17. Narzędzie raportowo-analityczne musi umożliwiać prezentację wyników na wykresach, w tym obsługiwać, co najmniej poniższe typy wykresów: słupkowy, kolumnowy, liniowy, kołowy, punktowy, powierzchniowy, pierścieniowy, radarowy, bąbelkowy, konturowy.
18. Narzędzie raportowo-analityczne powinno umożliwiać tworzenie wykresów mieszanych z zachowaniem logiki łączenia typów (np. punktowo-liniowy, kolumnowo-liniowy, kołowo-radarowy itp.)
19. Narzędzie raportowo-analityczne musi umożliwiać prezentację danych przestrzennych na tle wybranej przez użytkownika (i dostępnej w narzędziu) mapy. Funkcjonalność prezentacji danych przestrzennych na mapie nie oznacza wymagania zaawansowanych analiz GIS, a jedynie zapewnienie lokalizowania i prezentacji danych na tle obiektów mapowych.
20. Narzędzie raportowo-analityczne musi posiadać podstawowe funkcjonalności w zakresie realizacji i prezentacji na wynikach analiz statystycznych, w tym, co najmniej funkcje: minimum oraz maksimum, suma, średnia, odchylenie standardowe, licznosc (count) i licznosc unikalna (count distinct), dominanta, mediana.

str. 23



21. Narzędzie raportowo-analityczne powinno również posiadać inne funkcjonalności w zakresie realizacji i prezentacji na wynikach analiz statystycznych:
 - a. obliczanie kwantyli rzędu p , gdzie p jest parametrem wpisywanym przez użytkownika,
 - b. wyświetlanie wykresu dystrybuanty i gęstości prawdopodobieństwa.
22. Narzędzie raportowo-analityczne musi umożliwiać przechowywanie raportów w jednym centralnym repozytorium raportów. Zapisywane raporty mają być widoczne tylko dla osób z odpowiednimi uprawnieniami. Uprawnienia mają być nadawane zarówno dla konkretnych (indywidualnych) użytkowników jak i grup użytkowników.
23. Narzędzie raportowo-analityczne musi pozwalać użytkownikom na tworzenie i zapisywanie własnych wersji raportów w repozytorium, w którym raporty udostępniane są innym użytkownikom. Użytkownik musi mieć prawo udostępniania stworzonego przez siebie raportu grupie użytkowników lub pojedynczemu użytkownikowi.

II.4. Wymagania dodatkowe

- II.4.1. Wszystkie dostarczone elementy muszą pochodzić z oficjalnego kanału dystrybucji producenta.
- II.4.2. Dostarczone klucze licencyjne muszą zostać przypisane do konta Zamawiającego (konto Gminy Wrocław). Numer konta zostanie przekazany Wykonawcy po rozstrzygnięciu postępowania i wyłonieniu Wykonawcy. Na potrzeby odbioru zostanie zweryfikowane czy na portalu licencyjnym, na koncie Zamawiającego, pojawił się nowy klucz licencyjny na komplet dostarczanych licencji wraz ze wsparciem technicznym.

II.5. Gwarancja, wsparcie, usługi utrzymania i dokumentacja

- II.5.1. Wsparcie musi być świadczone bezpośrednio przez producenta zgodnie z SLA w ramach programu Software Assurance lub równoważnego. Dodatkowo warunki licencji i świadczenia usług producenta muszą być dostarczone w formie elektronicznej.

str. 24

