TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií

# Přechod IT služeb z on-premises na cloudové výpočetní platformy a služby

## Bakalářská práce

*Studijní program:* B2612 – Elektrotechnika a informatika
*Studijní obor:* 1802R022 – Informatika a logistika

*Autor práce:* **Petr Provazník**
*Vedoucí práce:* Ing. Martin Vlasák

Liberec 2018

TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

# Transition CSA IT services from on-premises to the cloud computing platform and service

## Bachelor thesis

*Study programme:* B2612 – Electrical Engineering and Informatics
*Study branch:* 1802R022 – Informatics and Logistics

*Author:* **Petr Provazník**
*Supervisor:* Ing. Martin Vlasák

Liberec 2018

Technická univerzita v Liberci
Fakulta mechatroniky, informatiky a mezioborových studií
Akademický rok: **2017/2018**

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

| | |
|---|---|
| Jméno a příjmení: | **Petr Provazník** |
| Osobní číslo: | **M17000171** |
| Studijní program: | **B2612 Elektrotechnika a informatika** |
| Studijní obor: | **Informatika a logistika** |
| Název tématu: | **Přechod IT služeb z on-premises na cloudové výpočetní platformy a služby** |
| Zadávající katedra: | **Ústav mechatroniky a technické informatiky** |

Zásady pro vypracování:

1. Analyzujte možnosti nabízených cloudových služeb, dostupné technologie a jejich specifika s ohledem na využití v Českých aeroliniích.
2. Vytvořte logický návrh architektury celého prostředí se zaměřením na využití nových cloudových technologií.
3. Prozkoumejte a navrhněte možnosti náhrady IBM Websphere MQ.
4. Zhotovte praktickou realizaci provozu aplikací v cloudovém prostředí s využitím alternativních messagingových nástrojů.

Rozsah grafických prací: **dle potřeby dokumentace**

Rozsah pracovní zprávy: **30–40 stran**

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:


[1] **ATI Cloud. [cit. 2017-10-09], dostupné z: https://www.sita.aero/solutions-and-services/ati-cloud**

[2] **Enterprise Cloud Computing. [cit. 2017-10-09], dostupné z: https://cloud.oracle.com/home**


Vedoucí bakalářské práce: **Ing. Martin Vlasák**

Ústav mechatroniky a technické informatiky

Datum zadání bakalářské práce: **10. října 2017**

Termín odevzdání bakalářské práce: **14. května 2018**

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2017

# Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 18.5.2018

Podpis:

# PODĚKOVÁNÍ

Děkuji panu Ing. Martinu Víchovi Vlasákovi za odborné vedení práce, věcné připomínky, dobré rady a vstřícnost při konzultacích a vypracování bakalářské práce.

# Abstrakt

Cílem této práce je zmapovat možnosti využití cloudových služeb jako náhrada on-premises infrastruktury pro provozování systémů letecké společnosti a jejich realizaci. Nalézt optimální řešení náhrady stávajícího messaging řešení IBM MQ a využít dostupných technologií pro zajištění bezpečného provozování 3 vrstvých aplikací v cloudovém prostředí.

# Klíčová slova

Cloud, on-premises

# Abstract

The aim of this work is to search possibilities of using cloud services as a substitute of on-site infrastructure for the operation of airline systems. To find the optimal solution for replacing existing IBM MQ messaging solution and use the technologies available to ensure the safe operation of three layer applications in a cloud environment.

# Keywords

Cloud, on-premises

# Obsah

**Seznam obrázků**

**Seznam použitých zkratek a symbolů**

MS Microsoft

IaaS Infrastruktura jako služba (např. virtuální server, storage, apod.)

PaaS Platforma jako služba (např. SQL databáze, webový server)

SaaS Software jako služba (např. CRM, ticketing system, apod.

SOA Servisně orientovaná architektura

IOPS Vstupní/výstupní operace za sekundu

SLA Service level agreement, smlouva mezi poskytovatelem a odběratelem služby definující podmínky poskytované služby

VPN Virtuální privátní síť

VNET Microsoft Azure virtuální síť

IPSec Bezpečnostní rozšíření IP protokolu

DMZ Z bezpečnostních důvodů oddělená podsíť

# 1 Úvod

Cloudové služby otevírají z pohledu zákazníka nové možnosti optimalizace nákladů na IT infrastrukturu Společnosti.

Společnost prochází změnou, kdy si musí vybudovat zcela nové IT prostředí. Což do jisté míry je výhodou, neboť nemusí být svázána historickými závazky. Tento fakt je jedinečnou příležitostí pro využití cloudových služeb v takovém rozsahu v jakém pravděpodobně ještě v globálním měřítku žádná obdobná společnost stejného odvětví neimplementovala.

Důvodem pro výběr tohoto tématu je příležitost k ověření skutečných možností cloudových služeb v praxi.

# 2 Analýza a popis stávajícího prostředí

## 2.1 Identifikace a popis aplikací z pohledu užitých technologií

K tomu abychom mohli docílit optimálního návrhu nové architektury pro provoz systému užívaných ve Společnosti je nejprve nezbytné identifikovat software z pohledu technologií a popsat vnitřní i vnější vazby.

Technologie, na které se popis zaměřuje, jsou především databázové platformy, operační systém, aplikační server případně jiné zvláštní užité technologie. Analýzou stávajícího prostředí byly identifikovány aplikace uvedené v příloze č. 1.

V souhrnu lze uvést, že stávající prostředí z pohledu využitých technologií tvoří:

- Operační systém: Microsoft Windows Server, RedHat Linux, AIX
- Databáze: Oracle Database 11i, MS SQL Server
- Aplikační server: GlassFish, Oracle Application server
- Ostatní technologie: IBM Websphere MQ

## 2.2 Vazby mezi aplikacemi

Tak jak provoz společností v daném odvětví vyžaduje a v posledních letech i přinášející ekonomický tlak na maximální efektivitu, jsou identifikované systémy navzájem široce integrované.

Interní analýzou vytvářím model popisující vzájemné vztahy mezi interními aplikacemi, aplikacemi a službami třetích stran.

# 3 Průzkum a analýza cloudových služeb

## 3.1 Cloudové služby a jejich členění

V dnešní době hojně užívaný pojem cloud v našem případě považujeme za datové centrum, jenž je z pohledu uživatele, tedy Společnosti nezávislé na lokaci. Poskytovatel cloudových služeb je zodpovědný za jeho vysokou dostupnost a maximální škálovatelnost.

### 3.1.1 Servisní model (service model)

#### IaaS[1]

Infrastruktura jako službu - zákazník si pronajímá infrastrukturu, kde platí za čas a výkon podle toho, jak ji využívá. Nabízenou infrastrukturu můžeme rozdělit do tří kategorii

- Compute, kde zákazník obvykle platí dle výkonu procesoru za čas, kdy jej využívá. To znamená v případě virtuálního stroje za čas, kdy je tento stroj spuštěn.
- Storage, kde zákazník si objednává a platí za diskové prostory
- Network, kde zákazník si objednává síťové služby jako VPN, VNET, load balancer, apod.

#### PaaS[2]

Platforma jako služba – zákazník pronajímá prostředí, middleware, v němž může vyvíjet, spravovat a spouštět své aplikace bez jakékoliv znalosti a nutnosti přípravy a následné údržby potřebné infrastruktury.

- Bežící prostředí
- Databáze
- Web server
- Vývojářské nástroje

[1]Infrastructure as a Service

[2]Platform as a Service

### SaaS[3]

Software jako služba – zákazník využívá aplikací, resp. služeb, které plně spravuje a instaluje poskytovatel cloudových služeb. Takto provozovaná aplikace, služba je dostupná uživatelům přes internet, případně nějakém IPSec tunelem. Zákazníkovi tak odpadá potřeba jakékoliv správy, infrastruktury, middleware až samotnou instalaci a provoz aplikace.

- CRM
- Email
- Virtual desktop
- Communication

### DaaS[4]

Data jako služba. Je odnoží SaaS.

Na platformě nezávislé poskytování dat zákazníkům prostřednictvím webových služeb a SOA[5]

## 3.1.2 Model nasazení (deployment model)

Jiným druhem členění cloudových služeb je dle formy nasazení.

### Privátní cloud

V případě privátního cloudu jde o prostředí připravené a provozované pro konkrétního zákazníka. Nedochází ke sdílení žádných služeb.

### Veřejný cloud

Jedná se prakticky o opak privátního cloudu, kde dochází k maximálnímu zefektivnění dostupných zdrojů jejich sdílením několika zákazníky.

---

[3]System as a Service

[4]Data as a Service

[5]Service oriented architecture

### Hybridní cloud

Jedná se o kombinaci privátního a veřejného cloudu, kde zákazník si může vybrat zda-li chce využívat pro něho dedikované zdroje či bude využívat vybraných sdílených služeb samozřejmě za výhodnějších finančních podmínek.

## 3.2 Výhody cloudových služeb

Klient objednávající si službu nemusí být znalý hardwaru a softwaru, z kterého je daná služba poskytovaná.

Služby v cloudovém prostředí jsou obvykle provozovány v režimu vysoké dostupnosti, který minimalizuje rizika nedostupnosti potřebné funkcionality.

Široká škálovatelnost v podání cloudových služeb nabízí rozšíření funkcí nebo zvýšení nebo naopak snížení výkonu, např. počtem užitých procesorů, téměř v reálném čase.

Efektivní využití sdílením hardwarových a softwarových prostředků obvykle přináší po ekonomické stránce možnost snížení provozních nákladů za infrastrukturu.

Vysoký důraz se u významných poskytovatelů cloudových služeb klade na bezpečnost, ať už jde o zabezpečení datového centra jako takového nebo o rozsah služeb, kterými si může uživatel zabezpečit svá data, aplikace či služby.

Uživatel není zatížen starostí životního cyklu hardwaru či softwaru, který je v cloudu nabízen. Odebírá vždy službu v požadovaném rozsahu bez ohledu, na jakém hardwaru zrovna běží.

Flexibilita umožňuje uživateli promptně reagovat na jeho potřeby při implementaci nových služeb v rámci společnosti nebo naopak při jejich redukci.

Globální síť datových center nabízí možnost dostupnosti služeb v požadované kvalitě po celém světě.

## 3.3 Nevýhody cloudových služeb

### Dislokace provozovaných aplikací a dat.

Za určitých okolností by fakt několik se stovek či tisíc kilometrů vzdálených uložených dat znamenal problém. Pakliže se nejedná o mandatorní požadavky, je toto s ohledem na možnosti dostupnosti datových center skutečně minimální problém.

### Obava ze zneužití dat.

Takovéto obavy již dnes u významných poskytovatelů cloudových služeb jsou neopodstatněné. Řada poskytovatelů cloudových služeb přijala mezinárodní normu ISO 27018 týkající se soukromí v cloudu. Rovněž orgány Evropské unie zabývající se ochranou dat uznávají tyto poskytovatele cloudových služeb v souladu s přísnými zákony Evropské unie na ochranu osobních údajů.

### Minimální vliv na režim údržby datového centra

Poskytovatelé cloudových služeb jsou těmi, kteří si určují čas na údržbu. Obvykle uživatele s určitým časovým předstihem upozorňují, že tehdy a tehdy může dojít k výpadku služeb z důvodu údržby. Je na uživateli, aby měl v rámci svého řešení z pohledu architektury navrženou ji tak, aby byla na takovéto případy připravena.

