ВСЕРОССИЙСКАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ

НАУЧНЫЙ СЕВИС В СЕТИ ИНТЕРНЕТ

Автоматизация распараллеливания программных комплексов

23 сентября 2016 г. | Абрау-Дюрсо

В.А. Бахтин, О.Ф. Жукова,
Н.А. Катаев, А.С. Колганов, В.А. Крюков,
Н.В. Поддерюгина, М.Н. Притула, О.А. Савицкая, А.А. Смирнов

Распределенная память – это сложно …

- Вычислительная работа должна распределяться между процессорами крупными порциями.
- Необходимо произвести не только распределение вычислений, но и распределение данных, а также обеспечить на каждом процессоре доступ к удаленным данным.
- Распределение вычислений и данных должно быть произведено согласованно.

Распределенная память – это сложно …

- Вычислительная работа должна распределяться между процессорами крупными порциями.
- Необходимо произвести не только распределение вычислений, но и распределение данных, а также обеспечить на каждом процессоре доступ к удаленным данным.
- Распределение вычислений и данных должно быть произведено согласованно.

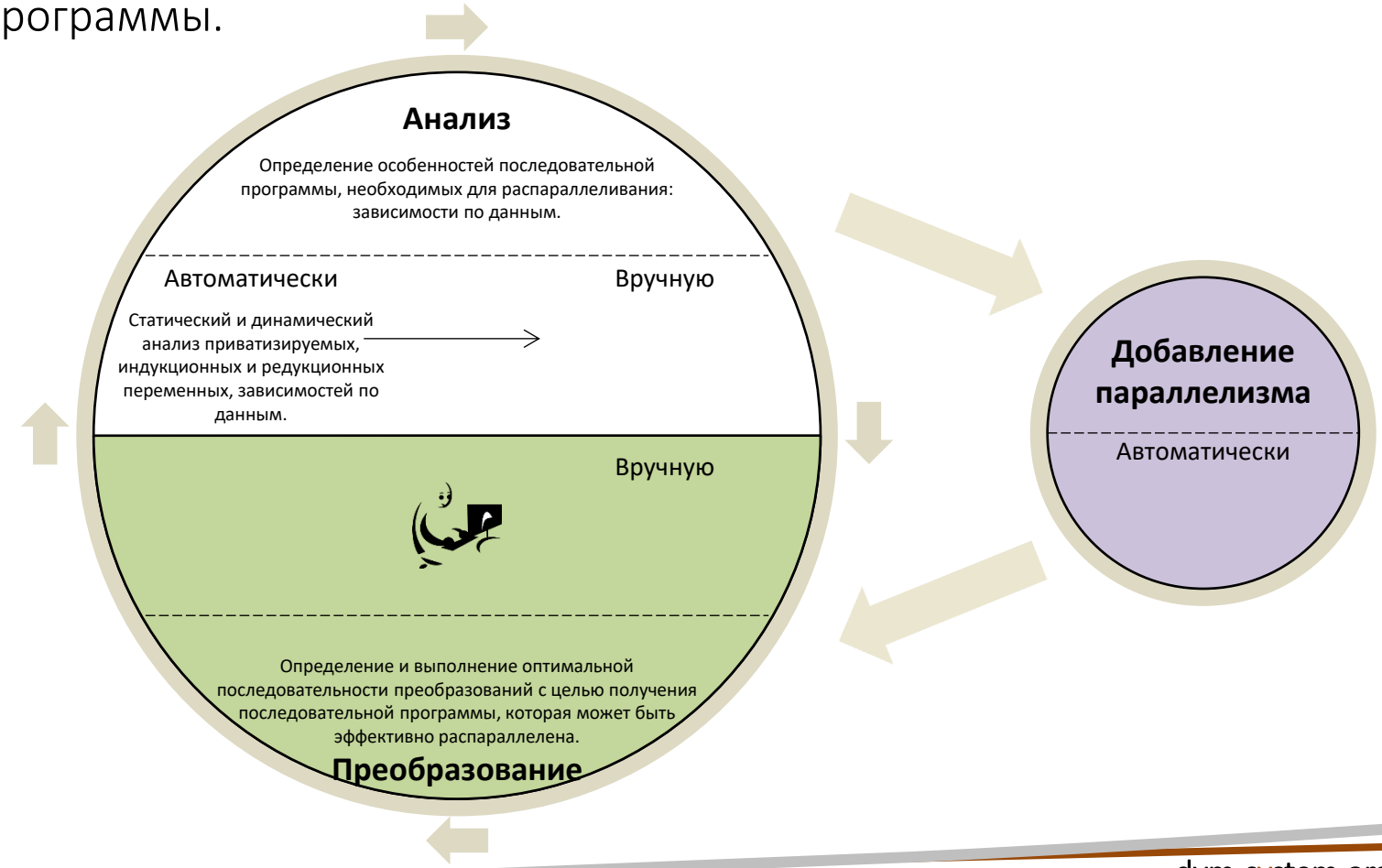
А для программных комплексов еще сложнее!

