Рассредоточение реализации приложений в распределенной среде предметных посредников

Вовченко Алексей
Институт Проблем Информатики РАН
Научный руководитель: Калиниченко Л.А.

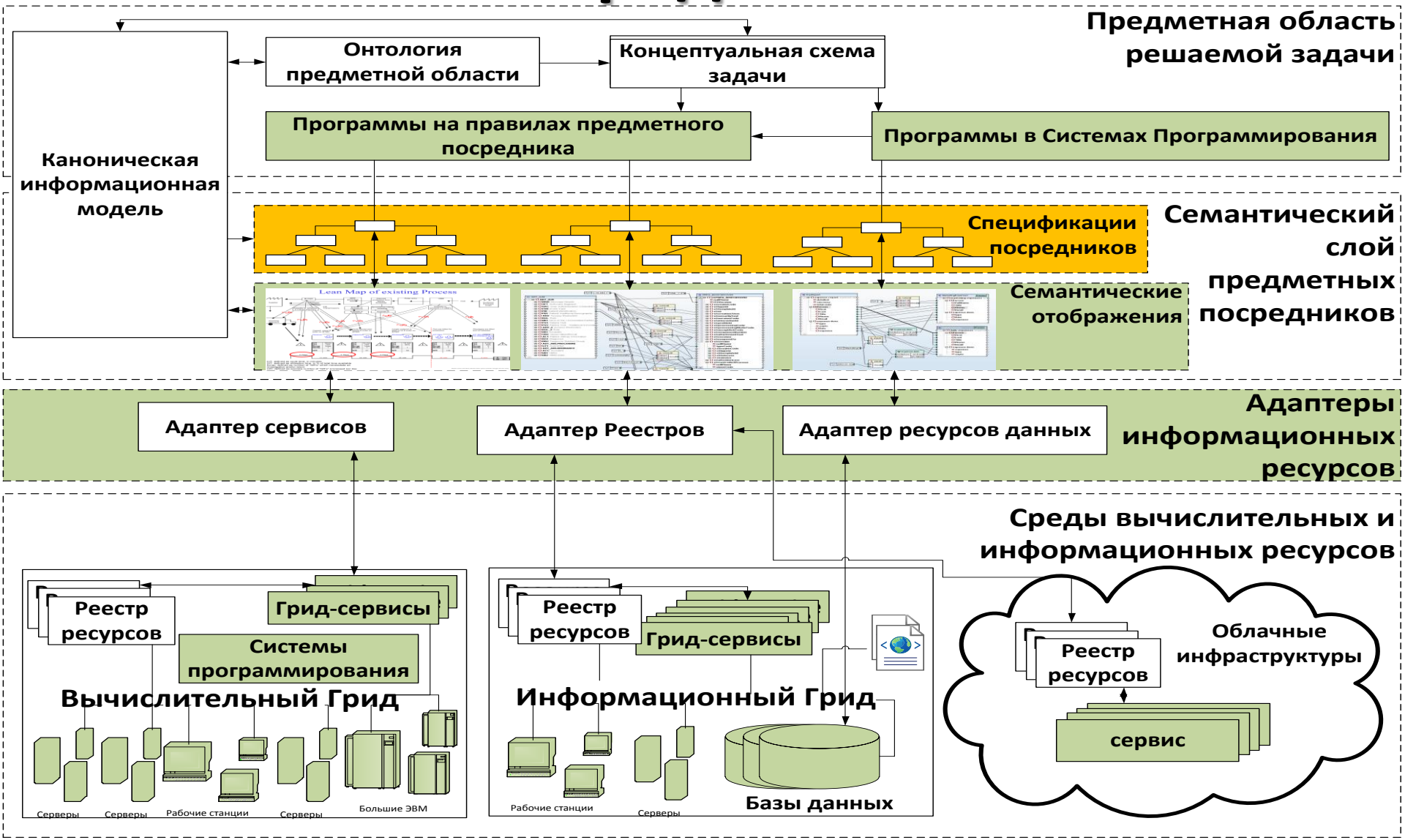
Диссертационный семинар
21.10.2011

RCDL’2011

План

- Среда предметных посредников
- Постановка задачи рассредоточения
- Беглая характеристика других задач работы
- Подход к построению эффективного рассредоточения
- Выводы

Инфраструктура предметных посредников



Подход движимый приложениями

- ***Подход движимый приложениями включает***: абстрактную **спецификацию предметной области** в виде онтологических концептов, структур данных, функций, процессов **независимо от существующих спецификаций ресурсов**
- Спецификация предметной области определяемая специалистами определяет **предметный посредник**
- **Регистрация релевантных ресурсов в посреднике основана на семантическом отображении схемы ресурса в схему посредника**
- Среда предметных посредников предоставляет:
  - 1) конструирование **канонической информационной модели** для унифицированного представления неоднородных онтологий, данных, сервисов и процессов;
  - 2) **построение схемы предметного посредника** как спецификации предметной области;
  - 3) **идентификацию и семантическую интеграцию** релевантных ресурсов в посреднике;
  - 4)спецификацию **алгоритма решения задачи** и ее **выполнение**.

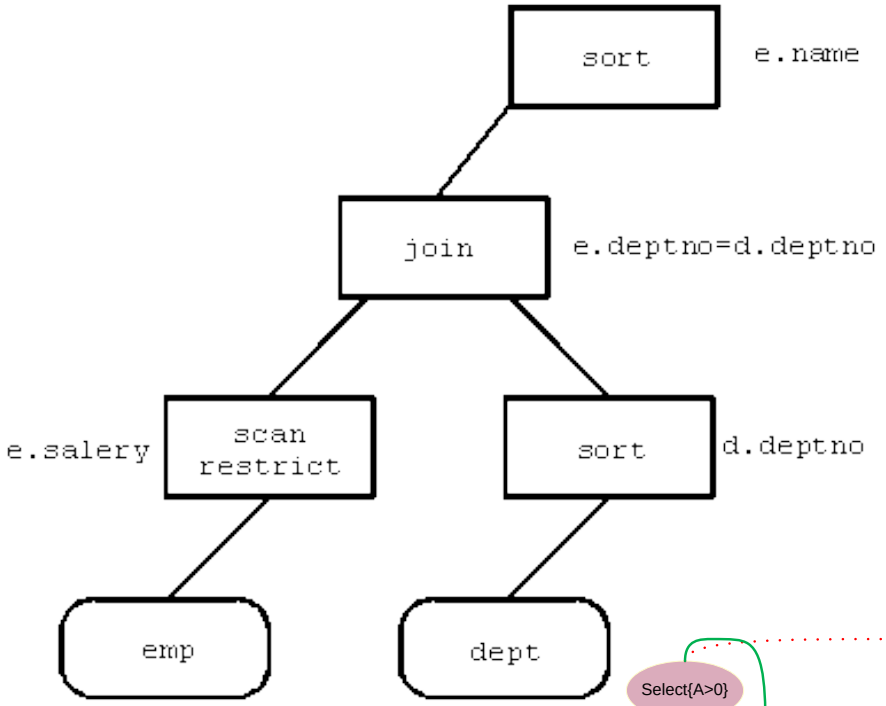
Исполнительный слой предметных посредников