Řešení se nabízí hned několik od rozložení služeb do více datových center, přes různé služby zajišťující řádnou dostupnost, které se rozvíjejí a každou chvílí je nabízena nová služba pro zajištění vysoké dostupnosti.

### Omezené možnosti individuálních řešení

Poskytovatelé cloudových služeb nabízejí svým zákazníkům odběr služeb z předloženého seznamu a zákazník de facto nemá možnost si tyto služby nějakým způsobem upravovat dle jeho konkrétních potřeb. Jako příklad můžeme uvést seznam konfigurací virtuálních strojů. V tomto případě konfigurací virtuálního stroje je myšlen počet jader procesoru a velikost paměti. Zákazník musí vybírat z toho, co je mu nabízeno, ale nemá možnost si tyto parametry individuálně přizpůsobit. Nabídka poskytovatelů je však většinou natolik široká, že ve většině případů si zákazník najde odpovídající konfiguraci.

## 3.4 Dostupnost technologií v cloudových službách

Rozsah dostupných technologií u poskytovatelů se výrazně liší. Lokální poskytovatelé cloudových služeb jsou většinou omezeni na poskytování služeb do úrovně IaaS a nejsou již schopni zajistit potřebné licence například pro provoz větších databází a operačních systémů. Významnější poskytovatelé cloudových služeb se snaží maximálně

rozšiřovat své portfolio nabízených služeb i mimo své vlastní produkty, avšak i zde jsou výrazné rozdíly.

Na úrovni operačních systému jsou obvykle dostupné vybrané distribuce Linux a Microsoft Windows Server.

## 3.5 Specifika cloudových služeb

Mnoho firem nabízí cloudové služby. Od malých firem zde v České republice, kde se i tento obor získává na popularitě, hlavně díky dotacím z Evropské unie až po nadnárodní společnosti jako je Google, Amazon, Microsoft, SITA, Oracle.

V našem případě se s ohledem na portfolio nabízených služeb, úroveň podpory a předpoklad rozvoje zaměřím na poslední tři jmenované společnosti.

### 3.5.1 SITA ATI Cloud

SITA ATI (air transport industry) Cloud nabízí cloudové prostředí specializované pro provoz společnost leteckého provozu. Jejich cloudové řešení je však významně zaměřeno na služby typu PaaS a SaaS, s využitím produktů SITA.

### 3.5.2 Microsoft Azure

Microsoft Azure, nabízí v cloudovém prostředí spojení vlastních technologií s technologiemi třetích stran. Prostředí Microsoft Azure se natolik dynamicky rozvíjí, že výčet dostupných služeb již za měsíc nebude aktuální.

Společnost Gartner v rámci hodnocení Magic Quadrants za poslední roky označila Microsoft Azure za lídra cloudových služeb.

Prostředí MS Azure nabízí možnost vysoké škálovatelnosti v reálném čase. Je možné definovat zvýšení výkonu na základě zvýšených parametrů čerpaných prostředků nebo je možné je předem plánovat dle určitého časového schéma. Příkladem může být zvýšení výkonu v ranních hodinách, kdy do práce přicházející zaměstnanci si spouští a pracují s aplikacemi a naopak po pracovní době výkon utlumit pouze na takovou úroveň, aby byla služba dostupná.

Další nespornou výhodou MS Azure je možnost integrace s vlastním on-premises prostředím pokud je postaveno na platformě Microsoft a v neposlední řadě sdílené služby s Office 365.

### 3.5.3 Oracle Cloud

#### 3.5.3.1 IaaS

V infrastrukturální částí cloudových služeb nabízí Oracle Cloud 2 základní metriky k provisioningu výpočtového prostředí – sdílené a dedikované.

V případě dedikovaných virtuálních strojů jsou výpočetní zdroje izolovány pro konkrétní stroj a je možné s tímto strojem navázat site-to-site VPN z on-premises prostředí. Naproti tomu v případě sdíleného výpočetního výkonu jsou zdroje rozkládány mezi více tenanty.

V oblasti datových úložišť Oracle Cloud nabízí a rozlišuje

Storage Cloud Service - obecné úložiště dostupné odkudkoliv z internetu

Storage Cloud Archive Service – úložiště designované pro velká data s častým přístupem.

Shared File Storage Service – datová úložiště vhodná pro sdílení mezi více virtuálními instancemi v Oracle Cloudu

Storage Cloude Software Appliance – vhodné řešení data přístupná v on-premises uložená však v Oracle Cloudu

Network

V oblasti síťových služeb Oracle Cloud nabízí možnost navázání VPN spojení, přičemž data jsou přenášena zabezpečeně internetem. Jelikož se jedná o přenos přes internet není možné garantovat parametry rychlosti, odezvy a celkové dostupnosti navázané spojení.

FastConnect je řešení , které oproti běžnému VPN tunelu redukuje rizika a přináší u záruku vyšší rychlosti, nižších latencí a dostupnosti. Prakticky se jedná o konektivitu na úrovni standardní síťové vrstvy s BGP routingem.

#### 3.5.3.2 PaaS

Nabídka Oracle Cloud v případě databázových služeb vychází portfolia Oracle Database, aktuálně ve verzích 11g nebo 12c, přičemž nabídka umožňuje vybírat z databáze běžící na dedikovaném stroji, plně funkční databáze běžící na Exadata nebo pouze 1 schéma ve sdílené databázi s omezenou velikostí (5, 20 nebo 50 GB).

Z oblasti messagingu nabízí Oracle Cloud standardní interface umožňující komunikaci s využitím standardních JMS API, REST API případně přenos zpráv přes HTTP protokol prostřednictvím webových služeb.

# 4 Návrh nové architektury prostředí IT

## 4.1 Výběr poskytovatele cloudových služeb

S ohledem na šíři portfolia služeb, jež Microsoft nabízí v podobě svých cloudových služeb Azure byl zvolen jako vhodných poskytovatel.

Rozhodujícím faktorem byla při výběru aktuální cena poskytovaných služeb, široká nabídka poskytovaných služeb, díky které je prakticky možné maximálně po drobných úpravách až na jedinou výjimku přenést, včetně zalicencování s Office 365, které je mimo rozsah tohoto projektu, všechny dosavadní užívané aplikace.

## 4.2 Návrh architektury v cloudovém prostředí

V kapitole 2.1 jsem identifikoval stávající užívané technologie. Z uvedeného seznamu vím, že aktuálně v portfoliu MS Azure nenalezneme unixové operační systémy. Je tedy nutné v rámci návrhu nové infrastruktury řešit náhradu za operační systém AIX a Solaris. Dále v portfoliu nenalezám linuxové distribuce RedHat, Oracle Application server (nově pouze verze WebLogic), Microsoft Websphere MQ jejichž náhradu je rovněž třeba řešit.

V průběhu analýzy vzniká potřeba provozování on-premis systému s databází Oracle. Část modulů bude provozována čistě v interní síti, avšak jeden z modulů bude muset být publikován do internetu a tudíž bude muset být provozován v DMZ. V tomto případě budeme řešit přístup k datům z DMZ, k čemuž jako vhodný nástroj identifikuji Service Bus v módu Relay.

### Náhrada AIX a Solaris

Konstatuji, že všechny provozované služby a aplikace na operačních systémech AIX a Solaris, až na jednu výjimku, bude možné provozovat operačních systémech MS Windows Server případně CentOS[6]. Na unixových operačních systémech jsou povětšinou provozovány DB Oracle, kterou je možné provozovat na MS Windows Server případně JAVA aplikace, které je možné zprovoznit jak na MS Windows Server, tak hlavně na

[6]Linuxová distribuce založena na Red hat Enterprise Linux komunitou vývojářů v roce 2004.

linuxovém operačním systému CentOS, který je zároveň svou kompatibilitou vhodnou náhradou za dosud provozovaný linuxový operační systém RedHat.

## Výběr a popis klíčových technologií služeb v prostředí MS Azure

### Operační systém

Jak už bylo zmíněno výše, hlavní platformou z pohledu operačních systému bude tvořit MS Windows Server 2012 a linuxová distribuce CentOS 7.1

### Availability set

Je významnou službou MS Azure nad 2 a více virtuálními servery, která zajistí maximální dostupnost v případě neočekávaných výpadků nebo údržbě některého z těchto virtuálních serverů.

### Security group

Je službou MS Azure, která umožňuje zabezpečit komunikaci směrem z a do virtuálních serveru zařazených do konkrétní Security groupy.

### VNET

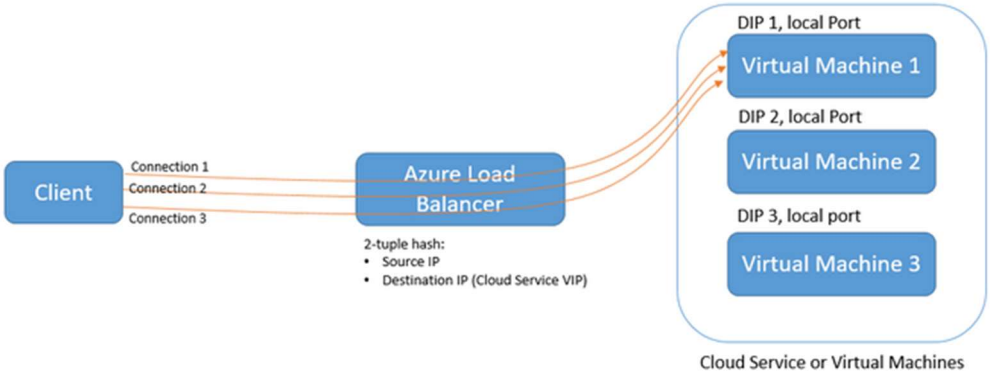
Jedná se prakticky o virtuální síť uvnitř jedné subskripce[7] MS Azure.

### Network load balancing

Služba umožňující rozdělení zatížení mezi více virtuálních strojů, V prostředí MS Azure identifikuji 2 základní typy balancingu z pohledu distribuce dostupnosti:

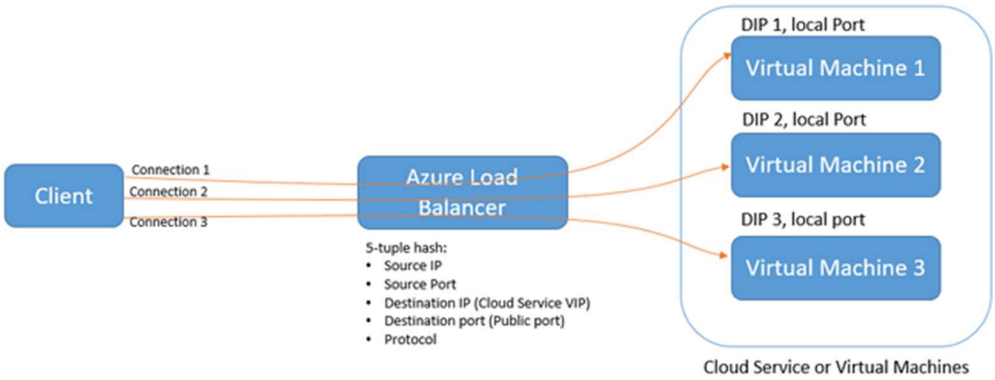
Prvním modelem je typ, který je asi obecně známý z on-premises prostředí a prakticky zajišťuje konektivity z jedné zdrojové IP adresy na jeden konkrétní virtuální stroj z virtuálních strojů zařazených do load balancingu

[7]Subskripce v tomto kontextu představuje prostředí v konkrétním datovém centru konkretního zákazníka




*Obrázek 4-1 MS Azure load balancing mode řízený zdrojovou IP adresou*[8]

Druhým modelem je typ, který distribuuje provoz na základě algoritmu užívajícího pětice klíčových faktorů (zdrojové IP adresy, zdrojového portu, cílové IP adresy, cílového portu a typu protokolu) mezi dostupné servery z celé skupiny load balancingu. Tento model dokáže být maximálně efektivní z pohledu rozložení zátěže mezi dostupné virtuální stroje. Z pozdějších testů však identifikuji, že je provozovatelný pouze i přizpůsobených aplikací. Aplikace, které vyžadují v případě komunikace zajištění kontinuity navázané spojení není provoz pod tím load balancingem možný.