- Многомодульность, многофункциональность, многоязыковость, использование библиотечных функций.
- Наложение по памяти разных массивов, отличающихся по своей форме или типу элементов.
- Ввод/вывод и аварийный останов программ, выполняемый редко и только на отдельных итерациях.

B.Armstrong and R.Eigenmann. Challenges in the automatic parallelization of large-scale computational applications. In Proceedings of SPIE/ITCOM 2001, volume 4528, pages 50– 60, Aug. 2001.

Автоматизация распараллеливания: САПФОР

- Помогает программисту эффективно отображать его программы на многоядерные кластеры с ускорителями.
- Организует параллельные вычисления полностью автоматически.
- Взаимодействует с программистом в терминах последовательной программы.



Поиск последовательности преобразований

– *автоматический выбор способа устранение проблемы*

Предусмотрен набор оптимизационных последовательностей, которые применяются, если указанная проблема удовлетворяет определенным критериям.

```
do j = 1,ny
  do i = 0,nx
    hy4 = hy(j+1)+2.d0*hy(j)+hy(j-1)
    ...
    Frox = ...
    ro1(i,j) = ro1(i,j) + Frox
    ro1(i+1,j) = ro1(i+1,j) - Frox
    E1(i,j) = E1(i,j) + FEx
    E1(i+1,j) = E1(i+1,j) - FEx
  enddo
enddo
```

⇨

```
do j = 1,ny
  do i = 0,nx
    hy4 = hy(j+1)+2.d0*hy(j)+hy(j-1)
    ...
    Frox = ...
    tmp1(i+1,j) = Frox
    ro1(i+1,j) = ro1(i+1,j) – Frox
    E1(i,j) = E1(i,j) + FEx
    E1(i+1,j) = E1(i+1,j) - FEx
  enddo
enddo

do i = 1,nx
  do j = 0, ny
    ro1(i,j) = ro1(i,j) + tmp1(i+1,j)
  enddo
enddo
```

«данные для записи витка ro1(i, j) и ro(i,j+1) находятся на разных процессорах»

Поиск последовательности преобразований

– *автоматический выбор способа устранение проблемы*

Предусмотрен набор оптимизационных последовательностей, которые применяются по критериям.

```
do j = 1,ny
  do i = 0,nx
    hy4 = hy(j+1)+
    ...
    Frox = ...
    ro1(i,j) = ro1(i
    ro1(i+1,j) = ro
    E1(i,j) = E1(i,j)
    E1(i+1,j) = E1(i
  enddo
enddo
```

«данные для
находятся на

Visualizer - [cavity_step_2]

File Project View Tools Panels Window Help

Project cavity_step_2
Source files
cavity_step_2.for

```
      Wy = tau*(rm*uxm*(uy(i+1,j)-uy(i,j))/h
     *     + (p(i+1,j+1)+p(i,j+1)-p(i+1,j-1)-p(i,j-1))/hy4
     *     + rm*uym*(uy(i+1,j+1)+uy(i,j+1)-uy(i+1,j-1)-uy(i,j-1))
     *     / hy4)
      Rs = tau*(uxm*(p(i+1,j)-p(i,j))/h+gm*pm*divu
     *     + uym*(p(i+1,j+1)+p(i,j+1)-p(i+1,j-1)-p(i,j-1))/hy4)
      Pxx = 2.d0*visc*(ux(i+1,j)-ux(i,j))/h-2.d0/3.d0*visc*divu
     *      + uxm*Wx+Rs
      Pxy = uxm*Wy + visc*((uy(i+1,j)-uy(i,j))/h
     *      + (ux(i+1,j+1)+ux(i,j+1)-ux(i+1,j-1)-ux(i,j-1))/hy4)
      Wx = ((ro(i+1,j)*ux(i+1,j)*ux(i+1,j)
     *     - ro(i,j)*ux(i,j)*ux(i,j)+p(i+1,j)-p(i,j))/h
     *     + (ro(i+1,j+1)*ux(i+1,j+1)*uy(i+1,j+1)
     *     + ro(i,j+1)*ux(i,j+1)*uy(i,j+1) - ro(i+1,j-1)*ux(i+1,j-1)
     *     * uy(i+1,j-1) - ro(i,j-1)*ux(i,j-1)*uy(i,j-1))/hy4)*tau
     *     / rm
      Fmx = rm*(uxm - Wx)
      Frox = - Fmx*hy(j)
      Ent = ((E(i+1,j)+E(i,j))*0.5d0 + pm)/rm
      FEx = (Pxx*uxm + Pxy*uym - Qx - Ent*Fmx)*hy(j)
      ro1(i,j) = ro1(i,j) + Frox
      ro1(i+1,j) = ro1(i+1,j) - Frox
      E1(i,j) = E1(i,j) + FEx
      E1(i+1,j) = E1(i+1,j) - FEx
    enddo
  enddo
c
```