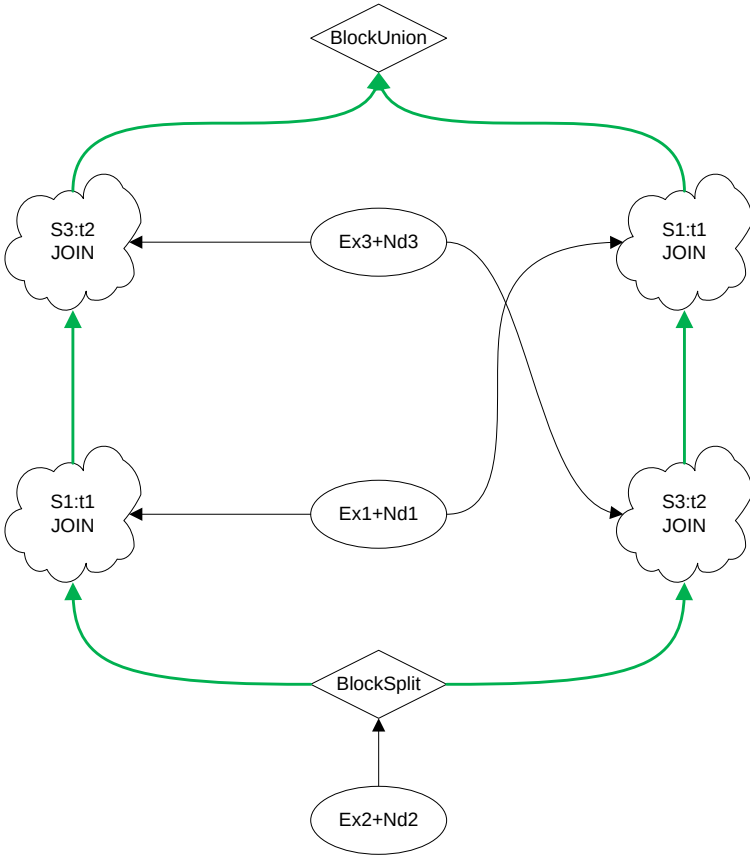
Постановка задачи

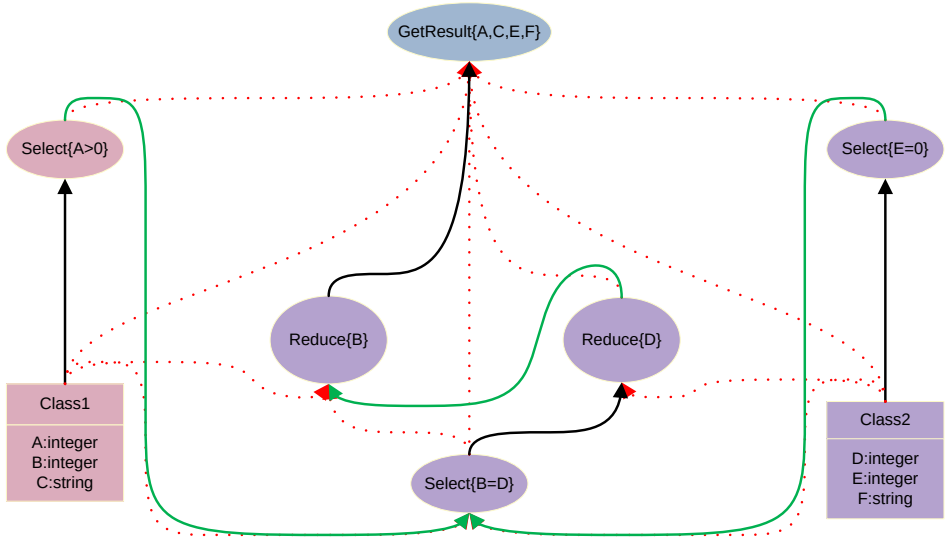
- Разработка подхода к построению эффективного рассредоточения для реализации алгоритма решения задач в среде предметных посредников
  - Много язычная среда программирования (Java, Datalog, PL/SQL, T-SQL, SYNTHESIS) используется для спецификации алгоритма решения задачи
  - Каждый язык программирования(ЯП) может использоваться для реализации различных частей спецификации алгоритма решения задачи
  - Множество возможных вариантов образует модель рассредоточения, представленную как граф зависимостей операций

Задача планирования

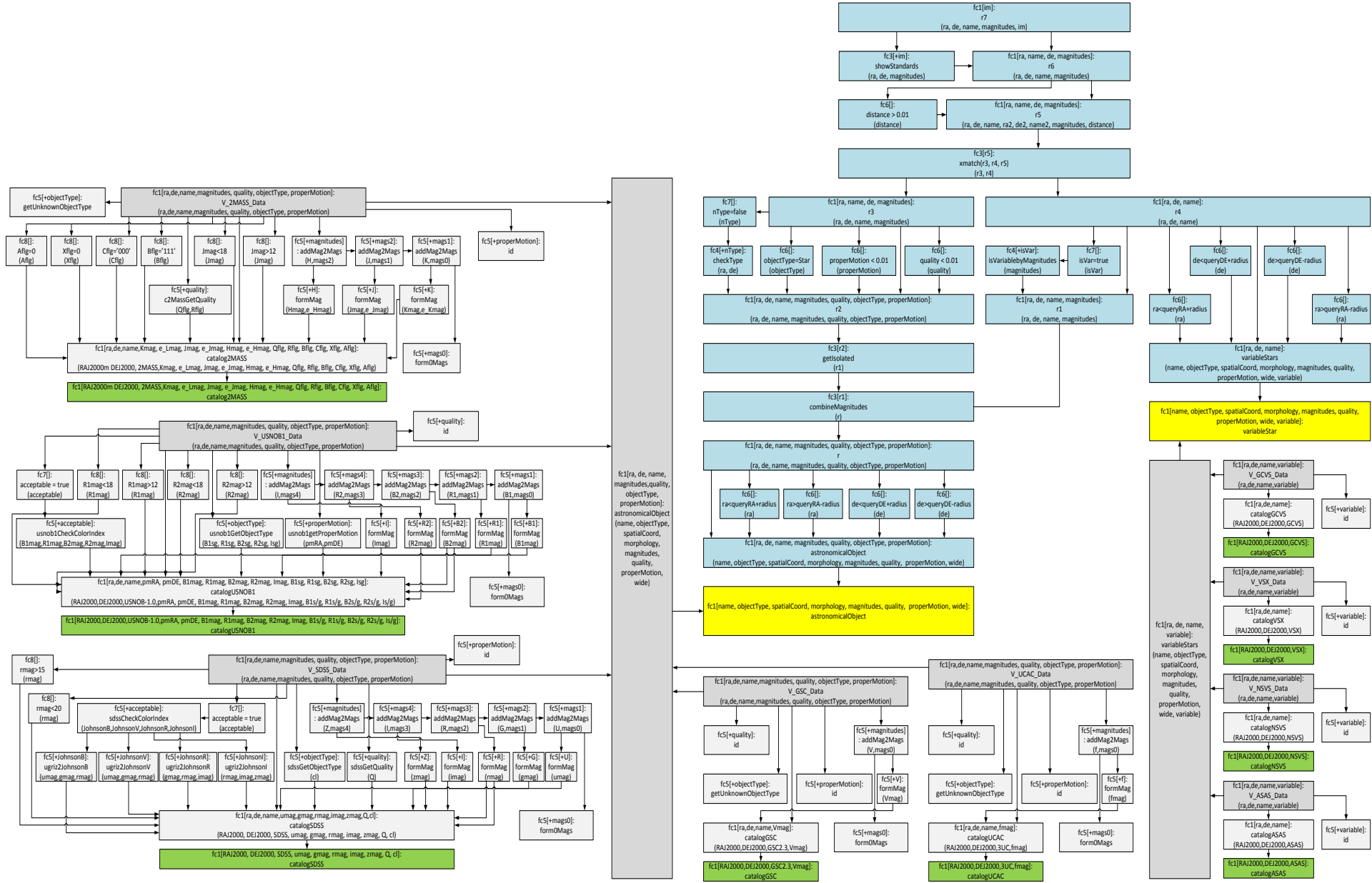
```
select * from emp e dept d
where e.salery>2000
and e.deptno=d.deptno
order by e.name
```





