# Стаття 2

# Ключові слова :

розробка онтологій, ontop, IDE, joint development , Protege, Protege Online

# Постановка проблеми

Онтології викооистовуються для оптимізації процесів, іще чогось і іще чогось. У інтернет магазинах можуть викорстовуватися для опису продуктів для зручного їх пошуку. У вікі онтології використовуються для зберігання інформації про явища та його звязку з іншими явищами.

Онтологія складається із мета моделі , в яку входять класи та проперті (дата проперті. що визначають значення та обєктні проперті, що визначають звязки між обєктами) та об’єкти, що є інстансами класів із уже заповненими пропертями.

У випадку, коли для роботи треба заповнити онтологію інформаціюєю, тобто створити обєкти уже попередньо створених класів, робота із онтологією за допомогою таких інструментів як Protege та Protege Online може не бути оптимізованою та забирати більше часу аніж треба.

Саме в цьому і є проблема, яка дослдіжується в даній статті. Проблема оптимізації внесення інформації в попередньо розроблену онтологію для подальшого аналізу та пошуку знань.

# Аналіз останніх досліджень

На сьогодні протеже дозволяє використовувати плагіни. Серед них і OnTop - плагін, що дозволяє звязувати реляційну базу даних із онтлогією.

У разі, ж коли треба не лише звязати дані із онтологією, а перенести їх, є такі рішення (чи немає).

таким чином не можна ніяк занести дані в онтологію якимось шляхом окрім як вручну.

# постановка завдання

Дослідити альтернативні методи внесення інформації в онтологію.

підрахувати кількість кліків, потрібних для створення обєкту в протеже і в моїй системі.

# Результати дослідження

Була зроблена спроба створити реляційне представлення онтології та створити систему, що дозволяє працювати із даними в онтології у вигляді таблиць.

Розроблену систему можна розвертати однією командою.

Керувати правами доступу доволі таки просто.

Представлені в системі функції - це ...

Таким чином людина. що не знайома із протеже та стандартом owl зможе без проблем працювати із даними в онтології так, наче це була б звичайна реляційна база даних.

Після наповнення онтології даними можна її викачати із системи і використовувати в інших системах.

Як працює система.

Справа на сторінці для роботи із даними знаходиться дерево класів. Це дерево класів онтології, вибраної користувачем.

При натиску по класу зявляється таблиця із обєктами даного класу. Обєктом (інстансом) даного класу вважається будь який інстанс у якого є .....

Для того, яби створити новий інстанс даного класу , треба натиснути кнопку New....

При натиску система читає мета дані онтології (дерево класів і проперті, що асоційовані із цими класами) і тоді на основі інформації про клас будує форму для стоврення обєкту даного класу. У формі також проводиться валідація (чи поле мандаторне чи ні) , для обджект пропертей створюється поле із випадаючим списком із усіма обєктами, що можуть бути обрані (список складається на основі інформації про пропертю і тип інстансів, на які вона може ссилатися).

Таким чином на одній формі можна заповнити усі поля обєкту. Це означає, що кількість мануальних дій. потрібних для створення і заповнення даних про один інстанс в онтології зменшується. І створення об’єктів в онтології стає більш прозорим.

вирізка із диплому

Також система дозволяє виконувати прості запити на мові запитів SPARQL. Підтримки різонерів іще немає.

Тому нажаль пошук знань у даній системі провести неможливо. Але якщо завдання передбачає створення великої кількості обєктів у онтології (наприклад занесення каталогу товарів електронного магазину) таку роботу можна простіше виконати паралельно в даній системі.

# Висновки

Використання розробленої системи значно зменшує час, потрібний для ознайомлення із онтологіями і степінь входження. Інтерфейс є простим. Система дозволяє працювати одночасно над однією онтологією в онлайн режимі із різних пристроїв. Також система дозволяє збільшити надійність даних, адже онтологія зберігається на сервері а не на персональному компютері.

Оскільки в системі передбачені різні ролі і степені доступу . система дозволяє підвищити степінь надійності та попередити помилки.

# Джерела інформації (30 штук)

https://github.com/ontop

https://www.w3.org/2001/sw/wiki/Ontop

https://link.springer.com/chapter/10.1007/978-3-642-41335-3\_35

https://link.springer.com/chapter/10.1007/978-3-662-45495-4\_6

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.7458&rep=rep1&type=pdf

https://vern.pw/1608881033.pdf

https://link.springer.com/chapter/10.1007/978-3-540-39964-3\_54

https://link.springer.com/chapter/10.1007/11595014\_22

https://corais.org/sites/default/files/ontology\_development\_101\_aguide\_to\_creating\_your\_first\_ontology.pdf

отут порівняння різних тулів для роботи із онтологіями

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.222.3953&rep=rep1&type=pdf

https://www.researchgate.net/profile/Thabet\_Slimani1/publication/288258366\_Ontology\_Development\_A\_Comparing\_Study\_on\_Tools\_Languages\_and\_Formalisms/links/56d1369608ae85c823487dc6.pdf

https://watermark.silverchair.com/btg194.pdf?token=AQECAHi208BE49Ooan9kkhW\_Ercy7Dm3ZL\_9Cf3qfKAc485ysgAAAr4wggK6BgkqhkiG9w0BBwagggKrMIICpwIBADCCAqAGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMMDX9t7F8ZendkVjIAgEQgIICcfrNN05TQWILVLLeW2dr\_cS07TfSN0qn7iUSFn3MGPtnkGrY-kzCvevVULRAQXOEBlW8fGMC7dIoyh6Z5Rgo0h3VDLBbZGBLlX3a93UJyZZn3N3XmAK-vIafqtrAlDYuuTtey53LlSvZ7T0Hf0HLWSJ3SsWsimBiL9UkHduzvFZMGQneh2NpKLp6Rd1vQ6djTicAYEwoV6HuOHDbyGpMT1\_UyNsn1Dd1fYlRGt3Fdzp-fVH\_FHb-2z7rteBvPapKCh30Zu8PuTM-Nl2SfkAEqksQ3hq4C3U5-CHJQ2SyGQyRLvrDt3hBDtgv5LRe-VL1nh4AuAWuaNbkkvT24lBljgrylh1FqW3Kdl-mgzkVNSmBTtkAY513rcR7rHbWwOfjDNLk-qR8cYG8Zof9E78\_Rrw1kWmsEueZV9itqMrA5K-A1B47EJFzfY1j4ByPvDo1VAy6Tp8uC6rGCJLsFYuFX2uC2SCg9XXUALr7xMf6f4nhinSVTUCv3fENa-G\_ga1nT9exGcmFwY84LsQtubuwogYgRKO9fOqcRgqOmpXM6VkrVkLRFrh483O2hXw-AL92V4835OYuuSEbVKiIjmgwjZZuaVDvumkfHr8RenjxuOQ8O6CcVZWT2FwNOm2H6PHFSxd6auNpIvSXwaw-ipxKm0aIyPC34gcXxTOZXz21jyW-0CHRXrtoqM4fiBREyHnYo51BePdNP-\_dSwZ\_JeDB4JCXVZFex7gurBnTHZUeMj6koYzbHzQ3p\_KNtRVZ\_SspFpqlK\_LqsgbAsc03VnykDDPVpdtQ6s3L7905aOIuavaminejkWkGQ5XC7TKWIMdtiPw