Loop / PU	Progra...	File name	dvm_xp_0 Decisions	Write varia
i=0,nx	cavity	cavity_step_2.for (202)	часть гнезда, данные для записи витка ro1(i,j) и ro1(i+1,j) находятся на разных процессорах;	
i=0,nx				
j=0,ny	cavity	cavity_step_2.for (259)	часть гнезда, данные для записи витка ro1(i,j) и ro1(i,j+1) находятся на разных процессорах;	
j=0,ny				
i=0,nx	cavity	cavity_step_2.for (345)	часть гнезда, данные для записи витка ux1(i,j) и ux1(i+1,j) находятся на разных процессорах;	
i=0,nx				

Output Decisions

Ready Ln 202

Поиск последовательности преобразований

– *активное вовлечение пользователя в процесс выбора*

Система предоставляет пользователю набор элементарных преобразований, которые могут быть выполнены автоматически.

```
for (I = 1; I < Nx + 1; ++I)
  for (J = 1; J < Nz + 1; ++J)
    #pragma x unroll
    for (II = 0; II < 2; ++II)
      for (JJ = 0; JJ < 2; ++JJ) {
        …
        Flx[I][J - 1 + JJ] = Flx[I][J - 1 + JJ] - Iloc[0] * Fl[0];
        Flz[I - 1 + II][J] = Flz[I - 1 + II][J] - Jloc[1] * Fl[1];
      }
```

```
for (I = 1; I < Nx + 1; ++I)
  for (J = 1; J < Nz + 1; ++J) {
    …
    Flx[I][J - 1] = Flx[I][J - 1] - 1 * Fl[0];
    Flz[I - 1][J] = Flz[I - 1][J] - 1 * Fl[1];
    …
    Flx[I][J] = Flx[I][J] - 1 * Fl[0];
    Flz[I - 1][J] = Flz[I - 1][J] - -1 * Fl[1];
    …
    Flx[I][J - 1] = Flx[I][J - 1] - -1 * Fl[0];
    Flz[I][J] = Flz[I][J] - 1 * Fl[1];
    …
    Flx[I][J] = Flx[I][J] - -1 * Fl[0];
    Flz[I][J] = Flz[I][J] - -1 * Fl[1];
}
```

#pragma x unroll [(index-list)]

index ::= var : int-value

Двухмерная задача моделирования процессов плавления многокомпонентных порошков при селективном лазерном спекании на основе многокомпонентной и многофазной гидродинамической модели.

Поиск последовательности преобразований

– *активное вовлечение пользователя в процесс выбора*

Сис… …й,
кото…

```
for (
 for (
  #p
  for
   fo
    }
```

Преобразования	Последовательность элементарных преобразований для программы Спекание 2D									Итого
	1	2	3	4	5	6	7	8	9	
Подстановка переменных	8									8
Вынесение инвариантов	1									1
Разворачивание циклов	5	1					4			10
Распределение циклов			1	2				2		5
Перестановка гнезд циклов					1				2	3
Объединение циклов					4				2	6
Перестановка циклов в гнезде										0
Сдвиг индекса						2				2
Итого	14	1	1	2	5	2	4	2	4	35

Двухмерная задача моделирования процессов плавления многокомпонентных порошков при селективном лазерном спекании на основе многокомпонентной и многофазной гидродинамической модели.

Инкрементальное распараллеливание

- Поэтапное распараллеливания, начиная с времяемких фрагментов.
- Постепенное добавления новых областей распараллеливания с сохранением уже найденных решений для ранее объявленных областей.
- Повышения качества распараллеливания выделенных областей за счет отказа от распараллеливания сложных фрагментов программы.
- Возможность ручного распараллеливания некоторых фрагментов программы и учет принятых программистом решений при распараллеливании других фрагментов.

