



PROGRAMŲ SISTEMŲ DINAMINĖS INTEGRACIJOS TAIKANT AUTONOMINIO SKAIČIAVIMO TECHNOLOGIJAS TYRIMAS

Software systems dynamic integration based on
autonomic computing

Doktorantas: Andrius Valatavičius
Vadovas: Prof. Dr. Saulius Gudas

Turinys

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

Doktorantūros informacija

- **Doktorantūros informacija**
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

- Preliminari disertacijos tema: Programų sistemų dinaminės integracijos taikant autonominio skaičiavimo technologijas tyrimas
- Darbo Vadovas: Prof. Dr. Saulius Gudas
- Pradžia 2014 Spalio 1d. – pabaiga 2018 Rugsėjo 30 d.

- Doktorantūros informacija
- **Tyrimo objektas ir tikslai**
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

Tyrimo objektas

- Programų sistemų integravimo ir jų sąveikavimo automatizavimas dinaminėje verslo aplinkoje

Tyrimo tikslas

- Ištirti programų sistemų integravimo bei jų sąveikų plėtojimo galimybes, taikant autonominio skaičiavimo metodus bei veiklos domeno žinių modelius

Planuojami gauti rezultatai

- Išanalizuoti eksperimento metu gautus rezultatus
- Įvertinti bei palyginti skirtingų programų sistemų tarpusavio sąveikumą (angl. interoperability), su kitais egzistuojančiais sąveikumo lyginimo metodais
- Pateikti torinius sprendimus kaip būtų galima pagerinti programų sistemų sąveikavimą automatizuotais metodais

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- **Planuojami gauti rezultatai**
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

Darbo planas 2016/2017

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- **Ataskaitinių metų darbo planas**
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

- Dalyvavimas tarptautinėje arba mokslinėje konferencijoje 2016 balandžio mėn.
- Dalyvavimas tarptautiniame seminare 2016 gruodžio mėn.
- Dalyvavimas tarptautinėje arba mokslinėje konferencijoje 2017 lapkričio mėn.
- Dalyvavimas tarptautiniame seminare 2017 gruodžio mėn.
- Straipsnis žurnale 2016 m. gruodžio mėn.
- Straipsnis žurnale 2017 m. gruodžio mėn.
- Paruošti eksperimentinio tyrimo aplinką integracijos savybių ir apribojimų tyrimui pagal siūlomą ir egzistuojančius metodus iki 2017 m. gegužės mėn.;
- Programinio prototipo sukūrimas demonstruojantis galimą metodo taikymą praktikoje iki 2017 gegužės mėn.;
- Gautų duomenų analizė ir apibendrinimas, išvadų parengimas: sukurto modelio įvertinimas; atlikto eksperimentinio tyrimo pagrindimas išvadų parengimas iki 2017 rugsėjo mėn.

Ataskaita už 2016/2017 metus

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- **Ataskaita**
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- Rezultatų pristatymas

- Išlaikyti visi egzaminai.
- Dalyvauta Doktorantų konsorciume BIR 2015 Estijoje tema: “Enterprise Software System Integration using Autonomic Computing”;
- Dalyvauta Doktorantų konsorciume DB&IS 2016 Latvijoje tema: “Modelling Dynamic Enterprise Environment to Maintain Interoperability of Applications”;
- Dalyvauta tarptautiniame seminare DAMSS „Data Analysis Methods for Software Systems“ 2016;
- Dalyvauta XVIII tarptautinėje kompiuterininkų konferencijoje LIKS 2017, tema: Towards deep knowledge based interoperability of applications. Straipsnis priimtas publikacijai žurnale „Informacijos Mokslai“;

Gauti moksliniai rezultatai

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- **Gauti moksliniai rezultatai**
- Kitų metų darbo planas
- Rezultatų pristatymas

- Dalinai sukurtas programinis prototipas demonstruojantis galimą metodo taikymą praktikoje, dėl techninių kliūčių nepabaigtas;
- Gautų duomenų analizė ir apibendrinimas, išvadų parengimas: sukurto modelio įvertinimas; atlikto eksperimentinio tyrimo pagrindimas išvadų parengimas;

2017/2018 m. darbo planas:

