

Základní informace

Jméno projektu	<i>Anzu - Molecular visualization framework</i>
Zkratka	<i>Anzu</i>
Vedoucí	<i>RNDr. Václav Krajíček, Ph.D. <vaclav.krajicek@eyen.se></i>
Konzultanti	
Anotace	<i>Framework pro zobrazování molekulárních struktur pracující s vektorovými a rastrovými 3D daty, který je založen na moderních technologiích (WebAssembly, cloudové služby) a architektuře (microservices). Systém bude využívat standardních formátů a k získávání dat bude přistupovat do veřejných databází (např. PDB), případně k uživatelem specifikovaným zdrojům. Hlavním cílem je dosáhnout vysoké variability a utilizace na straně koncových zařízení.</i>

Motivace

Se vznikem formátu WebAssembly se otevřela možnost pro software k zobrazování molekulárních struktur, který by kombinoval rychlost nativních aplikací s jednoduchou použitelností webových řešení. Existující webová řešení mají ale velmi omezenou funkcionalitu a zároveň nejsou snadno rozšiřitelná, což výrazně limituje jejich užitečnost.

Popis projektu

Cílem projektu je vytvořit software umožňující pohodlné zobrazení molekulárních struktur jak z veřejných databází, tak ze soukromých a lokálních zdrojů/cloudu. Bude kladen důraz na jasnou strukturu frameworku a snadnou rozšiřitelnost o novou funkcionalitu.

Zároveň s tvorbou projektu bude probíhat i komunikace s potenciálními uživateli a odborníky v oblasti molekulární biologie ohledně rozhraní a dalších případů použití, na které bude potřeba při návrhu myslet.

Příklady použití:

- 1. Prezentování výsledků rekonstrukční metody porovnáním s již existujícím jiným výsledkem*
- 2. Vizualizace molekuly nebo části různými způsoby pro lepší porozumění její struktury*
- 3. Rozšíření softwaru o nový typ zobrazení dat v rámci molekulárních struktur, např.*

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0203085>

***vizualizace** - Software by měl umožňovat zobrazení více struktur najednou, otáčení, přiblížení a změnu typu zobrazení nezávisle pro každý vzorek, nebo i pro více najednou. Podporovány by měly být všechny obvyklé typy zobrazení: surface, ribbons a atoms. Dále by mělo být možné interaktivně zobrazovat prostorové popisky daných struktur, např. značky atomů.*

***surface zobrazení** - Některé datové formáty používané pro uložení molekulárních dat mají volumetrickou strukturu. Vzhledem k výpočetní náročnosti zobrazení takových dat je bude potřeba*

předzpracovat do geometrické sítě pomocí známých algoritmů, např. marching cubes.

fitting struktur - Jednou z užitečných vlastností programů ke zobrazení molekulárních struktur je zobrazení více vzorků přes sebe s nastavitelnou průhledností. Pro tento postup je ale třeba položit více či méně odpovídající si struktury “co nejlépe” přes sebe. K tomu bude potřeba implementovat funkci, která takový fitting provede, a to i na různých formátech dat.

cloudový backend - Pro potřeby rozšiřitelnosti o novou funkcionalitu by měl výsledný software být schopen využít microservices, jak k získávání, tak ke zpracování dat. Přístup bude demonstrován na preprocessingu dat pomocí marching cubes.

GUI - Součástí výsledného software by mělo být i funkční grafické rozhraní, které dovolí i uživateli bez znalosti programování pohodlně ovládat a používat zmíněné funkce programu, hlavně v oblasti nahrávání a zobrazování struktur.

Modularita - GUI a jádro aplikace, stejně jako cloudové rozhraní, by měly být rozděleny čistým API. Výměna uživatelského rozhraní za jiné, např. jednodušší prezentační by měla být rozumně jednoduchá. Podobně jednoduché by mělo být i přidání nové cloudové funkce/zdroje dat.

Platforma, technologie

Výsledný software se bude skládat z několika částí. První část bude běžet na zařízení uživatele a podle výkonu aktuálního uživatelského zařízení bude poskytovat podmnožinu funkcionality celé aplikace. Druhá část, běžící v cloudu, bude poskytovat funkcionalitu, kterou nelze implementovat v uživatelské části.

Uživatelská část by měla být spustitelná přímo v prohlížeči, s rozumnou rychlostí a omezeními i na mobilních zařízeních. Zároveň bude existovat i nativní aplikace pro desktop. Většina práce bude napsána v jazyce C++.

Jádro uživatelské části bude tvořit herní engine s podporou kompilace do WebAssembly, který poskytne i základ renderovací pipeline a GUI prvky. V případě nedostačujících GUI elementů herního engine bude do projektu navíc integrována GUI knihovna, např. Qt.

Ke komunikaci s databází a cloudem bude využita interní síťová knihovna použitého herního enginu, případně bude využita jiná, WebAssembly kompatibilní, knihovna. Výsledný kód bude přeložen pomocí Emscripten do WebAssembly formátu nebo jako nativní aplikace.

Návrh cloudové části projektu bude realizován podle architektury Microservices, s možností využití technologií jako Serverless computing, Docker, Docker swarm, Kubernetes a podobných pro implementaci a orchestraci microservice. Cílem návrhu bude umožnit co nejjednodušší rozšiřování funkcionality v podobě přidávání dalších microservice. Implementace cloudového backendu bude v rámci projektu cílená na AWS cloud, s myšlenkou budoucího rozšíření na cloud jiných poskytovatelů. Ukázkové microservices budou implementovány za použití jazyka a technologií

vhodných pro daný úkol.

Odhad náročnosti

Projekt by mělo být možné realizovat v počtu 4 řešitelů. Z důvodu zaměření projektu ho lze na nejvyšší úrovni rozdělit na dvě oblasti: počítačová grafika a zpracování/distribuce dat. Tým by se tedy měl skládat zhruba z dvou vývojářů s kompetencemi v oblasti počítačové grafiky a dvou vývojářů s kompetencemi v oblasti distribuovaných systémů.

Projekt by měl být dokončen v červenci 2020.

Vymezení projektu

Diskrétní modely a algoritmy	
	diskrétní matematika a algoritmy
	geometrie a matematické struktury v informatice
	optimalizace
Teoretická informatika	
	Teoretická informatika
Softwarové a datové inženýrství	
X	softwarové inženýrství
X	vývoj software
X	webové inženýrství
	databázové systémy
	analýza a zpracování rozsáhlých dat
Softwarové systémy	
	systémové programování
	spolehlivé systémy
	výkonné systémy
Matematická lingvistika	
	počítačová a formální lingvistika
	statistické metody a strojové učení v počítačové lingvistice
Umělá inteligence	
	inteligentní agenti
	strojové učení
	robotika

Počítačová grafika a vývoj počítačových her	
X	počítačová grafika
	vývoj počítačových her

Poznámky

Kandidát na herní engine s podporou WebAssembly: <https://urho3d.github.io/>

Pravděpodobně nejsilnější konkurence: <https://www.litemol.org/Viewer/>