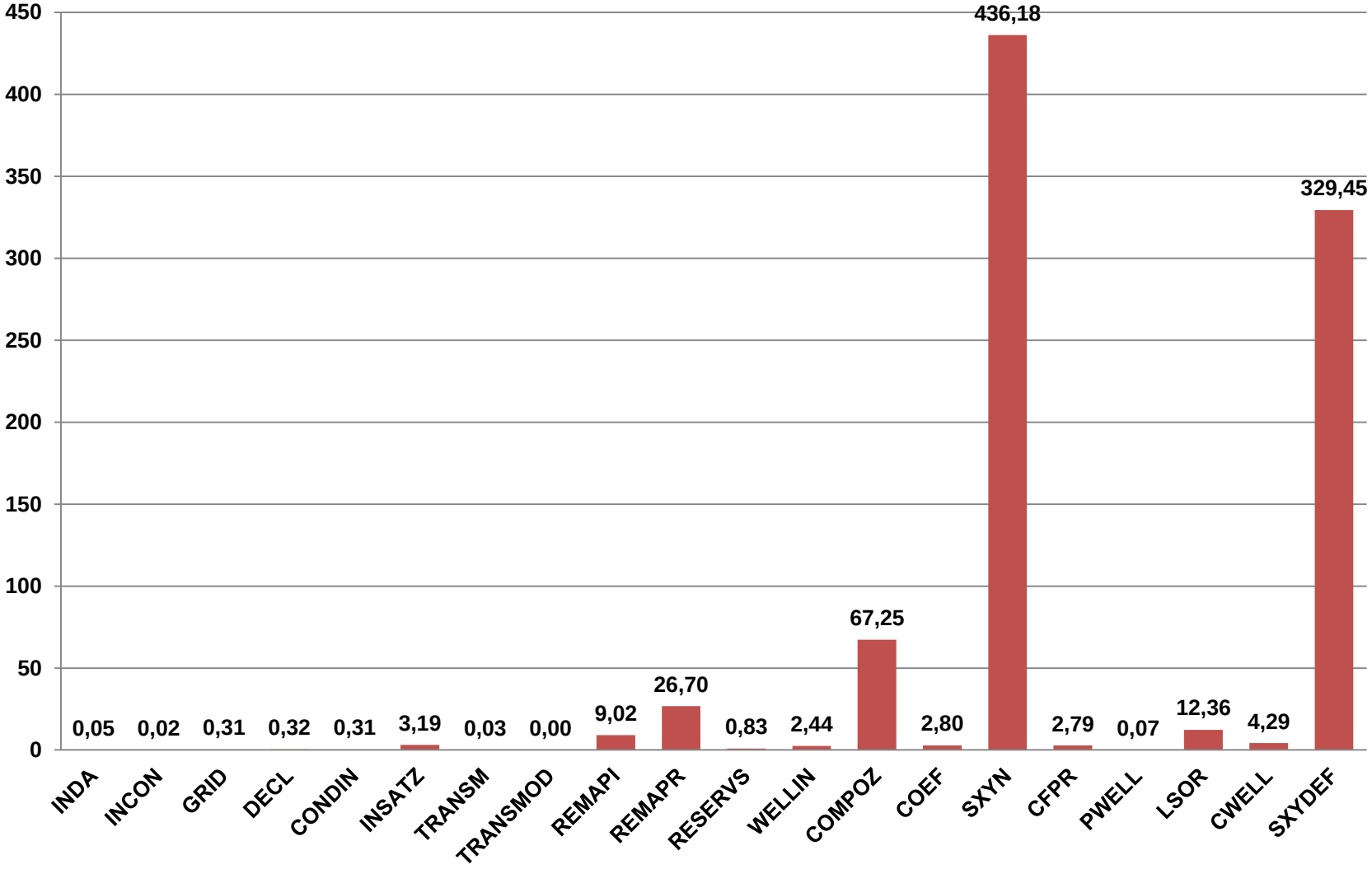
Композиционная модель

- Разработка в режиме истощения залежей летучей нефти и газоконденсатных залежей, где состав и свойства фаз сильно меняются с изменением давления.
- Закачка неравновесного газа (сухого или обогащенного) для мобилизации нефти за счет испарения в более подвижную газовую фазу или за счет конденсации в жидкую фазу промежуточных компонентов и достижения смесимости.
- Закачка в нефтяную залежь углекислого газа для мобилизации нефти за счет испарения легких фракций, уменьшения вязкости и набухания нефти.