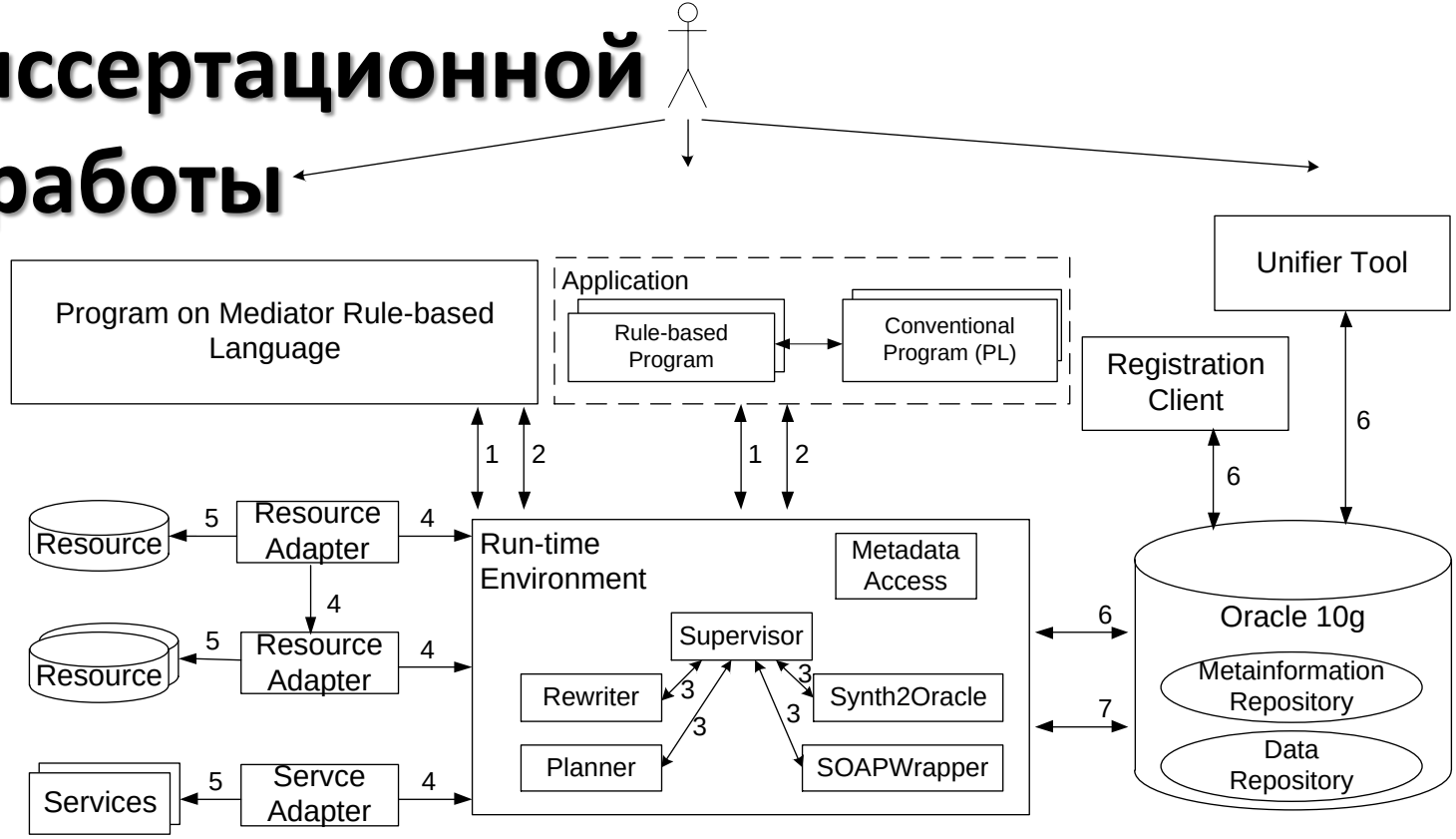


Пример графа зависимостей операций



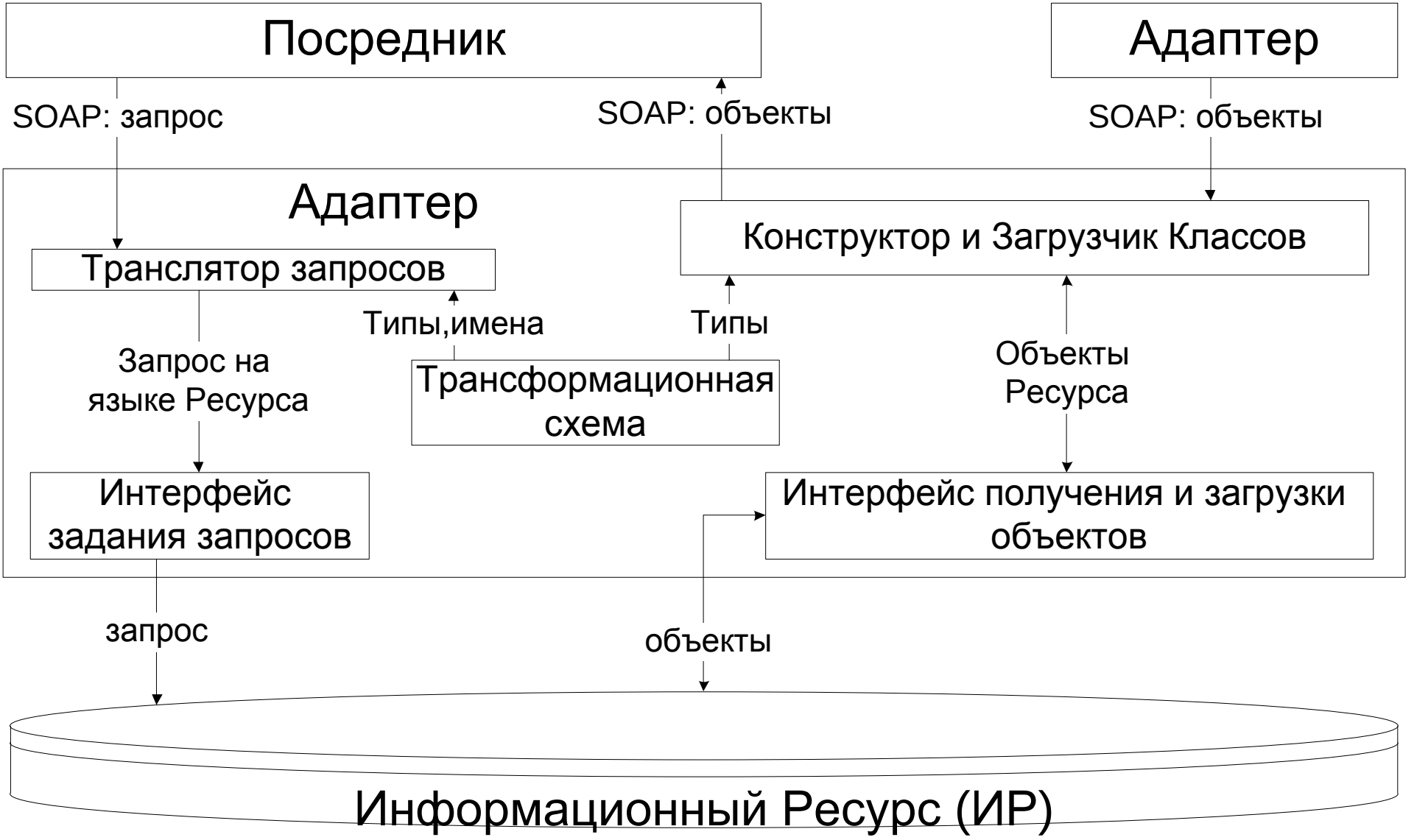
Обзор Диссертационной работы

Работа может быть разделена на три независимые задачи



- Задача построения рассредоточения
- Задача построения адаптеров информационных ресурсов
- Задача связывания декларативного языка предметных посредников и процедурных языков программирования

Построение адаптеров (1)



Построение адаптеров (2)

- Все методы адаптер должны допускать одновременное использование из нескольких потоков (thread safe)
- Поддержка сессий
- Поддержка возможности повторного использования результата запроса
- Поддержка возможности управления загрузкой данных в адаптер, а также возможности материализации запросов в адаптерах
- Поддержка Capabilities адаптеров
- Поддержка оценочных запросов
- Поддержка возможности задания запроса над ограниченной выборкой
- Поддержка статистики
- Программируемые адаптера

Связывание языка правил с ЯП (1)

	Синтакс	Типы	Связывание	Имена	Коллекции	Долговечность	Запросы	Ссылки	Рефакторинг
Тип-в-Тип		٧	٧*		٧*				
Тип-в-Объект									
ТипЯП-в-Тип			٧*		٧*				
Запрос-в-ЯП	٧		٧*	٧			٧		٧
Запрос-строка									
Запрос-объект	٧								٧
ЯМД-строка								٧	
ЯМД-в-ЯП								٧	
Д-Объекты						٧		٧	
Т-Объекты									
У-Коллекции					٧*				

Связывание языка правил с ЯП (2)

- предложена система характеристик (features), которыми может характеризоваться и оцениваться подход к сопряжению
- для решения всех проблем несоответствия импеданса подход к сопряжению должен обладать следующим набором характеристик: **Тип-в-Тип, Запрос-в-ЯП, Долговечные Объекты, Универсальные Коллекции** (статический подход)
- ни один из существующих подходов (ODMG 3.0, JDO, JDBC, SQLJ, OCCI, LINQ) полностью не реализует статический подход
- Предлагается одновременно реализовать как статический подход, преодолевающий несоответствие импеданса, но накладывающий ряд ограничений на возможности посредников, так и динамический подход, никак не ограничивающий возможности посредников

Определения: Граф зависимостей

Определение 2.2. Функциональной операцией будем называть конструкцию программы (спецификации), которой можно манипулировать в процессе построения рассредоточения как атомарной единицей.

Определение 2.3. Если функциональная операция op_1 среди входных параметров имеет выходные параметры функциональной операции op_2, то говорят, что операция op_1 зависит от операции op_2.

Определение 2.4. Граф зависимостей функциональных операций или граф зависимостей – это ориентированный, вообще говоря, несвязный граф, без циклов, в вершинах которого расположены функциональные операции. Вершины в графе именуются также как и операции. Дуги в графе выражают зависимости операций, так что если операция op_1 зависит от операции op_2, то в графе зависимостей существует дуга, направленная от вершины op_1 к вершине op_2.

Определения: Модель рассредоточения

Определение 2.6. Моделью рассредоточения будем называть граф зависимостей функциональных операций, для каждой операции которого определены назначения. Также в модели рассредоточения определены возможные назначения для каждой из операций.

Определение 2.11. Оценкой эффективности рассредоточения будем называть функционал ET (Execution Time), существенно зависящий от рассредоточения, такой что ET = Tplan + TE.

Определение 2.12. Минимальным рассредоточением называется такое рассредоточение, при котором оценка эффективности рассредоточения ET будет минимальной среди всех возможных рассредоточений.

Построение эффективного рассредоточения

- Перестановка операций в модели рассредоточения
- Execution Time (TE) представляет собой ***максимум***, если все ресурсы независимы и возможно параллельное выполнение
- В противном случае execution Time (TE) представляет ***сумму***, а выполнение последовательно

$$T_E = \max_{n=1..k}(T_n^R + T_n^T) + T^M + T^{PL} \qquad T_E = \sum_{n=1}^{k}(T_n^R + T_n^T) + T^M + T^{PL}$$

- k – число информационных ресурсов, T_n^R – время выполнения запроса на n-ом ресурсе, T_n^T – время передачи данных от n-ого ресурса в посредник или другой ресурс, T^M – время выполнения запроса в посреднике, T^{PL} – время выполнения программы в языке программирования.
- Эффективное рассредоточение минимизирует ET (определение 2.11)