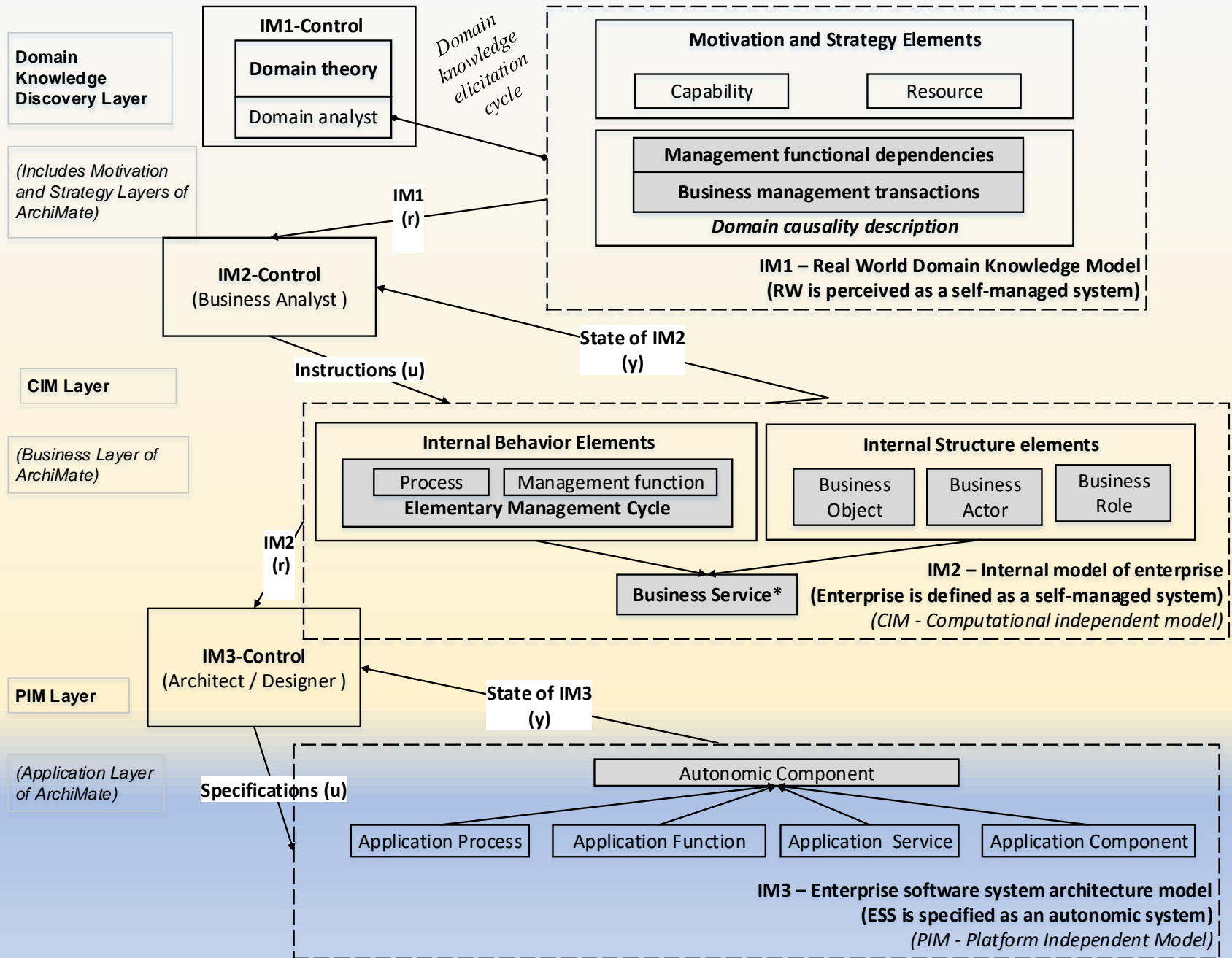
- Publikuoti straipsnį tarptautiniame moksliniame žurnale, tematika: „Software systems interoperability evaluation methods“.
- Publikuoti straipsnį tarptautiniame moksliniame žurnale, tematika: „Automation of Software systems interoperability“.
- Atskirų daktaro disertacijos dalių: tyrimo, metodikos, rezultatų, ginamų teiginių ir išvadų parengimas.
- Analitinės dalies parengimas
- Teorinės dalies parengimas
- Eksperimentinės dalies parengimas
- Bendrųjų išvadų formulavimas