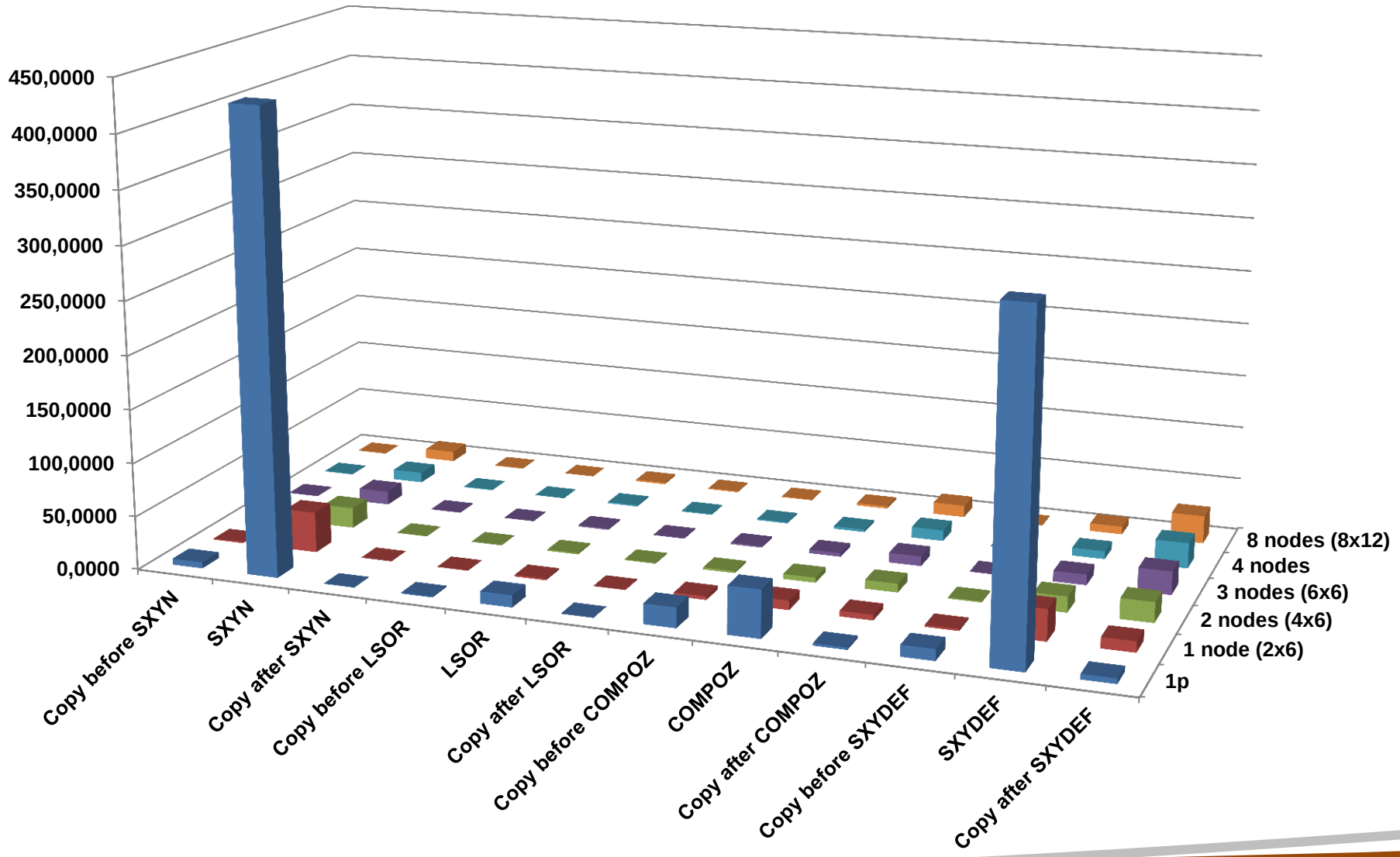
Распараллеливание

- Определение времяемких фрагментов программы.
- Распараллеливание времяемких фрагментов программы.
- Если распределяемые данные используются в других фрагментах программы, то требуется завести массивы-копии.
- Добавление распараллеливаемых фрагментов с сохранением уже найденных решений.