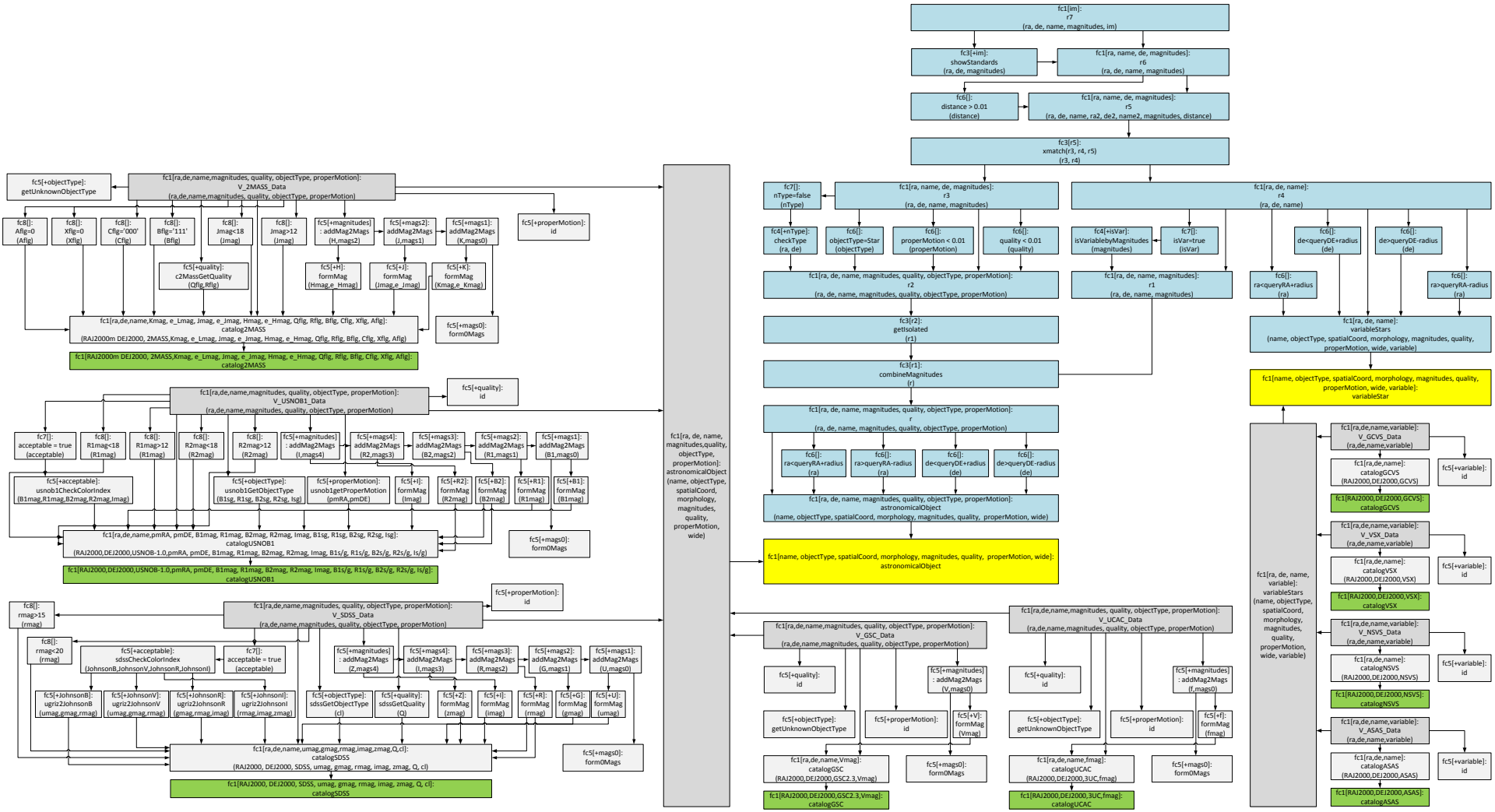
Подход построения эффективного рассредоточения (общий случай)

- Для каждого ЯП определяются его функциональные возможности в виде функциональных операций выразимых на нем
- По множествам функциональных возможностей определяются классы функциональных операций (универсальные конструкты) для группы языков
- Каждая операция в модели рассредоточения, характеризуется своим функциональным классом, а также назначением для реализации
- Множество зависимостей операций в модели рассредоточения представлено дугами
- Для модели рассредоточения и каждого ЯП реализуются трансляторы прямого и обратного отображения

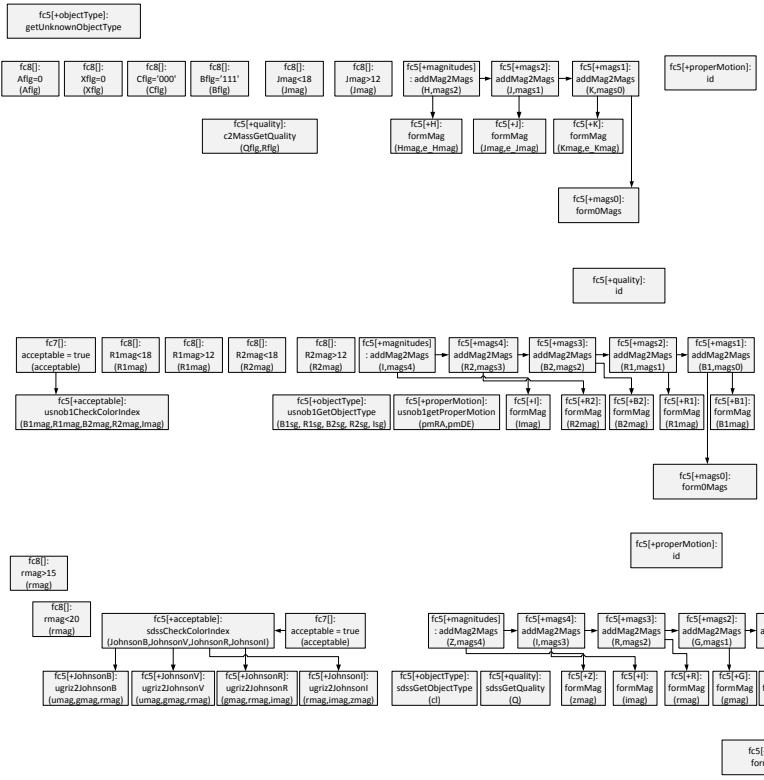
Подход построения эффективного рассредоточения (общий случай)

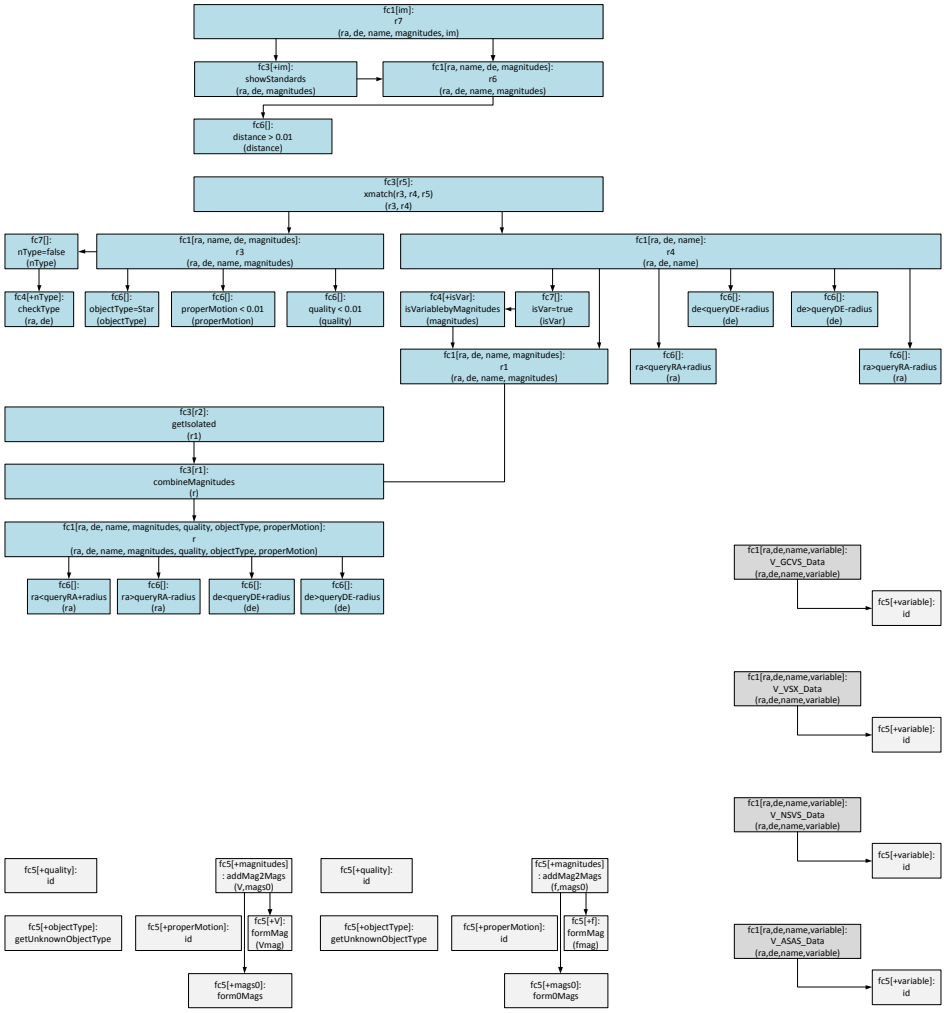
- Исходная спецификация алгоритма решения задачи представленная в многоязычной среде отображается в модель рассредоточения
- Автоматически (применение эвристических правил) или полуавтоматически (с участием эксперта) производится перестановка операций в модели рассредоточения
- Для оценки эффективности модель рассредоточения транслируется в спецификацию алгоритма решения задачи в многоязычной среде, после чего и происходит интерпретация (выполнение)
- В случае полного перебора необходимо проверить все варианты рассредоточений
- Для ускорения оценки эффективности, используется выполнение на ограниченной выборке