*Obrázek 4-2 MS Azure load balancing mode řízený 5-ticí klíčových faktorů*[9]

[8]https://azure.microsoft.com/en-us/documentation/articles/load-balancer-distribution-mode/

[9]https://azure.microsoft.com/en-us/documentation/articles/load-balancer-distribution-mode/

### Služby ADFS

Služba operačního systému MS Windows Server rozšiřující možnosti Active Directory směrem k webovým aplikacím. Její využití identifikuji pro provozování Office 365 a jako užitečné i nativní rozhraní pro změnu doménového hesla.

## 4.3 Výběr vhodného datového centra

Při výběru vhodného datového centra zohledňuji geografické umístění datového centra s ohledem na geografické umístění uživatelů. V našem případě, v případě tohoto projektu, se budeme zabývat pouze uživateli zde v České republice, konkrétně v Praze. Vybíráme jedno z datových center v Evropě, datové centrum North Europe provozované v irském Dublinu nebo datovém centru West Europe v holandském Amsterdamu. Z pohledu nákladů bychom měli vybrat datové centrum North Europe, avšak dále si ukážeme, že na základě testů bude tím správným datovým centrem West Europe.

## 4.4 Příprava prostředí

### 4.4.1 Vznik nové domény

S ohledem na celkový rozsah projektu, který překračuje obsah této práce pro snadnější přechod na nové prostředí a maximalizaci zajištění business potřeb vzniká nová interní doména, která nahradí stávající užívané domény.

### 4.4.2 Definici VLAN a jejich IP rozsahů

S ohledem na fakt, že nové prostředí vzniká de facto paralelně vedle stávajících prostředí a přechod z jednoho prostředí na druhé bude třeba vydefinovat si strukturu sítě a v ní jednotlivé VLANy taky, aby nedocházelo ke kolizi v IP rozsazích stávajícího on-premises, nového on-premises a Microsoft Azure prostředí.

Definuji IP rozsahy pro VLANy:

- Server segment [10.06.1.0/24]
- User segment LAN [10.06.16.0/23]
- User segment Wifi [10.06.21.0/23]

- Admin user segment [10.06.19.0/24]
- Tiskárny, skenery [10.06.24.0/24]
- FC tablety LAN i Wifi [10.06.23.0/23]
- Wifi user [10.06.25.0/23]
- Wifi admin [10.06.78.0/24]
- PCI Compliance [10.06.89.0/24]
- MS Azure [10.06.200.0/21]

### 4.4.3 On-premises prostředí

Cílem tohoto projektu bylo minimalizovat potřeby na provozování on-premises infrastruktury. Z pohledu infrastruktury budou v on-premises prostředí provozovány 2 fyzické servery s nainstalovaným HypeR-V v clusteru v konfiguraci 8 jader, 64 GB RAM.

Z analýz s ohledem na zajištění dostupnosti služeb a charakter jejich provozu byly identifikovány jako nezbytné nebo vhodné pro on-premises prostředí:

- Doménový řadič (na každém node 1)
- Tiskové řešení – print server, SafeQ
- DHCP server
- DNS
- Radius server
- SafeQ server
- RSA SecureID pro vícefaktorovou autentifikaci point-to-site VPN
- Systém operačního řízení publikovaný do prostředí SITA CUTE sloužící pro odbavení cestujích

### 4.4.4 Azure subscription

- VPN tunel

  Prakticky první co je nutné pro zprovoznění prostředí v cloudu vyřešit je konektivita, tedy navázání nějakého zabezpečeného spojení

Microsoft definuje podporované zařízení pro navázání site-to-site VPN. V našem případě pro pilotní fázi máme k dispozici Juniper SRX řady 3400, který je na tomto seznamu.

V případě, že zařízení není nalezeno na seznamu podporovaných, je možné, že stále VPN spojení site-to-site bude fungovat pokud užívané VPN zařízení splňuje minimální požadavky

Vytvářím VPN spojení prostřednictvím portálu MS Azure Classic.

Na portále na stránce Networks

Prostředí MS Azure ještě rozdělíme do tří subnetů:

- Subnet DMZ

  V rámci prvotní analýzy jsme identifikovali služby, dle charakteru jejich užívaní, které bude nezbytné umístit do prostředí DMZ.

  Takovouto službou je například ftp server, služba ADFS proxy, která zajišťuje komunikaci mezi ADFS a službou Office365

- Subnet Backend

  V prostředí backendu bude provozována převážná část služeb a infrastruktury.

- Subnet pro aplikace třetích stran

  Tento subnet bude určen pro provoz infrastruktury na které budou provozovány a spravovány aplikace třetími stranami.

Tím máme základní předpoklady pro přípravu prostředí splněny.

### 4.4.5 Pilotní testování

Pro pilotní testování byl vybrán klíčový systém postavený na databázi Oracle 11i neboť se jedná o modulární systém složený z tlustých a tenkých klientů a co do užitých technologií jde asi o nejkomplexnější užívaný systém.

### Schéma prostředí vybraného systému v prostředí MS Azure

V rámci provedené předpilotní analýzy jsme vytvořili schéma stávajícího prostředí (viz. příloha 10.2).

### Zprovoznění systému v prostředí MS Azure

Všechny aplikace a interface byly v MS Azure prostředí zprovozněny. Tlustí klienti byly v rámci pilotního testování distribuovány prostřednictvím Microsoft technologií RemoteApp[10] i App-V[11].

Přechod DB Oracle z verze Enterprise 11.2.0.1 na Standard 12.1.0.1 se jeví jako relativně bezproblémový. Obdobně je tomu se změnou operačního systému AIX na CentOS u aplikačního serveru na kterém běží JAVA démon.

Jak však ukázaly prvotní výsledky testování některé aplikace distribuované prostřednictvím App-V a spouštěné na klientské stanici vykazují velmi výrazné hodnoty zpomalení. U aplikací, které se běžně ve stávajícím prostředí startují 5 minut, nyní trvá více než hodinu, než aplikace nastartuje. Tuto charakteristiku nevykazují aplikace distribuované prostřednictvím Remote-App. Jak později zjišťujeme klíčovým faktorem je latence, neboť aplikace distribuované přes Remote-App běží fakticky na serveru v MS Azure prostředí a na klientskou stanici je distribuováno pouze její zobrazení naproti tomu aplikace distribuované přes App-V plnohodnotně komunikují s aplikačním a databázovým serverem z klientské stanice neboť tam aplikace fakticky běží. Více si problematiku latence rozebereme v další kapitole.

### Závěr z testování

Zaznamenané problémy s latencí jsou tak zásadní, že prakticky znemožňují bez provedení změn provoz systému v prostředí MS Azure.

Pro minimalizaci dopadu latence je nezbytné:

- Za datové centrum zvolit to, které je geograficky nejbližší. Například uvažujeme-li o komunikaci mezi datovými centry optikou při vzdušných vzdálenostech

[10]RemoteApp je způsobem distribuce aplikace prostřednictvím technologie vzdálené plochy

[11]App-V technologie distribuce aplikací v balíčcích na koncové stanice.

Praha-Dublin = 1467 km, Praha-Amsterdam=711, při zanedbání vlivů aktivních prvků, lomení signálu v optickém vlákně a útlumech, čistě s teoretickou hodnotou rychlosti světla je rozdíl mezi Amsterdamem a Dublinem 2 desítky ms.

- Problematické aplikace distribuovat prostřednictvím Remote-App nikoliv App-V
- U aplikací, které není možné jistých důvodů distribuovat přes RemoteApp, čímž může být například nezbytnost provozu aplikace v offline režimu, je nutné provést změny v kódu tak, aby se minimalizovala četnost komunikace mezi klientskou stanicí a cloudovým prostředím.

### 4.4.6 Kompletní testování

Před kompletním testováním je určeno několik testovacích skupin, přičemž každá z nich má na starosti přípravu a otestování konkrétní aplikace či konkrétní služby v prostředí MS Azure.

Připravujeme harmonogram a s ohledem na počet aplikací víme, že týmy budou testovat a vzájemně muset spolupracovat díky integraci jednotlivých aplikací po dobu cca 4 měsíců.

Je nutné mít k dispozici prostředí, myšleno provozuschopnou infrastrukturu a nezbytné služby a proto připravuji skripty pro vytvoření aplikačních serverů, definic služeb a nastavení klíčových parametrů. Příklady vybraných skriptů jsou uvedeny dále v příloze č. 10.3.

V rámci kompletního testování si rovněž ověřujeme možnost náhrady technologie IBM MQ technologií MS Biztalk a Azure Service Bus. Jak se později ukazuje, náhrada technologie bude možná, avšak bude v této oblasti třeba dořešit více detailů, které nebude možné zrealizovat v rámci testování a budou muset být řešeny až při nasazování konkrétních služeb.

### 4.4.7 Vlastní migrace aplikací

Jako první aplikaci k zmigrování vybíráme, stejně systém jenž byl vybrán pro pilotní testování.

Vlastní migraci předchází nezbytné přípravy, migrační plán, potřebná komunikace a samozřejmě stanovení termínu.

Přípravy jsou s ohledem na fakt, že se jedná o systém, který nevyužívají pouze interně, ale i jiné partnerské společnosti. Pro autentifikaci do systému se využívají doménové účty, je tedy nezbytné je v nové doméně vytvořit, nastavit uživatelům inicializační heslem připravit prostředí v kterém si je budou moci aktualizovat a o tomto je informovat. Rozhraním, pro změnu hesla je využito nativní služby ADFS, která umožňuje prostřednictvím webové stránky změnu doménového hesla.

Migrace s ohledem na provoz musí být plánována na noc mezi půlnocí a pátou hodinou, kdy je letový provoz nejslabší. V tomto čase je nezbytné provést export původní databáze, po zakomprimování její přenos do datového centra v Amsterdamu, po rozbalení importu, provedení potřebných změn v přesměrování messagingu do nového i starého prostředí pro případ návratu, nastartování všech potřebných služeb a informování uživatelů o spuštění systému v novém prostředí.

Úspěšná migrace systému AVES je vzorem pro migraci dalších systému.