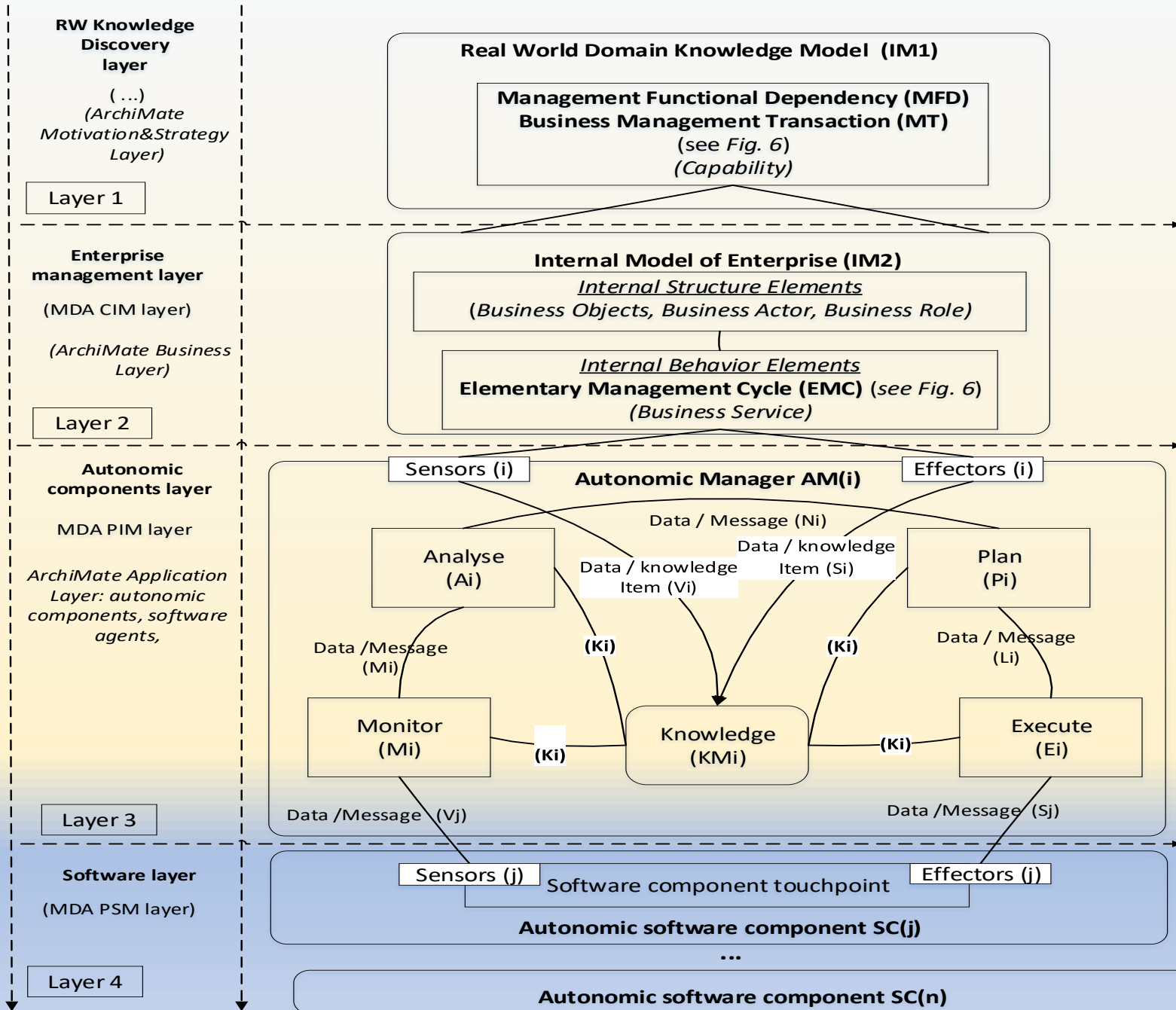
- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- **Kitų metų darbo planas**
- Rezultatų pristatymas

Rezultatų pristatymas

- Doktorantūros informacija
- Tyrimo objektas ir tikslai
- Planuojami gauti rezultatai
- Ataskaitinių metų darbo planas
- Ataskaita
- Gauti moksliniai rezultatai
- Kitų metų darbo planas
- **Rezultatų pristatymas**