Пример

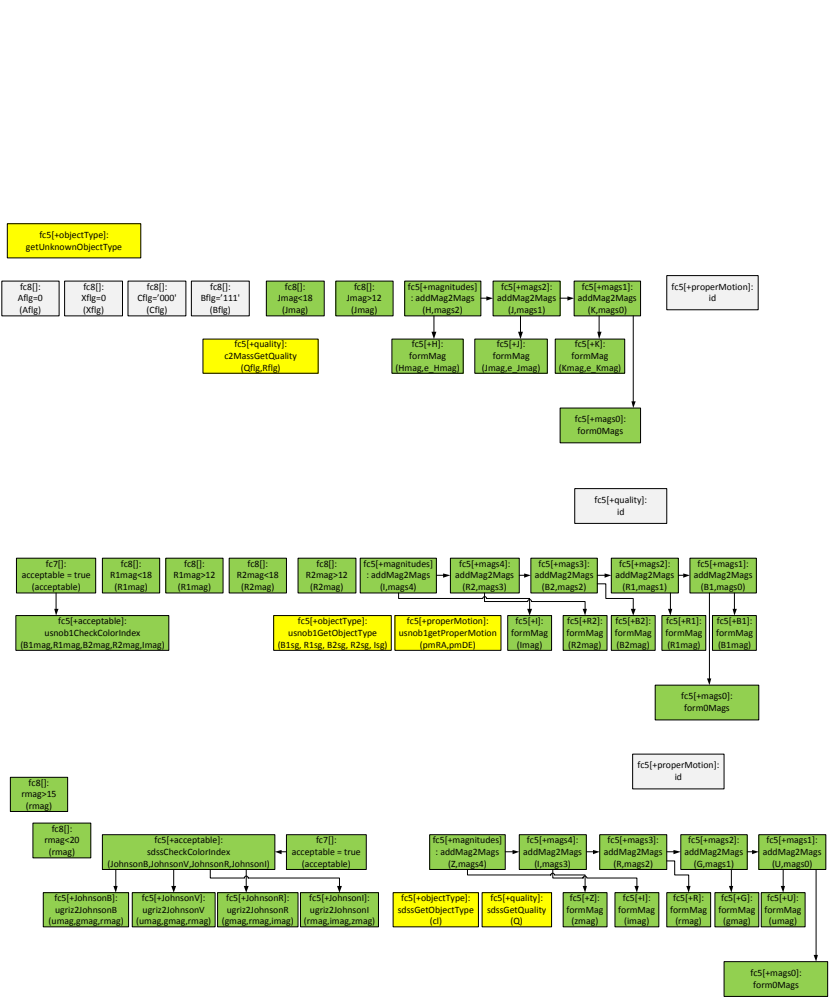

fc1[jm]:
r7
(ra, de, name, magnitudes, im)
fc3[+im]:
show5tandards
(ra, de, magnitudes)
fc1[ra, name, de, magnitudes]:
r6
(ra, de, name, magnitudes)
fc6[]:
distance > 0.01
(distance)
fc1[ra, name, de, magnitudes]:
r5
(ra, de, name, ra2, de2, name2, magnitudes, distance)
fc3[r5]:
xmatch(r3, r4, r5)
(r3, r4)
fc7[]:
nType=false
(nType)
fc1[ra, name, de, magnitudes]:
r3
(ra, de, name, magnitudes)
fc1[ra, de, name]:
r4
(ra, de, name)
fc4[+nType]:
checkType
(ra, de)
fc6[]:
objectType=Star
(objectType)
fc6[]:
properMotion < 0.01
(properMotion)
fc6[]:
quality < 0.01
(quality)
fc4[+isVar]:
isVariablebyMagnitudes
(magnitudes)
fc7[]:
isVar=true
(isVar)
fc6[]:
de<queryDE+radius
(de)
fc6[]:
de>queryDE-radius
(de)
fc6[]:
ra<queryRA+radius
(ra)
fc6[]:
ra>queryRA-radius
(ra)
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]:
r2
(ra, de, name, magnitudes, quality, objectType, properMotion)
fc1[ra, de, name, magnitudes]:
r1
(ra, de, name, magnitudes)
fc3[r2]:
getIsolated
(r1)
fc3[r1]:
combineMagnitudes
(r)
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]:
r
(ra, de, name, magnitudes, quality, objectType, properMotion)
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]:
astronomicalObject
(name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide)
fc1[name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide]:
astronomicalObject
fc1[ra, de, name]:
variableStars
(name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide, variable)
fc1[name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide, variable]:
variableStar
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]:
astronomicalObject
(name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide)
fc5[+objectType]:
getUnknownObjectType
fc1[ra,de,name,magnitudes, quality, objectType, properMotion]:
V_2MASS_Data
(ra,de,name,magnitudes, quality, objectType, properMotion)
fc8[]:
Aflg=0
(Aflg)
fc8[]:
Xflg=0
(Xflg)
fc8[]:
Cflg='000'
(Cflg)
fc8[]:
Bflg='111'
(Bflg)
fc8[]:
Jmag<18
(Jmag)
fc8[]:
Jmag>12
(Jmag)
fc5[+magnitudes]
: addMag2Mags
(H,mags2)
fc5[+mags2]:
addMag2Mags
(J,mags1)
fc5[+mags1]:
addMag2Mags
(K,mags0)
fc5[+properMotion]:
id
fc5[+quality]:
c2MassGetQuality
(Qflg,Rflg)
fc5[+H]:
formMag
(Hmag,e_Hmag)
fc5[+J]:
formMag
(Jmag,e_Jmag)
fc5[+K]:
formMag
(Kmag,e_Kmag)
fc1[ra,de,name,Kmag, e_Lmag, Jmag, e_Jmag, Hmag, e_Hmag, Qflg, Rflg, Bflg, Cflg, Xflg, Aflg]:
catalog2MASS
(RAJ2000m DEJ2000, 2MASS,Kmag, e_Lmag, Jmag, e_Jmag, Hmag, e_Hmag, Qflg, Rflg, Bflg, Cflg, Xflg, Aflg)
fc5[+mags0]:
form0Mags
fc1[RAJ2000m DEJ2000, 2MASS,Kmag, e_Lmag, Jmag, e_Jmag, Hmag, e_Hmag, Qflg, Rflg, Bflg, Cflg, Xflg, Aflg]:
catalog2MASS
fc1[ra,de,name,magnitudes, quality, objectType, properMotion]:
V_USNOB1_Data
(ra,de,name,magnitudes, quality, objectType, properMotion)
fc5[+quality]:
id
fc7[]:
acceptable = true
(acceptable)
fc8[]:
R1mag<18
(R1mag)
fc8[]:
R1mag>12
(R1mag)
fc8[]:
R2mag<18
(R2mag)
fc8[]:
R2mag>12
(R2mag)
fc5[+magnitudes]
: addMag2Mags
(I,mags4)
fc5[+mags4]:
addMag2Mags
(R2,mags3)
fc5[+mags3]:
addMag2Mags
(B2,mags2)
fc5[+mags2]:
addMag2Mags
(R1,mags1)
fc5[+mags1]:
addMag2Mags
(B1,mags0)
fc5[+acceptable]:
usnob1CheckColorIndex
(B1mag,R1mag,B2mag,R2mag,Imag)
fc5[+objectType]:
usnob1GetObjectType
(B1sg, R1sg, B2sg, R2sg, Isg)
fc5[+properMotion]:
usnob1getProperMotion
(pmRA,pmDE)
fc5[+I]:
formMag
(Imag)
fc5[+R2]:
formMag
(R2mag)
fc5[+B2]:
formMag
(B2mag)
fc5[+R1]:
formMag
(R1mag)
fc5[+B1]:
formMag
(B1mag)
fc1[ra,de,name,pmRA, pmDE, B1mag, R1mag, B2mag, R2mag, Imag, B1sg, R1sg, B2sg, R2sg, Isg]:
catalogUSNOB1
(RAJ2000,DEJ2000,USNOB-1.0,pmRA, pmDE, B1mag, R1mag, B2mag, R2mag, Imag, B1s/g, R1s/g, B2s/g, R2s/g, Is/g)
fc5[+mags0]:
form0Mags
fc1[RAJ2000,DEJ2000,USNOB-1.0,pmRA, pmDE, B1mag, R1mag, B2mag, R2mag, Imag, B1s/g, R1s/g, B2s/g, R2s/g, Is/g]:
catalogUSNOB1
fc8[]:
rmag>15
(rmag)
fc1[ra,de,name,magnitudes, quality, objectType, properMotion]:
V_SDSS_Data
(ra,de,name,magnitudes, quality, objectType, properMotion)
fc5[+properMotion]:
id
fc8[]:
rmag<20
(rmag)
fc5[+acceptable]:
sdssCheckColorIndex
(JohnsonB,JohnsonV,JohnsonR,JohnsonI)
fc7[]:
acceptable = true
(acceptable)
fc5[+magnitudes]
: addMag2Mags
(Z,mags4)
fc5[+mags4]:
addMag2Mags
(I,mags3)
fc5[+mags3]:
addMag2Mags
(R,mags2)
fc5[+mags2]:
addMag2Mags
(G,mags1)
fc5[+mags1]:
addMag2Mags
(U,mags0)
fc5[+JohnsonB]:
ugriz2JohnsonB
(umag,gmag,rmag)
fc5[+JohnsonV]:
ugriz2JohnsonV
(umag,gmag,rmag)
fc5[+JohnsonR]:
ugriz2JohnsonR
(gmag,rmag,imag)
fc5[+JohnsonI]:
ugriz2JohnsonI
(rmag,imag,zmag)
fc5[+objectType]:
sdssGetObjectType
(cl)
fc5[+quality]:
sdssGetQuality
(Q)
fc5[+Z]:
formMag
(zmag)
fc5[+I]:
formMag
(imag)
fc5[+R]:
formMag
(rmag)
fc5[+G]:
formMag
(gmag)
fc5[+U]:
formMag
(umag)
fc1[ra,de,name,umag,gmag,rmag,imag,zmag,Q,cl]:
catalogSDSS
(RAJ2000, DEJ2000, SDSS, umag, gmag, rmag, imag, zmag, Q, cl)
fc5[+mags0]:
form0Mags
fc1[RAJ2000, DEJ2000, SDSS, umag, gmag, rmag, imag, zmag, Q, cl]:
catalogSDSS
fc1[ra,de,name,magnitudes, quality, objectType, properMotion]:
V_GSC_Data
(ra,de,name,magnitudes, quality, objectType, properMotion)
fc5[+quality]:
id
fc5[+magnitudes]
: addMag2Mags
(V,mags0)
fc5[+objectType]:
getUnknownObjectType
fc5[+properMotion]:
id
fc5[+V]:
formMag
(Vmag)
fc1[ra,de,name,Vmag]:
catalogGSC
(RAJ2000,DEJ2000,GSC2.3,Vmag)
fc5[+mags0]:
form0Mags
fc1[RAJ2000,DEJ2000,GSC2.3,Vmag]:
catalogGSC
fc1[ra,de,name,magnitudes, quality, objectType, properMotion]:
V_UCAC_Data
(ra,de,name,magnitudes, quality, objectType, properMotion)
fc5[+quality]:
id
fc5[+magnitudes]
: addMag2Mags
(f,mags0)
fc5[+objectType]:
getUnknownObjectType
fc5[+properMotion]:
id
fc5[+F]:
formMag
(fmag)
fc1[ra,de,name,fmag]:
catalogUCAC
(RAJ2000,DEJ2000,3UC,fmag)
fc5[+mags0]:
form0Mags
fc1[RAJ2000,DEJ2000,3UC,fmag]:
catalogUCAC
fc1[ra, de, name, variable]:
variableStars
(name, objectType, spatialCoord, morphology, magnitudes, quality, properMotion, wide, variable)
fc1[ra,de,name,variable]:
V_GCVS_Data
(ra,de,name,variable)
fc1[ra,de,name]:
catalogGCVS
(RAJ2000,DEJ2000,GCVS)
fc5[+variable]:
id
fc1[RAJ2000,DEJ2000,GCVS]:
catalogGCVS
fc1[ra,de,name,variable]:
V_VSX_Data
(ra,de,name,variable)
fc1[ra,de,name]:
catalogVSX
(RAJ2000,DEJ2000,VSX)
fc5[+variable]:
id
fc1[RAJ2000,DEJ2000,VSX]:
catalogVSX
fc1[ra,de,name,variable]:
V_NSV5_Data
(ra,de,name,variable)
fc1[ra,de,name]:
catalogNSVS
(RAJ2000,DEJ2000,NSVS)
fc5[+variable]:
id
fc1[RAJ2000,DEJ2000,NSVS]:
catalogNSVS
fc1[ra,de,name,variable]:
V_ASAS_Data
(ra,de,name,variable)
fc1[ra,de,name]:
catalogASAS
(RAJ2000,DEJ2000,ASAS)
fc5[+variable]:
id
fc1[RAJ2000,DEJ2000,ASAS]:
catalogASAS