## 4.5 Identifikace možných rizik

### 4.5.1 VPN přes internet

Jelikož je spojení s prostředím Microsoft Azure v rámci testování realizováno prostřednictvím navázaného VPN spojení přes internet není možné co do výkonu a dostupnosti garantovat téměř nic. Toto je fakt, který v případě cloudových služeb obvykle musí zákazník akceptovat. V případě Microsoft Azure však existuje řešení, které si zmíníme v kapitole Express route (kapitola 5.1.2)

### 4.5.2 Latence

Jak jsme si ověřili v rámci pilotního testování, latence je významným faktorem, který musí uživatel cloudových služeb zohlednit.

V případě pilotního testování byla naměřena průměrná hodnota latence mezi klientskou stanicí a virtuálním strojem s v datovém centru v Dublinu 40 ms. Tato hodnota významně ovlivňuje provoz aplikací, kde dochází k časté komunikaci.

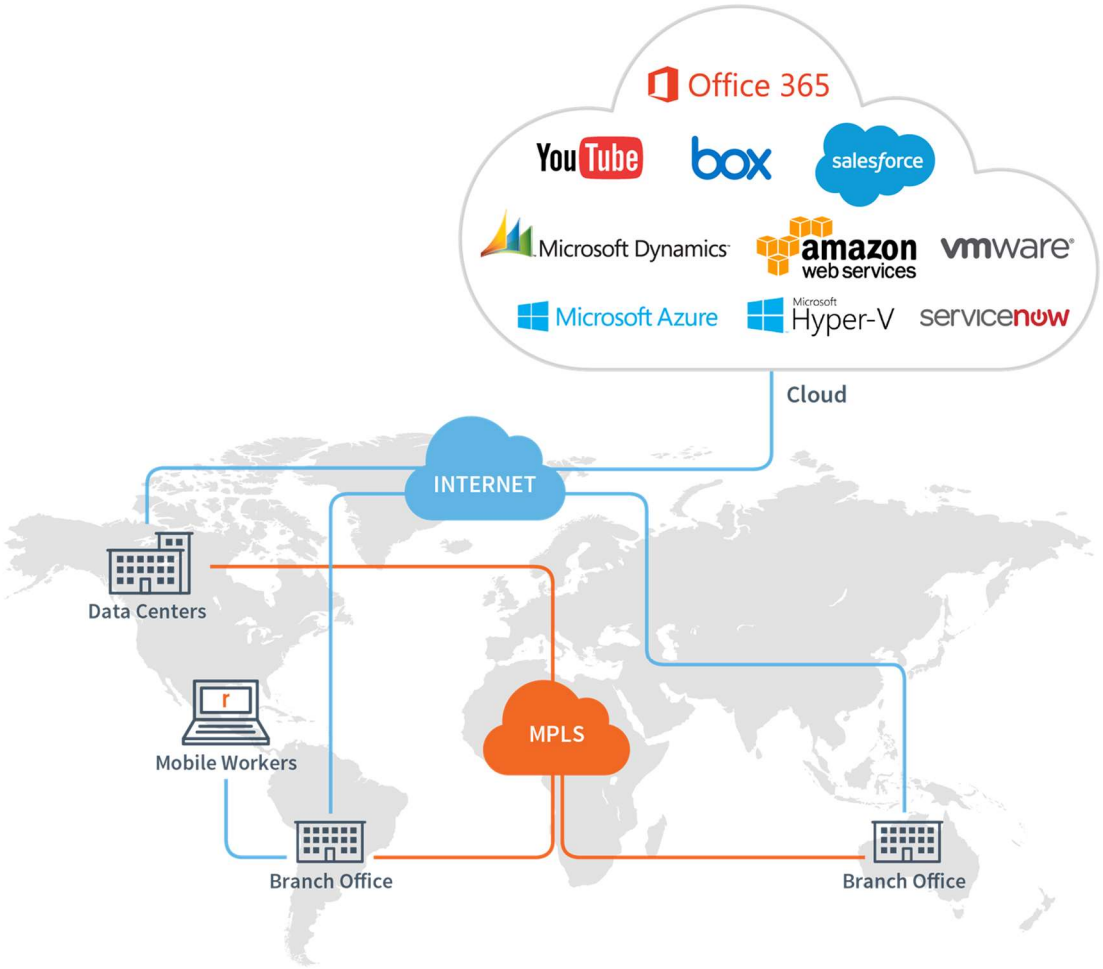
Latence má zásadní vliv na výkonost aplikací a to tak, že v jistých případech aplikace může být de facto nefunkční. Pokud se například při startu aplikace dotazuje do databáze

na 100.000 záznamů, jeden za druhým neboť na aplikační straně probíhá nějaké zpracování při latenci 40 ms bez ohledu na další vlivy pouze tato část akce se bude vykonávat 1h07m06s což může být z pohledu uživatele neakceptovatelné. Z pohledu aplikace je tedy při jejím návrhu s tímto technickým omezením počítat a kód aplikace pokud možno psát tak, aby byl omezen v maximální možné míře počet dotazů a odpovědí mezi klientskou stanicí a serverovou částí v cloudovém prostředí což může být aplikační server nebo databáze a přenášel se méně často, ale větší objem dat čímž se i využije celá šířka přenosového pásma.

Existují softwarová i hardwarová řešení, která dokáží v některých případech problémy s vysokou latencí eliminovat nebo alespoň snížit jeho dopad. Takovým příklad může být řešení společnosti RiverBed, která je i dle Gartner jedničkou na trhu co WAN optimizace.


*Obrázek 4-3 Magický kvadrant pro WAN optimalizaci*[12]

[12]http://www.riverbed.com/gb/solutions/wan-optimization.html

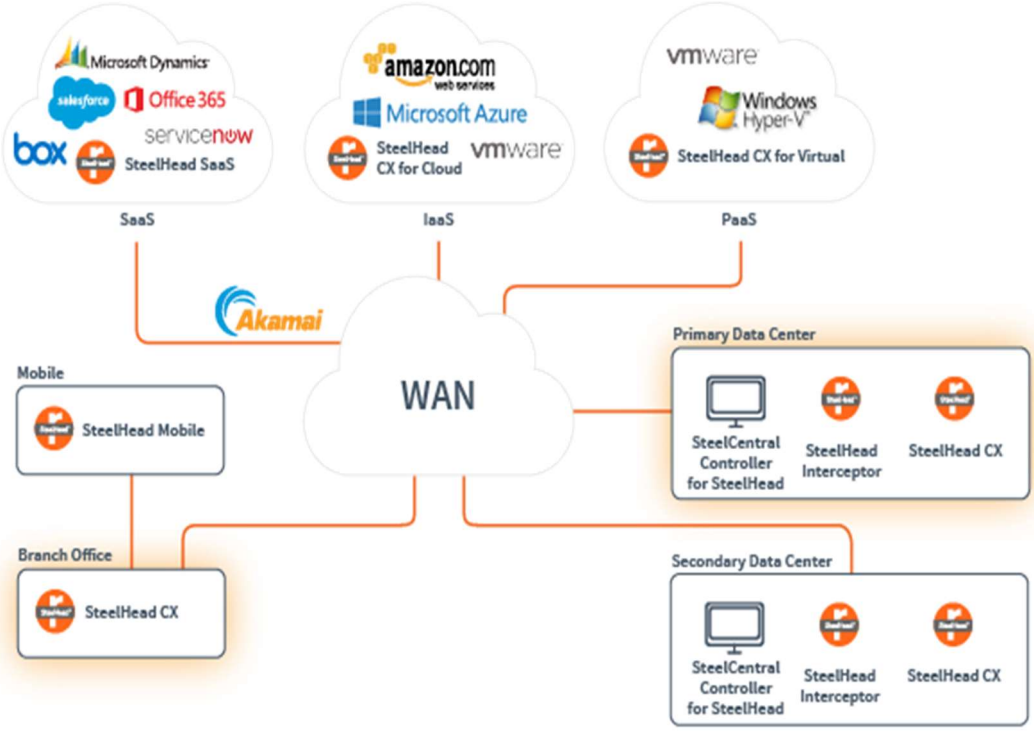



*Obrázek 4-4 WAN Riverbed optimalizace cloudových řešení*[13]

Riverbed SteelHead nabízí optimizační řešení pro akceleraci komunikace aplikací prostřednictvím hybridní WAN. Riverbed nabízí několik produktů z řad SteelHead a SteelCentral. Z našeho pohledu může být zajímavé řešení SteelHead CX a SteelHead SaaS

Riverbed CX – řešení jenž akceleruje přenos dat a aplikací přes hybridní WAN, kterou může být jak privátní VPN, tak veřejný internet.

[13]http://www.riverbed.com/gb/solutions/wan-optimization.html




*Obrázek 4-5 Schéma řešení Riverbed Stellhead CX*[14]

## Riverbed Steelhead CX

- Při optimalizaci dokáže identifikovat a rozlišit více než 1300 aplikací, ať už jde o on-premises či SaaS.

- Optimalizuje náročné protokoly jako SSL/TLS využitím vysoce výkonného hardware běžícího na SSD

- Cachování je jednou z dalších metod, kterou dokáže redukovat latenci.

## Riverbed Steelhead SaaS

Produkty řady SteelCentral nabízejí uživateli rozsáhlý monitoring díky němuž může rychle a efektivně identifikovat nejkritičtější oblasti z pohledu výkonosti aplikací,

## Testování

K praktickým testům nakonec nedošlo, neboť předběžnou analýzou charakteristiky problematické komunikace byl identifikován potenciálně relativně nízký vliv na optimalizaci komunikace, díky jejich problematické predikci a provozní režii. Dalším

[14]http://www.riverbed.com/gb/products/steelhead/steelhead-cx.html

důvodem byla optimalizace aplikací na úrovni kódu, která se pozitivně projevila v chování aplikace až tak, že negativní vliv latence byl uživatelsky nezaznamenatelný.

### 4.5.3 Omezený výkon I/O storage

Každá kategorie disku v cloudovém prostředí Microsoft Azure má své definované limity pro I/O operace. Tento faktor je pro výkonost složitějších systémů klíčový a jak se v průběhu testování ukazuje i dosti limitující.

Zjednodušeně řečeno, standardní storage s ohledem na počty IOPSů je dostačující pro jednoduché webové aplikace či na diskovém výkonu nezávislé aplikace. Co do výkonosti Premium storage jsou hodnoty již výrazně lepší, ale rovněž se to projevuje na nákladech.

Do určité míry omezení IOPSů lze kompenzovat svázáním více než jednoho disku ke konkrétnímu VM čímž dochází ke sčítání dostupných IOPSů jednotlivých disků.

# 5 Realizace provozu aplikací v cloudovém prostředí

## 5.1 Provoz aplikací v MS Azure cloudu

### 5.1.1 Premium Storage vs. Standard Storage

V průběhu implementace aplikací do cloudového prostředí zjišťuji, že pro provoz komplexnějších systémů je téměř nezbytné, z důvodů limitujících IOPSů, využít Premium Storage.

Další charakteristická vlastnost, která byla identifikována zvýšená latence mezi servery běžící na Premium storage a Standard storage což je pravděpodobně dáno dislokací dané infrastruktury a množstvím aktivních prvků v rámci daného datového centra.