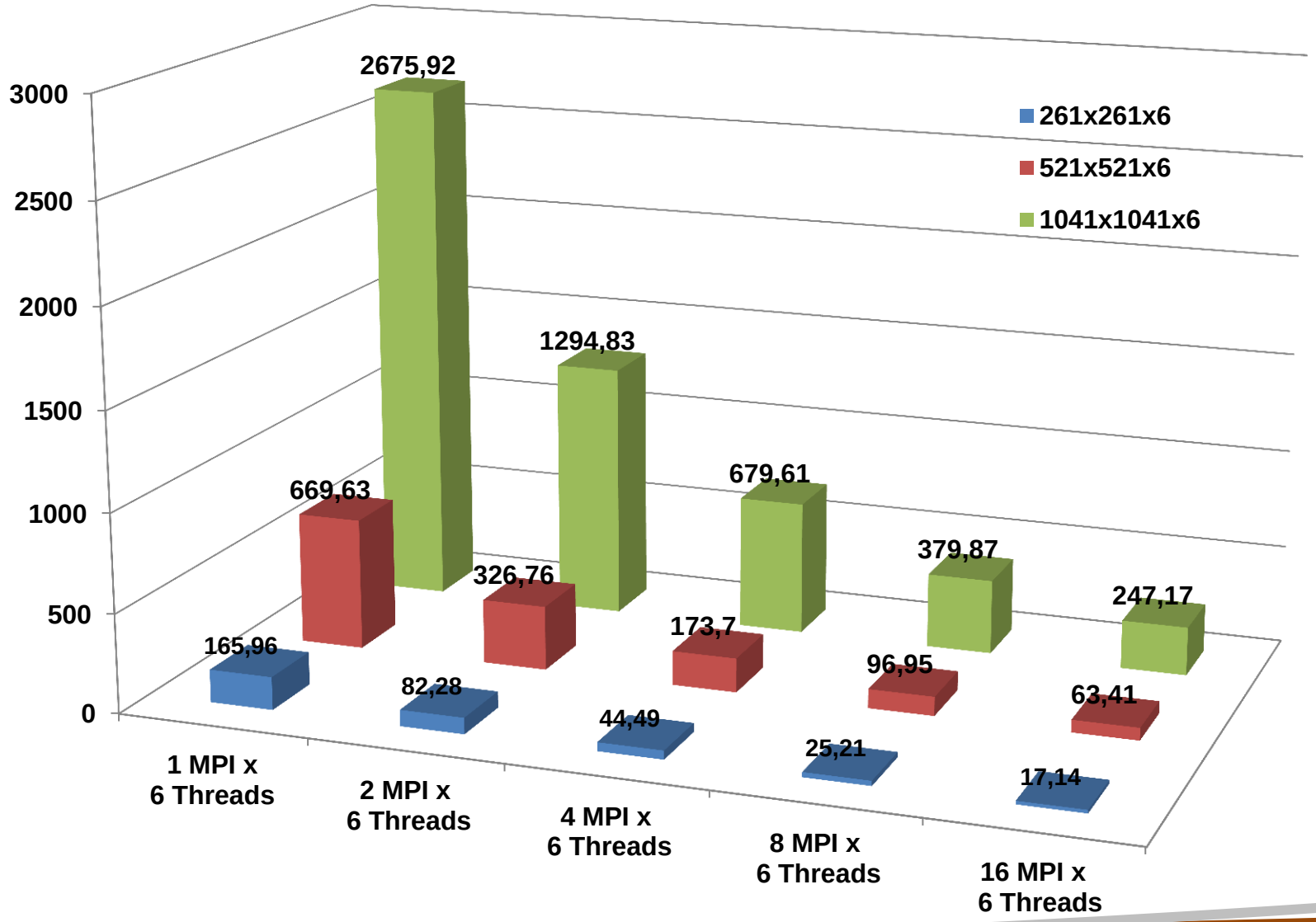
Времена выполнения различных процедур



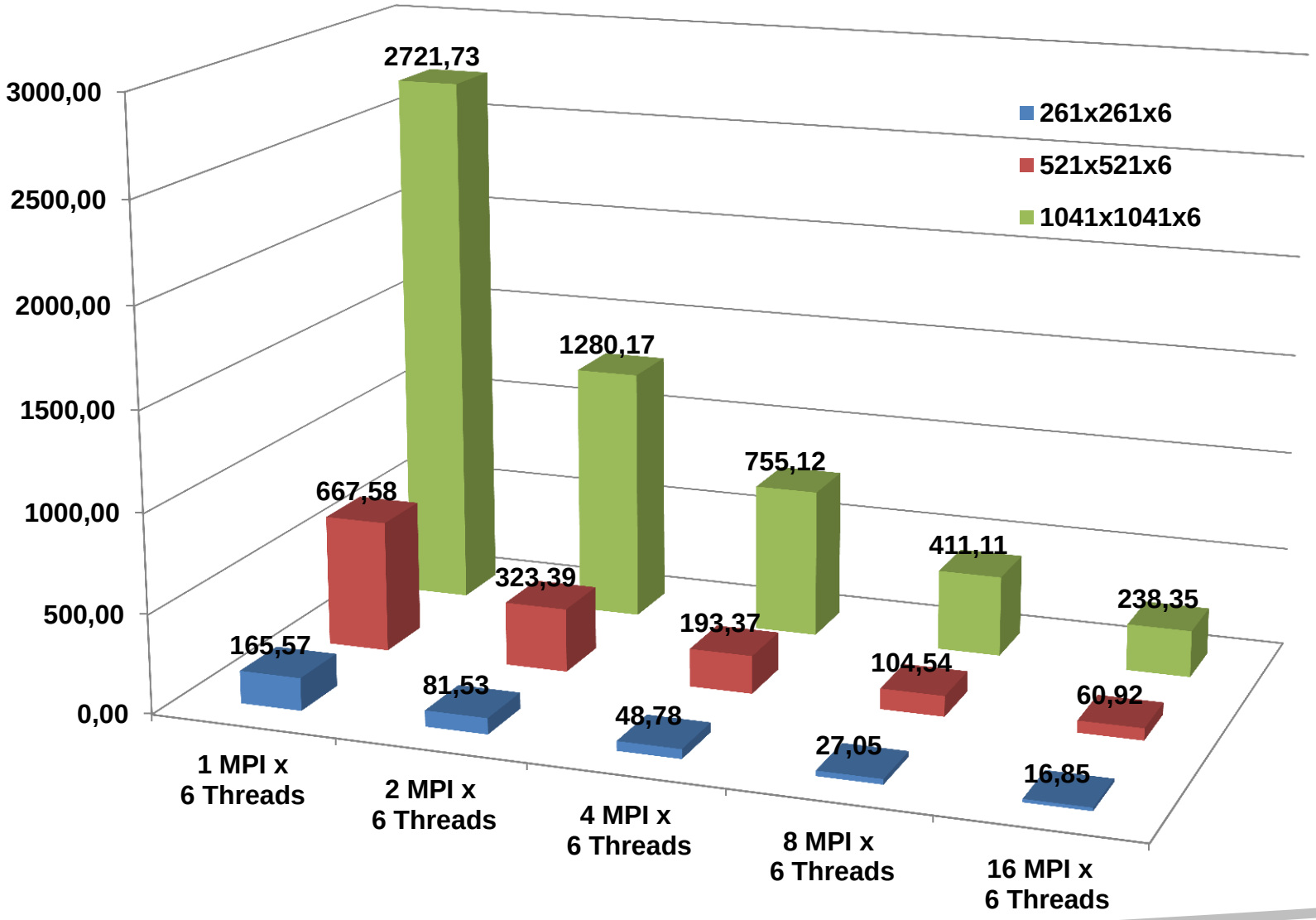
Накладные расходы на копирование данных между исходными массивами и их копиями



Времена выполнения 100 на суперкомьютере К-100. Сухой газ



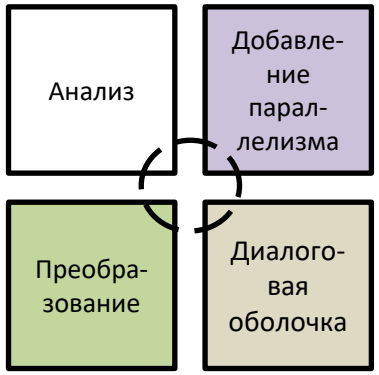
Времена выполнения 100 на суперкомьютере К-100. Жирный газ