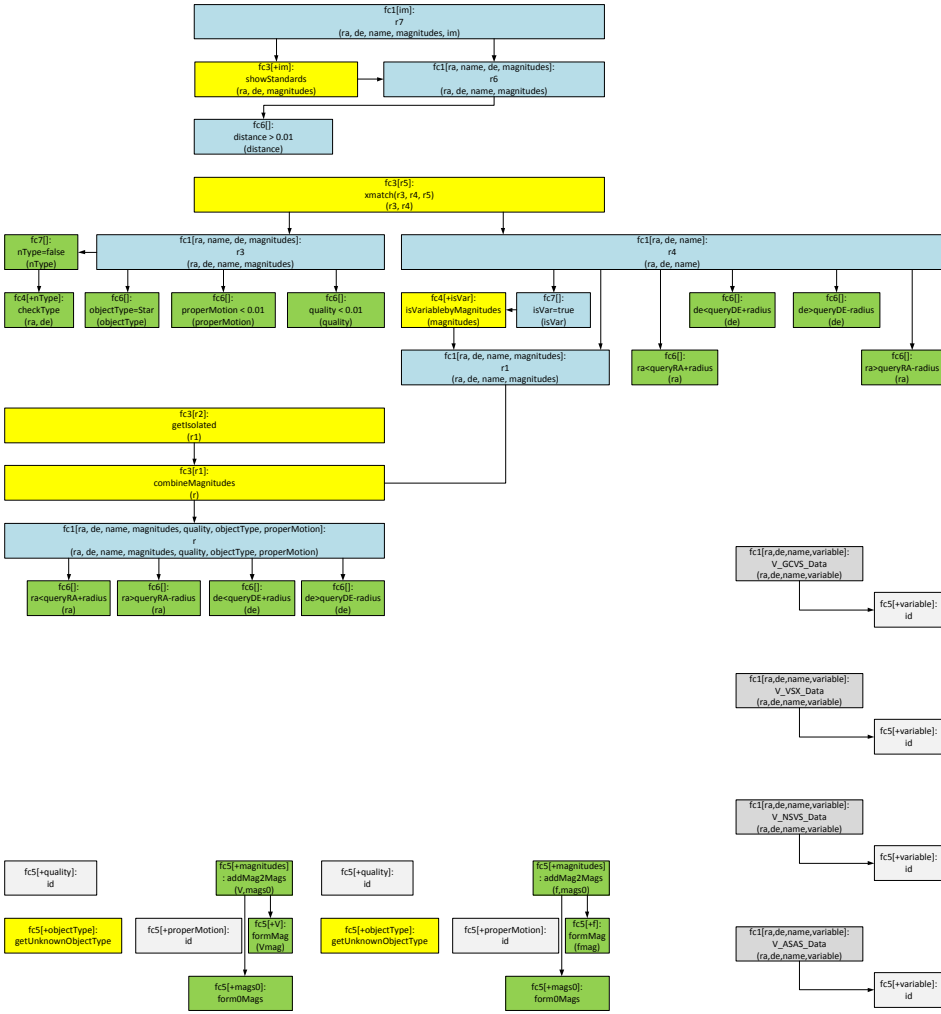
Пример


fc5[+objectType]:
getUnknownObjectType
fc8[]:
Aflg=0
(Aflg)
fc8[]:
Xflg=0
(Xflg)
fc8[]:
Cflg='000'
(Cflg)
fc8[]:
Bflg='111'
(Bflg)
fc8[]:
Jmag<18
(Jmag)
fc8[]:
Jmag>12
(Jmag)
fc5[+magnitudes]
: addMag2Mags
(H,mags2)
fc5[+mags2]:
addMag2Mags
(J,mags1)
fc5[+mags1]:
addMag2Mags
(K,mags0)
fc5[+properMotion]:
id
fc5[+quality]:
c2MassGetQuality
(Qflg,Rflg)
fc5[+H]:
formMag
(Hmag,e_Hmag)
fc5[+J]:
formMag
(Jmag,e_Jmag)
fc5[+K]:
formMag
(Kmag,e_Kmag)
fc5[+mags0]:
form0Mags
fc5[+quality]:
id
fc7[]:
acceptable = true
(acceptable)
fc8[]:
R1mag<18
(R1mag)
fc8[]:
R1mag>12
(R1mag)
fc8[]:
R2mag<18
(R2mag)
fc8[]:
R2mag>12
(R2mag)
fc5[+magnitudes]
: addMag2Mags
(I,mags4)
fc5[+mags4]:
addMag2Mags
(R2,mags3)
fc5[+mags3]:
addMag2Mags
(B2,mags2)
fc5[+mags2]:
addMag2Mags
(R1,mags1)
fc5[+mags1]:
addMag2Mags
(B1,mags0)
fc5[+acceptable]:
usnob1CheckColorIndex
(B1mag,R1mag,B2mag,R2mag,Imag)
fc5[+objectType]:
usnob1GetObjectType
(B1sg, R1sg, B2sg, R2sg, Isg)
fc5[+properMotion]:
usnob1getProperMotion
(pmRA,pmDE)
fc5[+I]:
formMag
(Imag)
fc5[+R2]:
formMag
(R2mag)
fc5[+B2]:
formMag
(B2mag)
fc5[+R1]:
formMag
(R1mag)
fc5[+B1]:
formMag
(B1mag)
fc5[+mags0]:
form0Mags
fc5[+properMotion]:
id
fc8[]:
rmag<15
(rmag)
fc8[]:
rmag<20
(rmag)
fc5[+acceptable]:
sdssCheckColorIndex
(JohnsonB,JohnsonV,JohnsonR,JohnsonI)
fc7[]:
acceptable = true
(acceptable)
fc5[+magnitudes]
: addMag2Mags
(Z,mags4)
fc5[+mags4]:
addMag2Mags
(I,mags3)
fc5[+mags3]:
addMag2Mags
(R,mags2)
fc5[+mags2]:
addMag2Mags
(G,mags1)
fc5[+mags1]:
addMag2Mags
(U,mags0)
fc5[+JohnsonB]:
ugriz2JohnsonB
(umag,gmag,rmag)
fc5[+JohnsonV]:
ugriz2JohnsonV
(umag,gmag,rmag)
fc5[+JohnsonR]:
ugriz2JohnsonR
(gmag,rmag,imag)
fc5[+JohnsonI]:
ugriz2JohnsonI
(rmag,imag,zmag)
fc5[+objectType]:
sdssGetObjectType
(cl)
fc5[+quality]:
sdssGetQuality
(Q)
fc5[+Z]:
formMag
(zmag)
fc5[+I]:
formMag
(imag)
fc5[+R]:
formMag
(rmag)
fc5[+G]:
formMag
(gmag)
fc5[+U]:
formMag
(umag)
fc5[+mags0]:
form0Mags