### 5.1.2 Express Route

ExpressRoute před pilotním testováním neexistovala v České republice možnost volby ExpressRoute do datového centra v Amsterdamu. Jediný provider poskytující ExpressRoute v České republice nabízí spojení pouze do datového centra v Dublinu což pomineme-li SLA, tak z pohledu snahy o snížení velikosti latence je irelevantní.

Před rozhodnutím o přechodu do cloudového prostředí již byl znám záměr českých telekomunikačních společností vybudovat ExpressRoute do datového centra v Amsterdamu.

Po fázi implementace Express Route zjišťuji, že připojením přes Express Route došlo cca k 10% snížení latence (cca 16ms Amsterdam), ale hlavně latence je stabilní a nedochází k výkyvům jako v případě připojení přes klasickou VPN užitím čistého internetu.

# 6 Náhrada IBM Websphere MQ

## 6.1 Charakteristika provozu IBM Websphere MQ

V prostředí společnosti je implementován messagingový nástroj IBM Websphere MQ včetně komponenty IBM MQ Broker.

Využívá se k přenosu datových zpráv, souborů mezi interními systémy či interním systémem a systémem externího partnera či hostované služby.

Pro monitoring přenosu zpráv přes MQ byla vyvinuta interní aplikace, která identifikuje zprávu při vstupu do MQ, zaznamenává její předávání mezi frontami až do okamžiku kdy opouští MQ.

## 6.2 Technologie v prostředí MS Azure jako náhrada IBM MQ

Náhradou IBM MQ v prostředí MS Azure můžou být technologie a služby MS Azure Service Bus a MS Biztalk. Pro plnohodnotnou náhradu stávajícího řešení však bude třeba vyvinout řešení, které dokáže zabezpečit plnohodnotný monitoring přenosu zpráv, jejich transformace a dokáže je předávat potřebným kanálům IBM MQ, ftp, smtp.

Na základě analýzy bylo rozhodnuto, že pro potřeby monitoringu a transformace zpráv bude vyvinuta worker role nad Azure Service Bus jako služba v prostředí MS Azure.

Logika předávaní zpráv pro různé typy kanálů s využitím tzv. „obálky“, která sebou nese potřebné informace, bude implementována v MS Biztalk.

## 6.3 Integrace Azure ServiceBus a MS Biztalk

### 6.3.1 Azure ServiceBus

#### Fronty

Pro předávání zpráv do MS Azure Service Bus byla vyvinuta univerzální knihovna, jejiž implementací se standardizuje a usnadní způsob předávání zpráv mezi aplikací a Service Busem

#### Limity

Obrovským limitujícím faktorem pro messaging MS Azure Serice Bus je maximální velikost zprávy 256kB. Na základě analýzy a s ohledem potřeby rychlého nasazení přijatelného řešení byla implementována do univerzální knihovny funkcionalita komprimovaní zpráv. Tímto způsobem se u aktuálně předávaných zpráv dostaneme vždy

pod 220kB čímž s „obálkou“ a dalšími properties splníme limit 256kB. V pozdějších fázích optimalizace služeb bude implementace rozšířena o možnost ukládání obsahu na filesystém a předávání pouze odkazu.

### 6.3.2 MS Biztalk

#### Biztalk Standard vs. Biztalk Enterprise

Standard edice MS Biztalk nabízí možnost nasazení maximálně 5 aplikaci. S ohledem na charakteristiku implementace MS Biztalk v prostředí tento faktor není rozhodující a proto bylo rozhodnuto o verzi Standard.

## 6.4 Azure ServiceBus Relay

### 6.4.1 Azure Service Bus Relay

#### Charakteristika Azure Service Bus Relay

Azure Service Bus Relay podporuje tradiční jednosměrný provoz, provoz typu požadavek/odpověď a provoz peer-to-peer. Taky podporuje distribuci událostí na úrovni internetu, která umožňuje scénáře typu publikování+odběr a obousměrnou socketovou komunikací pro zvýšenou účinnost mezi body.

Azure Relay má dvě funkce:

- Hybridní připojení – Pomocí otevřených webových soketů umožňuje scénáře s podporou více platforem.
- Přenosy WCF – Pomocí technologie Windows Communication Foundation (WCF) umožňuje vzdálená volání procedur. WCF Relay je starší verze nabídky přenosu, kterou již mnozí uživatelé používají ve svých programovacích modelech WCF.

#### DMZ a užití Azure Service Bus Relay

Azure Service Bus Relay v našem případě umožní aktivní komunikaci směrem k backendu z prostředí DMZ či čistě internetu bez nutnosti otevírání jakékoliv příchozí komunikaci na úrovni firewallů směrem do vnitřní infrastruktury Společnosti.

# 7 Závěr

Projekt byl zacílen na přípravu IT prostředí restrukturalizované společnosti za využití nových moderních technologií s cílem optimalizovat využití zdrojů a minimalizovat tak náklady bez nutnosti snižování úrovně dostupnosti, bezpečnosti, funkcionalit a uživatelského komfortu. Měl přinést možnost dynamicky reagovat na potřeby společnosti k zajištění konkurenceschopnosti. K tomuto účely se nabízí využití cloudových služeb.

V první fázi projektu bylo mým úkolem identifikovat potřebné technologie na jejichž základě jsem započal s definicí návrhu řešení celkové architektury. Příprava vlastního prostředí byla dalším logickým krokem, které jsem společně s kolegy započal a kterému musela ještě předcházet příprava business case. Vedle vlastní přípravy prostředí bylo mým úkolem řízení jednotlivých kroků celé implementace, to znamená řízení kolektivu testujícího a implementujícího jednotlivé služby, komunikace uvnitř i mimo společnost s partnery a dodavateli, rozdělení aktivit do několika etap, vytvoření projektového plánu a zajištění dostupnosti zdrojů v potřebný čas.

Již první úkol identifikace užívaných technologií a jejich dostupnost v cloudovém prostředí byl důležitým krokem pro rozhodnutí o vhodném řešení. V identifikovaných, aktuálně užívaných technologiích figuroval i operační systém UNIX. Opuštění operačních systémů AIX, jejich náhrada linuxovou distribucí a změna platformy řešení messagingu byly klíčové faktory, které bylo třeba vyřešit. Jak se později ukázalo, dostupnost UNIXových systému v cloudovém prostředí je problematická. Velcí poskytovatelé cloudových služeb většinou nabízejí ve svém portfoliu operační systémy Windows a různé distribuce Linuxu. Přechodu z UNIXu na Windows či LINUX však nic nebránilo, kromě messagingového nástroje, neboť dodavatelé užívaných systému zajišťují nebo v rámci projektu zajistili podporu o tyto cílové operační systémy.

Po přípravě prostředí a pilotním testování, se ukázalo, že největším a zásadním rozdílem, který může vyústit ve značné problémy je nárůst latence. Tento fakt, se ukázal klíčový pro několik dvouvrstvých aplikací u kterých bylo třeba v rámci projektu, přepsat a optimalizovat část kódu, kde tlustý klient aplikace komunikuje s vysokou frekvencí s databází .

Tento fakt, do jisté míry měla řešit technologie společnosti Riverbed, nicméně úprava software užívaných aplikací se ukázala jako dostačující a nebylo třeba přistoupit k

nasazení takovéhoto řešení. Toto zvažované řešení by případně bylo nasazeno v momentě kdy i kvůli dalším službám typy Office365 bude narůstat množství přenášených dat, kde i podle analýz této technologie se ukazuje její vysoká efektivita díky predikovatelnosti podoby datové komunikace na rozdíl od zákaznických, specifických systémů.

Dnes již je asi možné říci, že tento unikátní projekt byl úspěšný neboť všechny aplikace a systémy u kterých bylo záměrem je přenést do cloudu, tam již běží.

Ikdyž se stálé vynořují nová a nová fakta, nové překážky, tak za pomocí celého týmu, celé společnosti, která prošla za poslední čas obrovskou přeměnou, celou řadou partnerů, v první řadě společnosti Microsoft se je daří úspěšně řešit tak, že budeme-li již nyní provádět vyhodnocení této radikální změny, tak bude hodnocena pozitivně.

## Seznam použitých zdrojů


Riverbed:

Riverbed Steelhead SaaS [online]. [cit. 2018-03-01]. Dostupné z: http://www.riverbed.com/gb/products/steelhead/steelhead-saas.html

Microsoft Azure:

Microsoft Poweshell Installation [online]. [cit. 2018-03-01]. Dostupné z: https://azure.microsoft.com/cs-cz/documentation/articles/powershell-install-configure/

Microsoft VPN Gateway [online]. [cit. 2018-03-01]. Dostupné z: https://azure.microsoft.com/cs-cz/documentation/articles/vpn-gateway-about-vpngateways/#gateway-requirements

Microsoft MSDN [online]. [cit. 2018-03-01]. Dostupné z: https:// msdn.microsoft.com

# 8 Seznam příloh

# 9 Přílohy

## 9.1 Mapa systémů

### 9.1.1 Legenda

#### 9.1.1.1 Základní rozdělení systémů:

1) Interní systém
   - bílý čtverec s modrým orámováním
     např.

2) Interní systém bez GUI
   - šedý čtverec s modrým orámováním
     např.

3) Externí systém/prostředí/partner
   - modrý obdélník
     např.

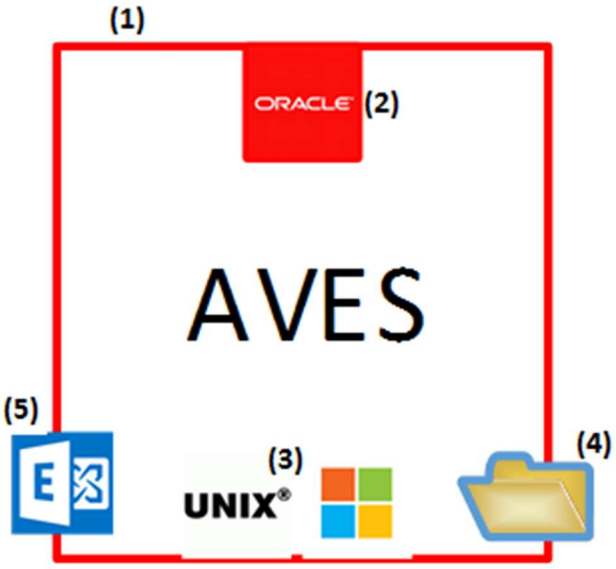
4) Interní prostředí čí používaná technologie
   - zelený čtverec s modrým orámováním
     např.