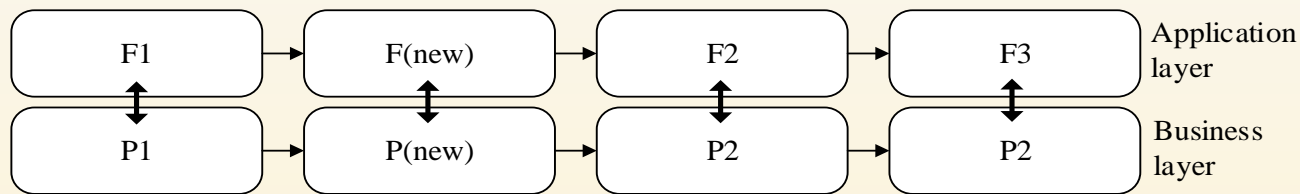
- Teorinis naujumas
 - Sąveikiuose (angl. interoperable) programų sistemose išnaudojami veiklos architektūros modeliai
- Inžinerinis naujumas
 - Tinklo servisų ir modelių analizavimas jų panaudojimas kaip žinių komponentes integruojamose sistemose
- Prototipo kūrimas
 - Sudėtinga sukurti automatiškai sąveikaujančių sistemų prototipą. Prototipo kūrimo metu iškilo techninių problemų neleidžiančių jo pabaigti nes reikia aprėpti struktūras skirtingų programų sistemų ir verslo veiklos modelių, kas reikalauja labai daug laiko
- Tyrimo duomenų analizės metodo sukūrimas
 - Kuriama vertinimo sistema kuri leistu palyginti kiek procentaliai programų sistemos yra interoperabilios savaime ir koks interoperabilumas būtų pasiektas naudojant verslo veiklos medelius.
 - Taip pat šį metodą galėsime palyginti su egzistuojančiais metodais kadangi pastebėjome jog egzistuojantys metodai nepakankamai objektyviai vertina sistemų interoperabilumą.
- Teorinis sprendimo vertinimas
 - Kol kas negalime pilnai įvertinti autonominės integracijos kūrimo efektyvumo nes neturime kaip vertinti sistemų tarpusavio sąveikumo. Tačiau mūsų iškeltos prielaidos gali padėti teorinius pagrindus išnaudoti autonomines technologijas programų sistemų integracijų ir interoperabilumo srityje. Kas galėtų būti geras indelis tolesniems darbams.



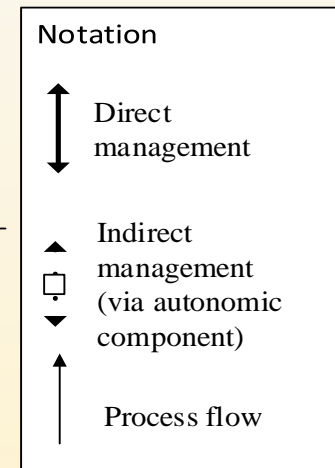
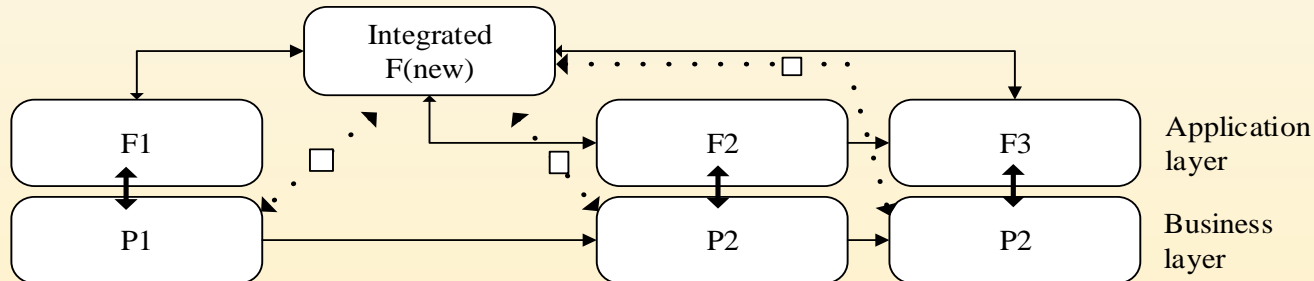


Rezultatų pristatymas

a) Legacy system (a lack of interoperability capability)



b) System with interoperability component



The new application *Integrated F(new)* is an autonomic component (application layer in fig. 11b), which is able for self-integration with existing (legacy) applications due to the internal knowledge (captured in KMi, see fig. 10). In this case, due to such self-integration functionality of the *Integrated F(new)*, there is no need for changes in the existing business processes (Business layer in fig. 11b).

Prototipo kūrīmo stadija

Application interoperability experiment

This is experimentation platform for application interoperability solutions.
Here you can:
*select all accessible applications for interoperability testing.
*view their statistics and data
*select method for interoperability

Start test Test time started: 0000-00-00 00:00:00
Test time running: 00:00:00

Integration method

C# integration
 Enterprise Application Integration (EAI) - Talend
 Integration Agents
 Autonomic
 Autonomic with internal model

Run Manually

Some simple interoperability methods needs to by run manually.
They commonly use sheduled jobs to run tasks.
We wouldnt like to wait long for
results so run manually button helps.

Select application 1 Select application 2

Read

Select Components Select Components

Records: 0 Records: 0 Difference: 0

If difference 0 - Application components are interoperable

Pagrindinės problemos

- Prototipo kūrimas sudėtingas reikalauja daugiau pastangų
- Reikia užbaigti kurti skirtingų programų sistemų vertinimo metodą

Děkoju už děmesj