fc1[im]:
r7
(ra, de, name, magnitudes, im)
fc3[+im]:
showStandards
(ra, de, magnitudes)
fc1[ra, name, de, magnitudes]:
r6
(ra, de, name, magnitudes)
fc6[]:
distance > 0.01
(distance)
fc3[r5]:
xmatch(r3, r4, r5)
(r3, r4)
fc7[]:
nType=false
(nType)
fc1[ra, name, de, magnitudes]:
r3
(ra, de, name, magnitudes)
fc1[ra, de, name]:
r4
(ra, de, name)
fc4[+nType]:
checkType
(ra, de)
fc6[]:
objectType=Star
(objectType)
fc6[]:
properMotion < 0.01
(properMotion)
fc6[]:
quality < 0.01
(quality)
fc4[+isVar]:
isVariablebyMagnitudes
(magnitudes)
fc7[]:
isVar=true
(isVar)
fc6[]:
de<queryDE+radius
(de)
fc6[]:
de>queryDE-radius
(de)
fc1[ra, de, name, magnitudes]:
r1
(ra, de, name, magnitudes)
fc6[]:
ra<queryRA+radius
(ra)
fc6[]:
ra>queryRA-radius
(ra)
fc3[r2]:
getIsolated
(r1)
fc3[r1]:
combineMagnitudes
(r)
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]:
r
(ra, de, name, magnitudes, quality, objectType, properMotion)
fc6[]:
ra<queryRA+radius
(ra)
fc6[]:
ra>queryRA-radius
(ra)
fc6[]:
de<queryDE+radius
(de)
fc6[]:
de>queryDE-radius
(de)
fc1[ra,de,name,variable]:
V_GCVS_Data
(ra,de,name,variable)
fc5[+variable]:
id
fc1[ra,de,name,variable]:
V_VSX_Data
(ra,de,name,variable)
fc5[+variable]:
id
fc1[ra,de,name,variable]:
V_NSVS_Data
(ra,de,name,variable)
fc5[+variable]:
id
fc1[ra,de,name,variable]:
V_ASAS_Data
(ra,de,name,variable)
fc5[+variable]:
id
fc5[+quality]:
id
fc5[+objectType]:
getUnknownObjectType
fc5[+magnitudes]
: addMag2Mags
(V,mags0)
fc5[+properMotion]:
id
fc5[+V]:
formMag
(Vmag)
fc5[+mags0]:
form0Mags
fc5[+quality]:
id
fc5[+objectType]:
getUnknownObjectType
fc5[+magnitudes]
: addMag2Mags
(f,mags0)
fc5[+properMotion]:
id
fc5[+f]:
formMag
(fmag)
fc5[+mags0]:
form0Mags

Пример


fc5[+objectType]: getUnknownObjectType
fc8[]: Aflg=0 (Aflg)
fc8[]: Xflg=0 (Xflg)
fc8[]: Cflg='000' (Cflg)
fc8[]: Bflg='111' (Bflg)
fc8[]: Jmag<18 (Jmag)
fc8[]: Jmag>12 (Jmag)
fc5[+magnitudes] : addMag2Mags (H,mags2)
fc5[+mags2]: addMag2Mags (J,mags1)
fc5[+mags1]: addMag2Mags (K,mags0)
fc5[+properMotion]: id
fc5[+quality]: c2MassGetQuality (Qflg,Rflg)
fc5[+H]: formMag (Hmag,e_Hmag)
fc5[+J]: formMag (Jmag,e_Jmag)
fc5[+K]: formMag (Kmag,e_Kmag)
fc5[+mags0]: form0Mags
fc5[+quality]: id
fc7[]: acceptable = true (acceptable)
fc8[]: R1mag<18 (R1mag)
fc8[]: R1mag>12 (R1mag)
fc8[]: R2mag<18 (R2mag)
fc8[]: R2mag>12 (R2mag)
fc5[+magnitudes] : addMag2Mags (I,mags4)
fc5[+mags4]: addMag2Mags (R2,mags3)
fc5[+mags3]: addMag2Mags (B2,mags2)
fc5[+mags2]: addMag2Mags (R1,mags1)
fc5[+mags1]: addMag2Mags (B1,mags0)
fc5[+acceptable]: usnob1CheckColorIndex (B1mag,R1mag,B2mag,R2mag,Imag)
fc5[+objectType]: usnob1GetObjectType (B1sg, R1sg, B2sg, R2sg, Isg)
fc5[+properMotion]: usnob1getProperMotion (pmRA,pmDE)
fc5[+I]: formMag (Imag)
fc5[+R2]: formMag (R2mag)
fc5[+B2]: formMag (B2mag)
fc5[+R1]: formMag (R1mag)
fc5[+B1]: formMag (B1mag)
fc5[+mags0]: form0Mags
fc5[+properMotion]: id
fc8[]: rmag>15 (rmag)
fc8[]: rmag<20 (rmag)
fc5[+acceptable]: sdssCheckColorIndex (JohnsonB,JohnsonV,JohnsonR,JohnsonI)
fc7[]: acceptable = true (acceptable)
fc5[+magnitudes] : addMag2Mags (Z,mags4)
fc5[+mags4]: addMag2Mags (I,mags3)
fc5[+mags3]: addMag2Mags (R,mags2)
fc5[+mags2]: addMag2Mags (G,mags1)
fc5[+mags1]: addMag2Mags (U,mags0)
fc5[+JohnsonB]: ugriz2JohnsonB (umag,gmag,rmag)
fc5[+JohnsonV]: ugriz2JohnsonV (umag,gmag,rmag)
fc5[+JohnsonR]: ugriz2JohnsonR (gmag,rmag,imag)
fc5[+JohnsonI]: ugriz2JohnsonI (rmag,imag,zmag)
fc5[+objectType]: sdssGetObjectType (cl)
fc5[+quality]: sdssGetQuality (Q)
fc5[+Z]: formMag (zmag)
fc5[+I]: formMag (imag)
fc5[+R]: formMag (rmag)
fc5[+G]: formMag (gmag)
fc5[+U]: formMag (umag)
fc5[+mags0]: form0Mags


fc1[im]: r7 (ra, de, name, magnitudes, im)
fc3[+im]: showStandards (ra, de, magnitudes)
fc1[ra, name, de, magnitudes]: r6 (ra, de, name, magnitudes)
fc6[]: distance > 0.01 (distance)
fc3[r5]: xmatch(r3, r4, r5) (r3, r4)
fc7[]: nType=false (nType)
fc1[ra, name, de, magnitudes]: r3 (ra, de, name, magnitudes)
fc1[ra, de, name]: r4 (ra, de, name)
fc4[+nType]: checkType (ra, de)
fc6[]: objectType=Star (objectType)
fc6[]: properMotion < 0.01 (properMotion)
fc6[]: quality < 0.01 (quality)
fc4[+isVar]: isVariablebyMagnitudes (magnitudes)
fc7[]: isVar=true (isVar)
fc6[]: de<queryDE+radius (de)
fc6[]: de>queryDE-radius (de)
fc1[ra, de, name, magnitudes]: r1 (ra, de, name, magnitudes)
fc6[]: ra<queryRA+radius (ra)
fc6[]: ra>queryRA-radius (ra)
fc3[r2]: getIsolated (r1)
fc3[r1]: combineMagnitudes (r)
fc1[ra, de, name, magnitudes, quality, objectType, properMotion]: r (ra, de, name, magnitudes, quality, objectType, properMotion)
fc6[]: ra<queryRA+radius (ra)
fc6[]: ra>queryRA-radius (ra)
fc6[]: de<queryDE+radius (de)
fc6[]: de>queryDE-radius (de)
fc1[ra,de,name,variable]: V_GCVS_Data (ra,de,name,variable)
fc5[+variable]: id
fc1[ra,de,name,variable]: V_VSX_Data (ra,de,name,variable)
fc5[+variable]: id
fc1[ra,de,name,variable]: V_NSVS_Data (ra,de,name,variable)
fc5[+variable]: id
fc1[ra,de,name,variable]: V_ASAS_Data (ra,de,name,variable)
fc5[+variable]: id
fc5[+quality]: id
fc5[+objectType]: getUnknownObjectType
fc5[+magnitudes] : addMag2Mags (V,mags0)
fc5[+properMotion]: id
fc5[+V]: formMag (Vmag)
fc5[+mags0]: form0Mags
fc5[+quality]: id
fc5[+objectType]: getUnknownObjectType
fc5[+magnitudes] : addMag2Mags (f,mags0)
fc5[+properMotion]: id
fc5[+f]: formMag (fmag)
fc5[+mags0]: form0Mags