#### 9.1.1.2 Popis prostředí interní aplikace/systému:

1) červené orámování indikuje autentifikaci uživatelů dané aplikace vůči AD
2) ikony v horní části čtverce indikují užité databázové technologie
3) ikony v dolní části čtverce indikují užité technologie aplikačních serverů či operačních systémů
4) ikona v pravém spodním rohu indikuje využití DFS
5) ikona v levém spodním rohu indikuje komunikaci systému s MS Exchange
6) Přehled dalších a užitých ikon:

   - Microsoft

- IBM Websphere MQ
- Apache
- Tomcat
- FoxPro
- Oracle
- GlassFish
- Linux
- Unix

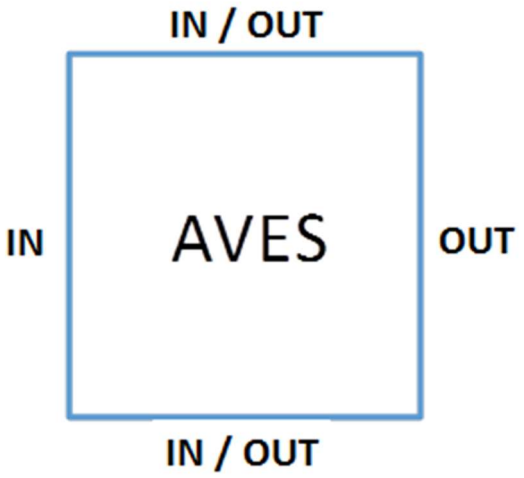



9.1.1.3 Popis prostředí externího systému/partnera:

1) Ikona v levém horním rohu indikuje, že systém/partner komunikuje prostřednictvím IBM Websphere MQ

#### 9.1.1.4 Napojení interface

1) Vstup do systému je značen připojením k levé hraně systému
2) Výstup ze systému je značen připojením k pravé hraně systému
3) Oboustranná komunikace je značena připojením k dolní či horní hraně systému



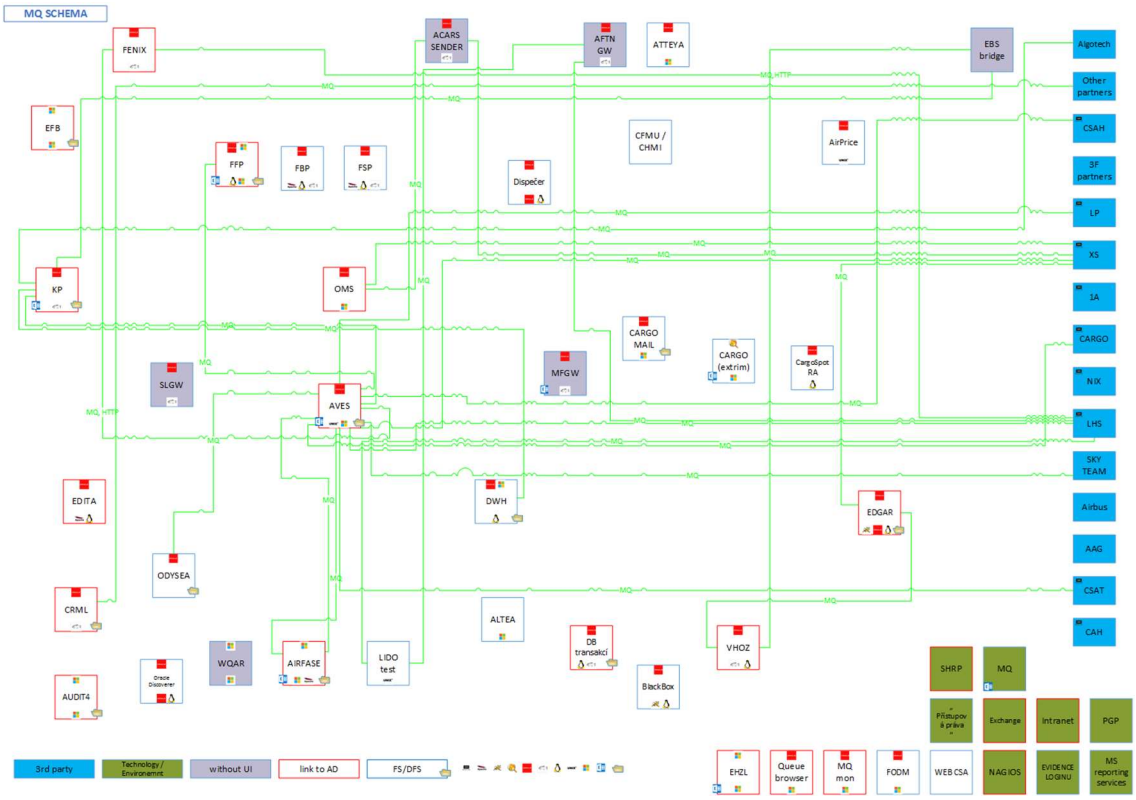

#### 9.1.1.5 Typy interface

Každý typ interface je přiřazen minimálně jedné vrstvě, které jsou barevně odlišeny.

1) - komunikace prostřednictvím MQ. Za MQ může být další typ komunikace.
2) - komunikace prostřednictvím protokolů http, https, SOAP (webové služby)
3) - obecná TCP/IP komunikace
4) - komunikace prostřednictvím FTP, FTPS, SFTP, SCP
5) - komunikace prostřednictvím přímého databázového linku

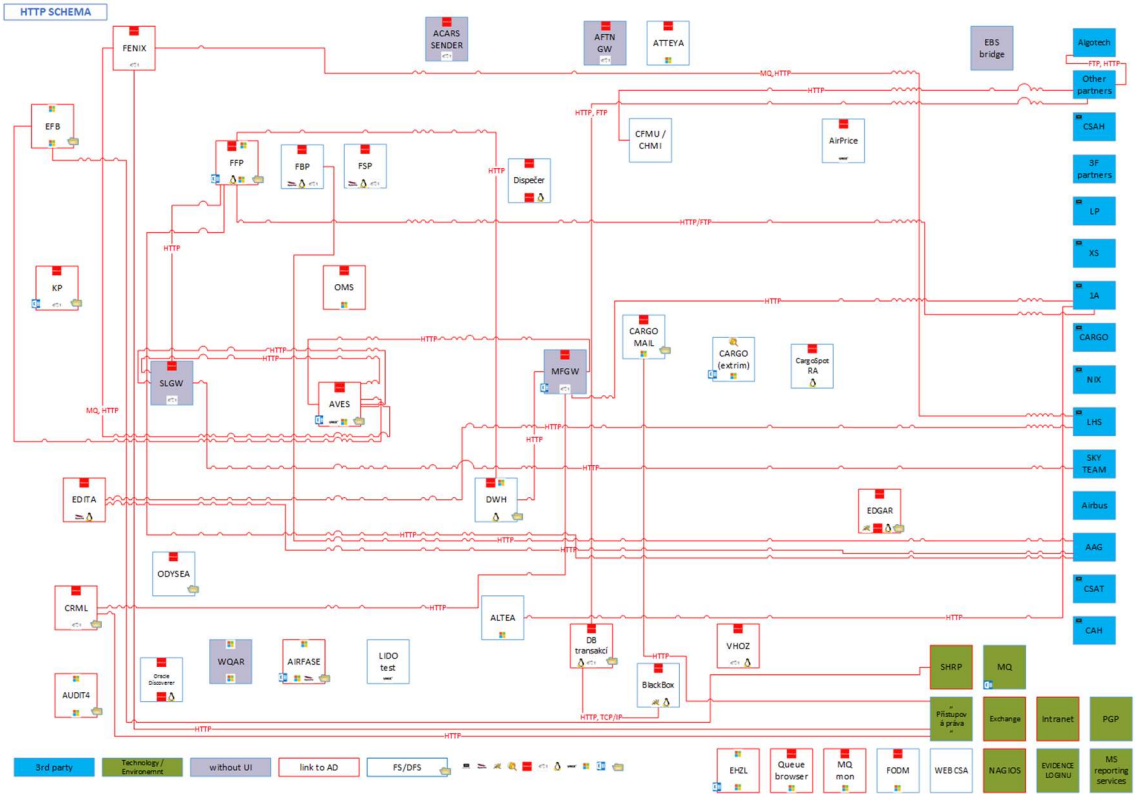
### 9.1.2 Hlavní komunikační schéma

## 9.1.3 MQ komunikační schéma



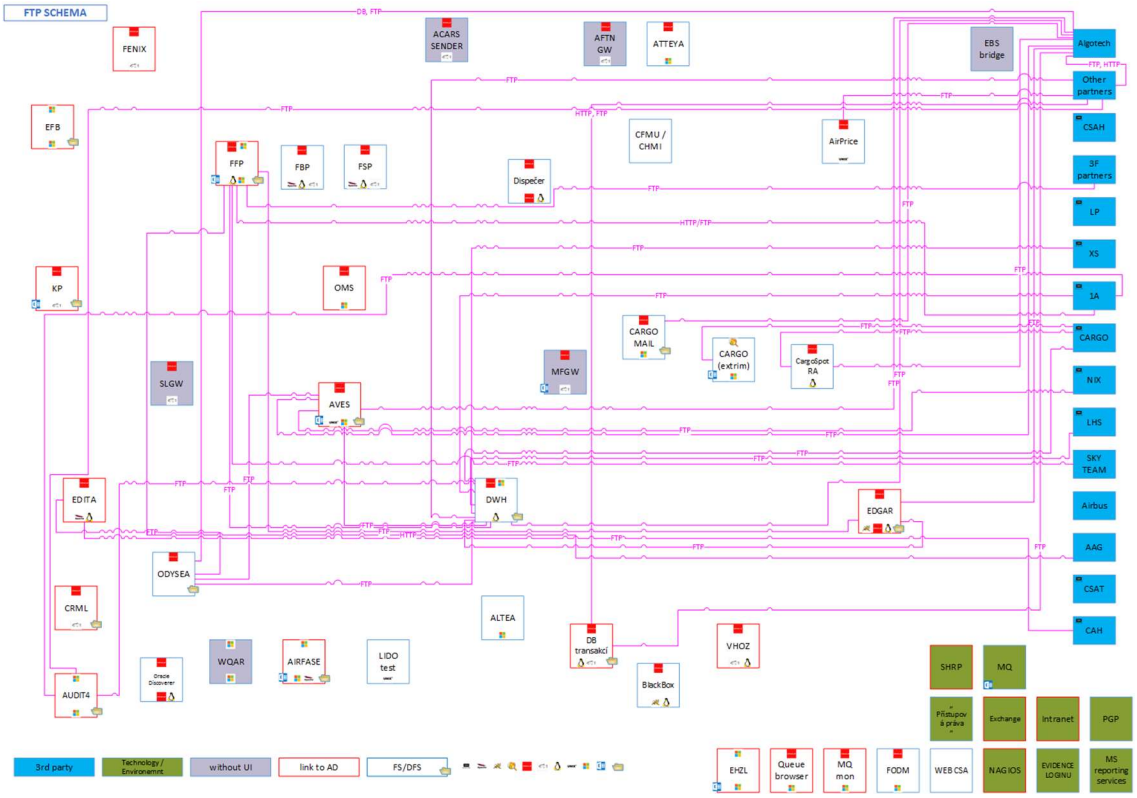

*Obrázek 9-1 MQ komunikační schéma*

## 9.1.4 Webové služby komunikační schéma



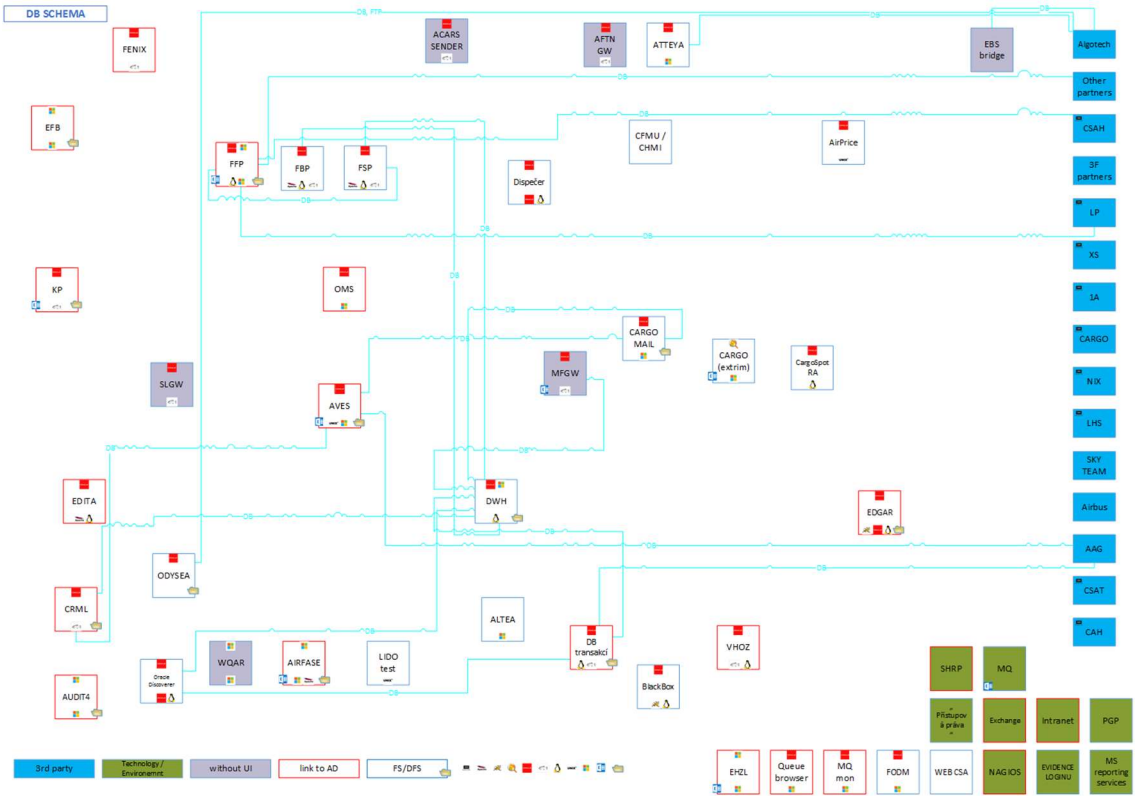

*Obrázek 9-2 Komunikační schéma webových služeb*

## 9.1.5 FTP komunikační schéma



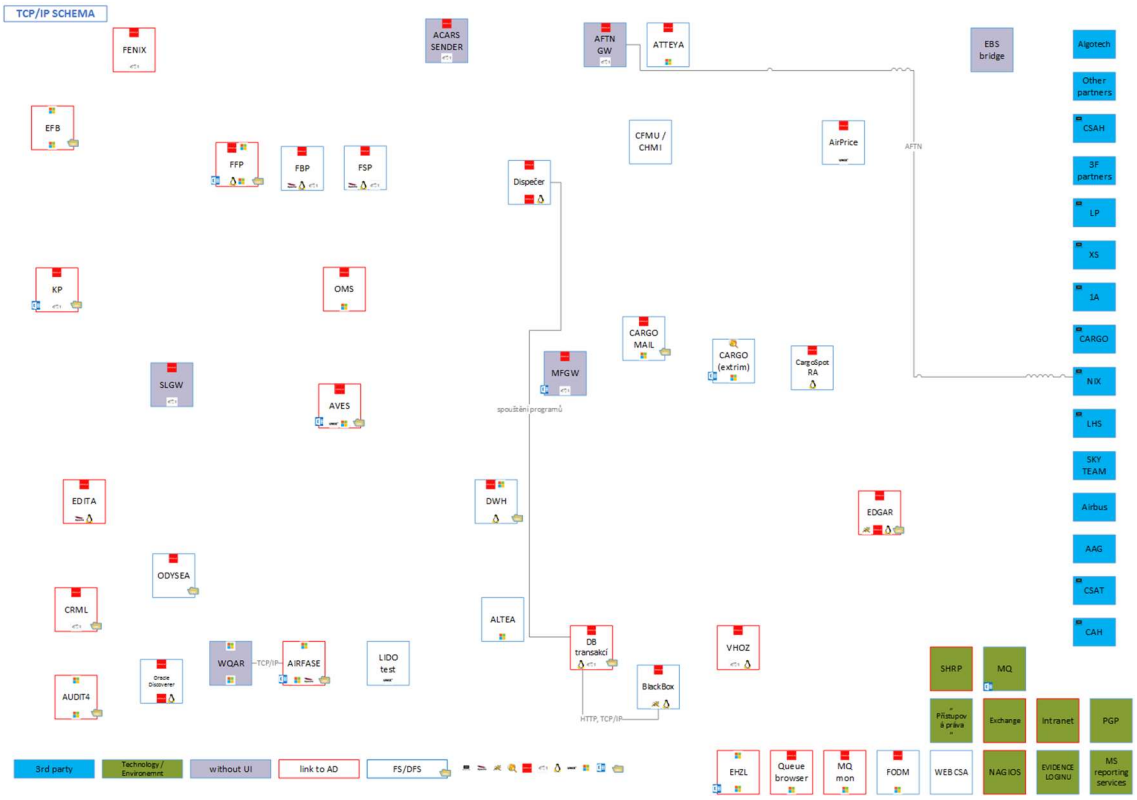

*Obrázek 9-3 Komunikační schéma FTP/SFTP protokolů*

## 9.1.6 Databázové komunikační schéma



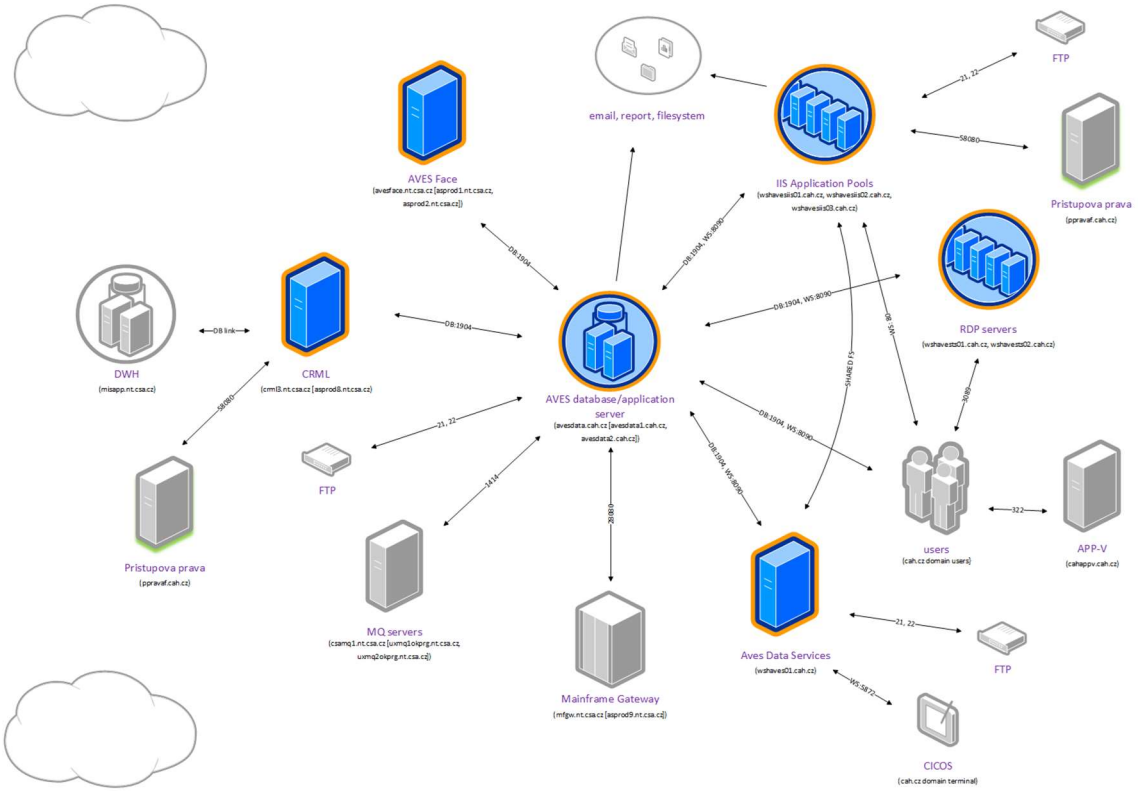

*Obrázek 9-4 Komunikační schéma databázových linků*

## 9.1.7 Ostatní typ komunikace



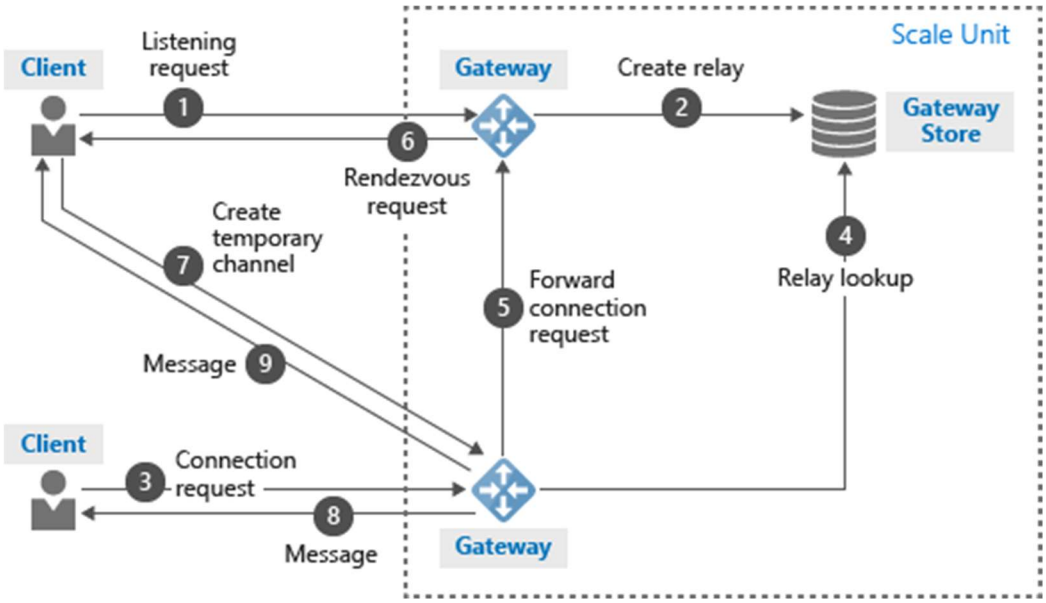

*Obrázek 9-5 Jiné typu komunikace*

## 9.2 AVES schéma




*Obrázek 9-6 Schéma AVES prostředí*

## 9.3 Komunikační schéma Azure Service Bus Relay




*Obrázek 9-7 Komunilační schéma Azure Service Bus Relay*

## 9.4 Powershell skripty pro vytvaření prostředí

Po definici prostředí a analýze potřebných výkonových kapacit připravuji skripty pro vytvoření prostředí.