Пример во взглядах

```
//    GAV for SDSS
    v_SDSS_Data(x/[ra, de, name, magnitudes, quality, objectType, properMotion])
 :- CATALOG_SDSS.catalogSDSS(x/[ra:RAJ2000, de:DEJ2000, name:SDSS, umag, gmag, rmag,
imag, zmag, Q, cl])
    & ProgramableA.ugriz2JohnsonB(umag, gmag, rmag, JohnsonB)
    & ProgramableA.ugriz2JohnsonV(umag, gmag, rmag, JohnsonV)
    & ProgramableA.ugriz2JohnsonR(gmag, rmag, imag, JohnsonR)
    & ProgramableA.ugriz2JohnsonI(rmag, imag, zmag, JohnsonI)
    & ProgramableA.formMag(umag, 0, 'UGRIZ_U', U)
    & ProgramableA.formMag(gmag, 0, 'UGRIZ_G', G)
    & ProgramableA.formMag(rmag, 0, 'UGRIZ_R', R)
    & ProgramableA.formMag(imag, 0, 'UGRIZ_I', I)
    & ProgramableA.formMag(zmag, 0, 'UGRIZ_Z', Z)
    & ProgramableA.form0Mags(mags0)
    & ProgramableA.addMag2Mags(U, mags0, mags1)
    & ProgramableA.addMag2Mags(G, mags1, mags2)
    & ProgramableA.addMag2Mags(R, mags2, mags3)
    & ProgramableA.addMag2Mags(I, mags3, mags4)
    & ProgramableA.addMag2Mags(Z, mags4, magnitudes)
    & ProgramableA.sdssGetQuality(Q, quality)
    & ProgramableA.sdssGetObjectType(cl, objectType)
    & ProgramableA.sdssCheckColorIndex(JohnsonB, JohnsonV, JohnsonR, JohnsonI,
acceptable)
    & acceptable = true
    & id(0, properMotion)
```

Пример во взглядах

```
//    GAV for SDSS
    v_SDSS_Data(x/[ra, de, name, magnitudes, quality, objectType, properMotion])
 :- CATALOG_SDSS.catalogSDSS(x/[ra:RAJ2000, de:DEJ2000, name:SDSS, umag, gmag, rmag,
imag, zmag, Q, cl])
    & CATALOG_SDSS.getMagnitudes(umag, gmag, rmag, imag, zmag, magnitudes)
    & CATALOG_SDSS.sdssGetQuality(Q, quality)
    & CATALOG_SDSS.sdssGetObjectType(cl, objectType)
    & CATALOG_SDSS.sdssCheckColorIndex(umag, gmag, rmag, imag, zmag, acceptable)
    & acceptable = true
    & id(0, properMotion)
```

Пример в правилах и ЯП (1)

```
1      r(x/[ra, de, name, magnitudes, objectType, properMotion, quality])
:- astronomicalObject(x1/[ra: spatialCoord.ra, de: spatialCoord.de, objectType,
properMotion, quality, magnitudes])
& ra < queryRA + radius & ra > queryRA - radius
& de < queryDE + radius & de > queryDE – radius;

2      combineMagnitudes (r/AstronomicalObject, r1);

3      getIsolated(r1, r2);

4      r3(x/[ra, de, name, magnitudes])
:- r2(x1/[ra, de, name, objectType, properMotion, quality, magnitudes])
& checkType(ra, de, 'G', nType) & nType = false
& objectType = Star
& properMotion < 0.01
& quality  < 0.01;
```

Пример в правилах и ЯП (2)

```
5      r4(x/[ra, de, name])
:- r1(x1/[ra, de, name, magnitudes])
& isVariablebyMagnitudes(ra, de, isVar) & isVar = true;

6      r4(x/[ra, de, name])
:- variableStar(x1/[ra: spatialCoord.ra, de: spatialCoord.de,
name]);
& ra < queryRA + radius & ra > queryRA - radius
& de < queryDE + radius & de > queryDE – radius;

7      xmatch(r3, r4, r5);

8      r6(x/[ra, de, name magnitudes])
:- r5(x1/[ra, de, name, magnitudes, distance])
& distance > 0.01;

9      r7(im/Image)
:- r6(x/ra, de, name, magnitudes])
& showStadards(ra, de, radius, magnitudes, im);
```

Пример в правилах и ЯП (3)

```
1      # Synchronized Parallel
Aladin.getImage(queryRA, queryDE, radius);

2      # Synchronized Parallel
Set<String> galaxies = GetNedGalaxies(queryRA, queryDE, radius);

3      # Synchronized Parallel
#SYNTHESIS
Class r(x/[ra, de, name, magnitudes])
:- astronomicalObject(x1/[ra: spatialCoord.ra, de:
spatialCoord.de, name, objectType,
properMotion, quality, magnitudes])
& ra < queryRA + radius & ra > queryRA - radius
& de < queryDE + radius & de > queryDE – radius;
& objectType = Star
& properMotion < 0.01
& quality  < 0.01;
#SYNTHESIS
```

Пример в правилах и ЯП (4)

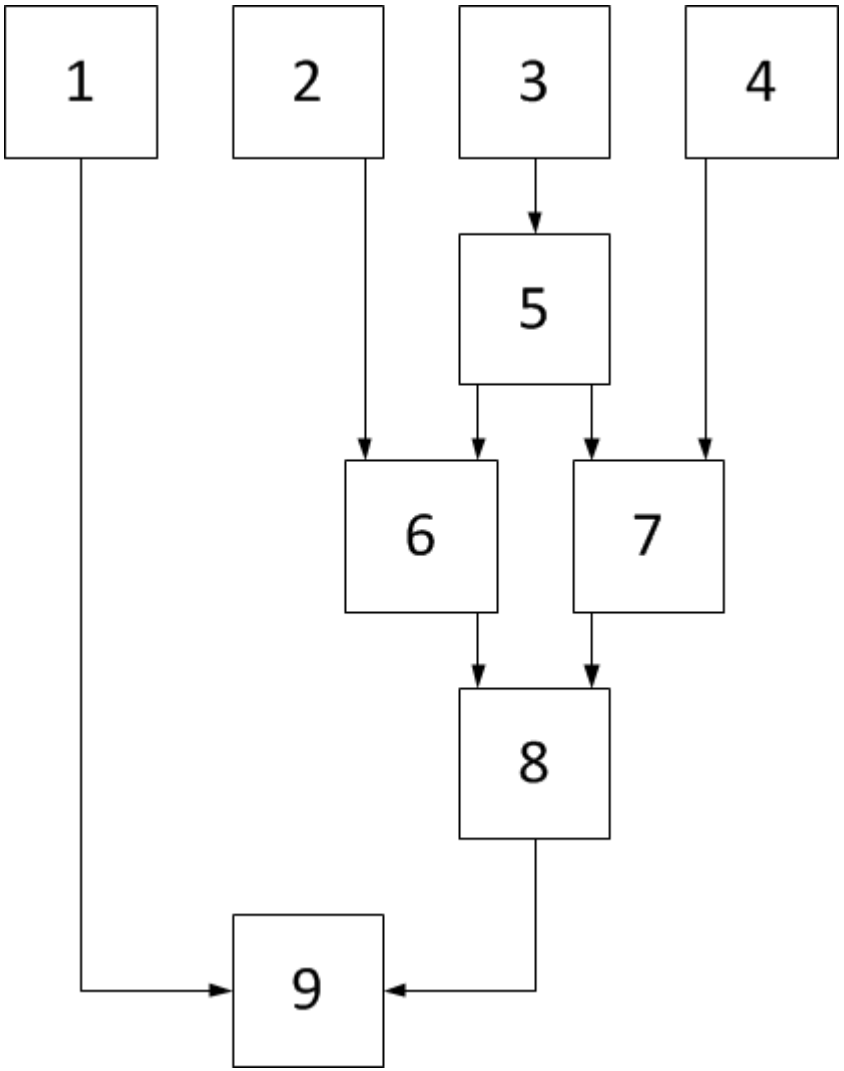
```
4      # Synchronized Parallel
#SYNTHESIS
Class r4(x/[ra, de, name])
:- variableStar(x1/[ra: spatialCoord.ra, de: spatialCoord.de,
name]);
& ra < queryRA + radius & ra > queryRA - radius
& de < queryDE + radius & de > queryDE – radius;
       #SYNTHESIS

5      Set<Class_r> r1 = combineMagnitudes (r);
```

Пример в правилах и ЯП (5)

```
6      # Synchronized Parallel
Set<Class_r> r2 = getIsolated(r1);
       Set<Class_r> r3 = getNotGalaxie(r2, galaxies);

7      # Synchronized Parallel
Set<Class_r4> r5 = getVariablebyMagnitudes(r1);
       r5.add(r4);

8      Set<Class_r> r6 = xmatch(r3, r5, 0.01);

9   Aladin.putStandardsOnImage(r6);
```

подобное изменение программы позволило улучшить среднее время выполнения почти в 10 раз

Пример в правилах и ЯП (6)



Выводы

- Представлена общая постановка задачи рассредоточения
- Даны понятия, необходимые для постановки задачи рассредоточения:
  - модель рассредоточения,
  - рассредоточенная реализации или рассредоточение,
  - минимальное рассредоточение.
- Для обобщенной среды предметных посредников определены классы функциональных операций
- На основе анализа возможности перестановок операций, разработан набор экспертных правил для оптимизированного построения эффективного рассредоточения
- Разработан метод полного перебора для автоматического построения эффективного рассредоточения
- Разработан метод полуавтоматического построения эффективного рассредоточения

Научная новизна диссертационной работы

- подход к построению квази-минимального рассредоточения реализации алгоритма решения задач, позволяющий генерировать варианты рассредоточений, а также оценивать их эффективность;

- предложен подход сопряжения систем программирования с предметными посредниками, включающий реализаций как статического подхода, ориентированного на решения максимального числа проблем не соответствия импеданса, так и динамического подхода, ориентированного на предоставление максимальных возможностей для разработчиков;

- разработана архитектура программируемых адаптеров информационных ресурсов, обеспечивающая эффективное рассредоточение реализации программ посредников среди ресурсов и адаптеров, а также подход к полуавтоматическому созданию адаптеров.

Спасибо за внимание!
Вопросы?