Nainstaluji powershell moduly pro MS Azure z PowerShell galerie a připojím se k připravené subskripci

```
# Instalace Azure Resource Manager
Install-Module AzureRM

# Instalace Azure Service Management modulu
Install-Module Azure

#Připojení
$cred = Get-Credential
Login-AzureRmAccount -Credential $cred
```

Začnu připravovat prostředí inicializací proměnných.

```
$subscription = "PP Subscription"
$location="West Europe"
$label="PP - Lokalita 01"
$vmTimeZone="Central Europe Standard Time"

$OSNM = "saos01"

$DATANM = "sadata01"

$BACKUPNM = "sabackup01"

$vNetName = "vnet"

$credentials = Get-Credential -Message "Jméno a heslo účet lokálního
administrátora."
$vmLogon = $credentials.UserName
$vmPassword = $credentials.GetNetworkCredential().Password
```

Vytvoří storage účty jímž budou přiřazovaný nově vytvořená storage virtuálních strojů.

```
New-AzureStorageAccount -StorageAccountName $OSNM -Location $location -type
"Standard_LRS" -Description $label -Label $label

New-AzureStorageAccount -StorageAccountName $DATANM -Location $location -type
"Standard_LRS" -Description $label -Label $label
```

```
New-AzureStorageAccount -StorageAccountName $BACKUPNM -Location $location -type
"Standard_GRS" -Description $label -Label $label
```

Vytvoří VNET

```
$scriptDirectory = _GetScriptDirectory
$networkConfig = $scriptDirectory + "\VNet\NetworkConfig.xml"
```

kde NetworkConfig.xml obsahuje:

```
<NetworkConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://schemas.microsoft.com/ServiceHosting/2011/07/NetworkConfiguration">
  <VirtualNetworkConfiguration>
    <Dns>
      <DnsServers>
        <DnsServer name="DNS2" IPAddress="1.1.1.2"/>-->
      </DnsServers>
    </Dns>
    <VirtualNetworkSites>
      <VirtualNetworkSite name="ppbak1vnet" Location="West Europe">
        <DnsServersRef>
          <DnsServerRef name="DNS2"/>-->
        </DnsServersRef>
        <AddressSpace>
          <AddressPrefix>10.70.50.0/22</AddressPrefix>
        </AddressSpace>
        <Subnets>
          <Subnet name="GatewaySubnet">
            <AddressPrefix>10.70.50.0/29</AddressPrefix>
          </Subnet>
          <Subnet name="Subnet_DMZ">
            <AddressPrefix>10.70.51.64/25</AddressPrefix>
          </Subnet>
          <Subnet name="Subnet_Backend">
            <AddressPrefix>10.70.51.0/24</AddressPrefix>
          </Subnet>
        </Subnets>
      </VirtualNetworkSite>
    </VirtualNetworkSites>
  </VirtualNetworkConfiguration>
</NetworkConfiguration>

Set-AzureVNetConfig -ConfigurationPath $networkConfig
```

Inicializuje proměnné pro vytvoření virtuálních strojů

```
$WinIMG  = Get-AzureVMImage | where { $_.ImageFamily -eq "Windows Server 2012 R2
Datacenter" } | where { $_.Location.Split(";") -contains $location} | Sort-Object -
Descending -Property PublishedDate
       write-host $WinIMG .PublisherName
```

```
	write-host $WinIMG .Description
	W2K12R2DCImageName = $WinIMG [0].ImageName
```

Vytvoří si funkci na vytváření virtuálních strojů

```
function CrtVM(

	[string]$computerName,
	[string]$imageName,
	[bool]$LinuxFlag,
	[string]$vmSize,
	[string]$vmSubnetName,
	[string]$vmIPAddress,
	[string]$saOSName,
	[string[]]$saDataNameSizeInGB,
	[string[]]$endpointDefs,
	[string]$customScriptUri,
	[string]$customScriptName
	)
{
	$serviceName = "cs" + $computerName
	$service = _CreateService $serviceName
	_CrVM $computerName $serviceName $null $null $imageName $LinuxFlag $vmSize
$vmSubnetName $vmIPAddress $saOSName $saDataNameSizeInGB $customScriptUri
$customScriptName
	_AddEP $computerName $serviceName $null $endpointDefs
}
```

která volá další funkce _CrVM ,_AddEP

```
function _CrVM(

	[string]$computerName,
	[string]$serviceName,
	[string]$vmAsName,
	[string]$imageName,
	[bool]$LinuxFlag,
	[string]$vmSize,
	[string]$vmSubnetName,
	[string]$vmIPAddress,
	[string]$saOSName,
	[string[]]$saDataNameSizeInGB,
	[string]$customScriptUri,
	[string]$customScriptName
	)
{

	$vmName = "v" + $computerName
	$diskOSName = $computerName + "-osdisk"
	$mLoc = "https://" + $saOSName + ".blob.core.windows.net/vhds/" + $diskOSName
+ ".vhd"
```

```
	Set-AzureSubscription -SubscriptionName $subscription -CurrentStorageAccount
$saOSName

	If ($vmAsName) {

		$vm = New-AzureVMConfig -Name $vmName -InstanceSize $vmSize -
HostCaching "ReadWrite" -AvailabilitySetName $vmAsName -Label $label -ImageName
$imageName -DiskLabel $diskOSName -MLoc $mLoc
	}
	else {

		$vm = New-AzureVMConfig -Name $vmName -InstanceSize $vmSize -
HostCaching "ReadWrite" -Label $label -ImageName $imageName -DiskLabel $diskOSName
-MLoc $mLoc
	}

	$vm = $vm |
			Set-AzureSubnet $vmSubnetName |
			Set-AzureStaticVNetIP -IPAddress $vmIPAddress

	switch ($LinuxFlag)
	{
		$true { $vm = Add-AzureProvisioningConfig -Linux -VM $vm -LinuxUser
$vmLogon -Password $vmPassword -NoSSHEndpoint; break }
		default { $vm = Add-AzureProvisioningConfig -Windows -VM $vm -
AdminUsername $vmLogon -Password $vmPassword -DisableWinRMHttps -NoRDPEndpoint -
NoWinRMEndpoint -TimeZone $vmTimeZone; break }
	}

	If ($customScriptUri) {
		$vm = Set-AzureVMCustomScriptExtension -VM $vm -FileUri
$customScriptUri -Run $customScriptName
	}

	$lun=0
	for ($i=0; $i -lt $saDataNameSizeInGB.Length; $i=$i+2) {

		$saDataName = $saDataNameSizeInGB[$i + 0]
		$dataDiskSizeInGB = $saDataNameSizeInGB[$i + 1]

		$diskDataName = $computerName + "-data-disk" + ($lun + 1)
		$mLoc = "https://" + $saDataName + ".blob.core.windows.net/vhds/" +
$diskDataName + ".vhd"

		$vm = $vm |
			Add-AzureDataDisk -CreateNew -DiskSizeInGB $dataDiskSizeInGB -
DiskLabel $diskDataName -LUN $lun -HostCaching "None" -MLoc $mLoc

		$lun++
	}

		$vm = New-AzureVM -ServiceName $serviceName -VM $vm -VNetName
$vNetName


function _AddEP(
```

```
	[string]$computerName,
	[string]$serviceName,
	[string]$ILBName,
	[string[]]$endpointDefs
	)
{
	$vmName = "v" + $computerName
	$vm = Get-AzureVM -ServiceName $serviceName -Name $vmName

	for ($i=0; $i -lt $endpointDefs.Length; $i=$i+5) {

		$EPointName = $endpointDefs[$i + 0]
		$EPointLocPort = $endpointDefs[$i + 1]
		$EPointPubPort = $endpointDefs[$i + 2]
		$EPointProbeProtocol = $endpointDefs[$i + 3]
		$EPointProbePath = $endpointDefs[$i + 4]

		if (!$EPointProbeProtocol) {

			Add-AzureEndpoint -VM $vm -Name $EPointName -Protocol tcp -
LocalPort $EPointLocPort -PublicPort $EPointPubPort
		}
		else {

		If ($ILBName) {

				$ILBSETName = "ILBSet" + $EPointName
				Add-AzureEndpoint -VM $vm -Name $EPointName -LBSetName
$ILBSETName -InternalLoadBalancerName $ILBName -Protocol tcp -LocalPort
$EPointLocPort -PublicPort $EPointPubPort -ProbePort $EPointLocPort -ProbeProtocol
$EPointProbeProtocol -ProbeIntervalInSeconds 5 -ProbeTimeoutInSeconds 11 -ProbePath
$EPointProbePath
			}
			else {

				$ELBSETName = "ELBSet" + $EPointName
				Add-AzureEndpoint -VM $vm -Name $EPointName -LBSetName
$ELBSETName -Protocol tcp -LocalPort $EPointLocPort -PublicPort $EPointPubPort -
ProbePort $EPointLocPort -ProbeProtocol $EPointProbeProtocol -
ProbeIntervalInSeconds 5 -ProbeTimeoutInSeconds 11 -ProbePath $EPointProbePath
			}
		}
	}

	$vm = $vm | Update-AzureVM
}
```

### Vytváří funkci pro virtuální stroje ve farmě

```
function AddVM2Farm(

	[System.Object]$farm,
	[string]$computerName,
	[string]$vmSize,
```

```
        [string]$vmSubnetName,
        [string]$vmIPAddress,
        [string]$saOSName,
        [string[]]$saDataNameSizeInGB,
        [string[]]$endpointDefs,
        ) {

        _CrVM $computerName $farm.ServiceName $farm.ASName $WinIMG  $false $vmSize
$vmSubnetName $vmIPAddress $saOSName $saDataNameSizeInGB
        _AddEP $computerName $farm.ServiceName $farm.ILBName $endpointDefs
}
```

A nyní již přímým voláním vytváří potřebné virtuální stroje pro testovací prostředí modulárního systému AVES v MS Azure prostředí včetně služeb ADFS a ADFS WAP proxy.

```
CrtVM "petr_bakwp-vmdcrep" $WinIMG "Large" "Subnet_Backend" "10.70.51.30" `
             $OSNM @($DATANM,200 ) `
             @("RDP","3389","3300",$null,$null) $null $null



$farmCloudService = CreateHAFarm "avdtsvc"

AddVM2Farm $farmCloudService "avdtsvc01" "Medium" "Subnet_Backend" "10.70.51.45" `
             $OSNM @($DATANM,200 ) `
             @("RDP","3389","3300",$null,$null)

$farmCloudService = CreateHAFarm "avdtsvc"

AddVM2Farm $farmCloudService "avdtsvc01" "Medium" "Subnet_Backend" "10.70.51.46" `
             $OSNM @($DATANM,200 ) `
             @("RDP","3389","3300",$null,$null)



$farmCloudService = CreateExternalLBFarm "adfswap"

AddVM2Farm $farmCloudService "myadfswp01" "Large" "Subnet_DMZ" "10.70.51.105" `
             $OSNM @($DATANM,200 ) `
             @("HTTP","443","443","http","/", "RDP","3389","3300",$null,$null)


$farmCloudService = CreateInternalLBFarm "adfs" "Subnet_Backend" "10.70.51.50"

AddVM2Farm $farmCloudService "myadfs01" "Large" "Subnet_Backend" "10.70.51.47" `
             $OSNM @($DATANM,200 ) `
             @("HTTP","443","443","http","/", "RDP","3389","3300",$null,$null)
```