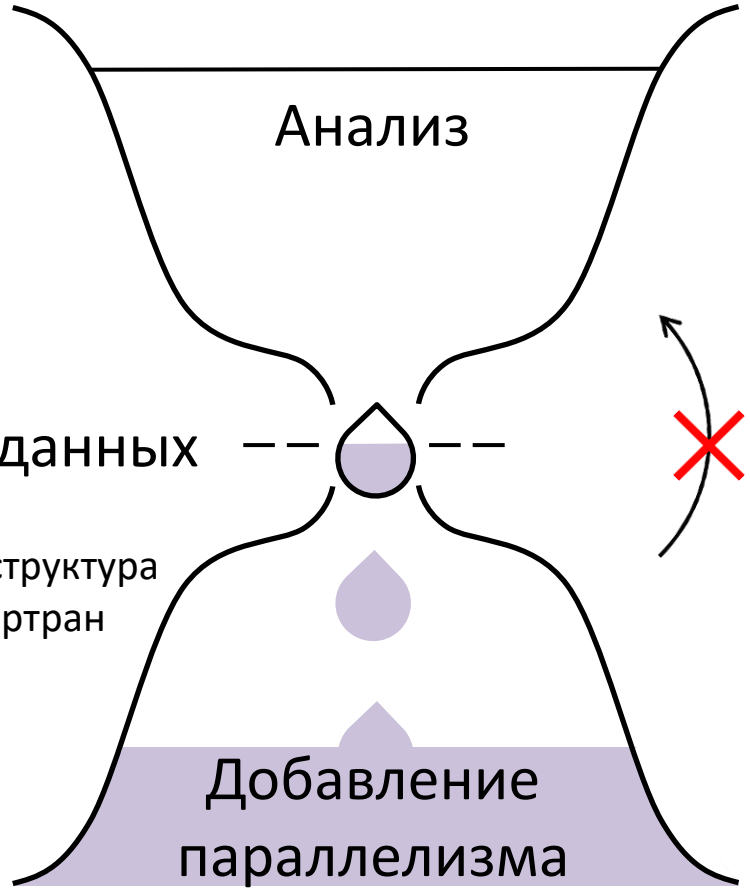


Архитектура системы САПФОР: настоящее

– *итерационное распараллеливание*

Набор независимых компонент

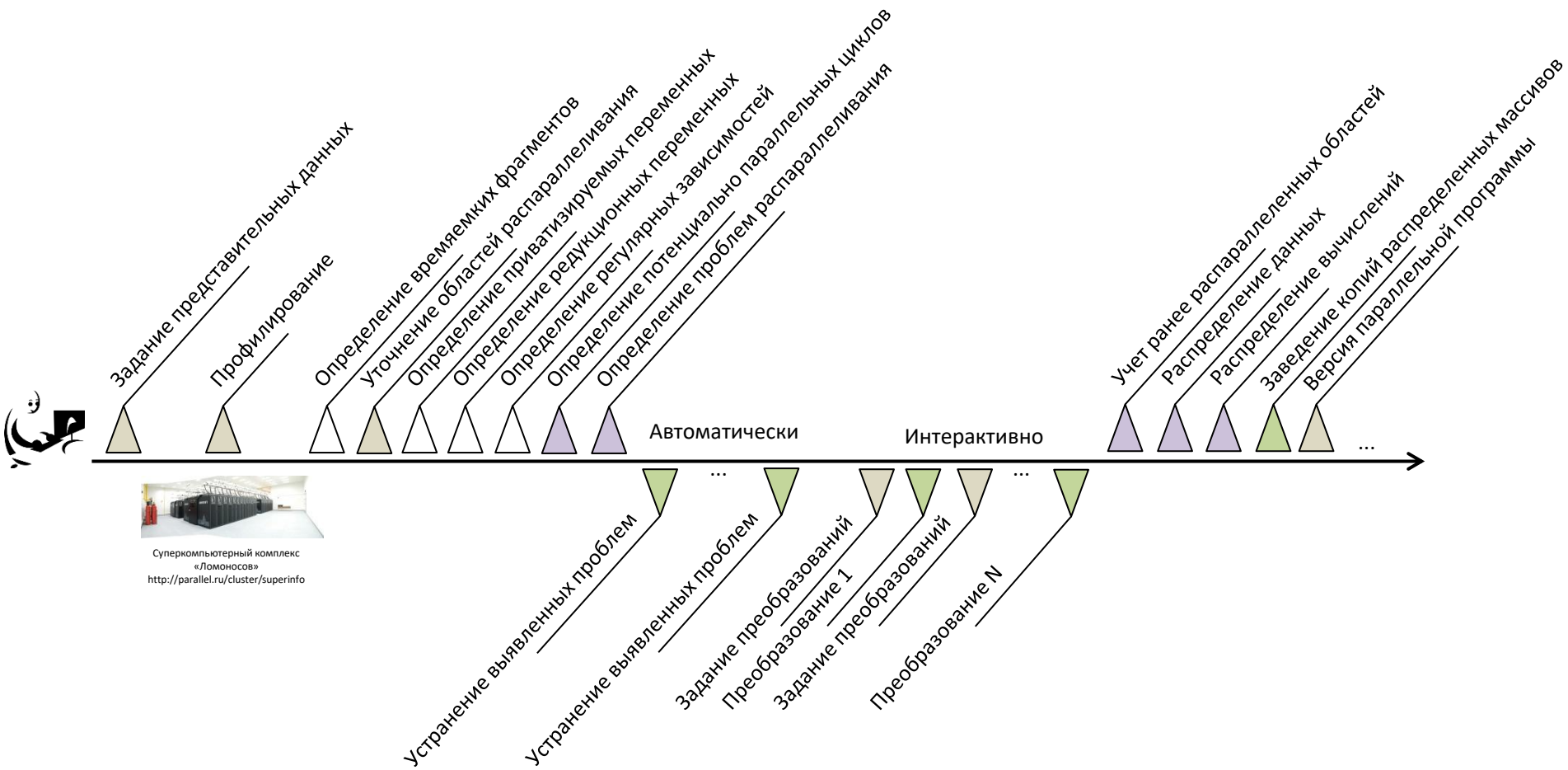




- Контекстно-независимая структура
- Ориентированность на Фортран
- Огрубление информации

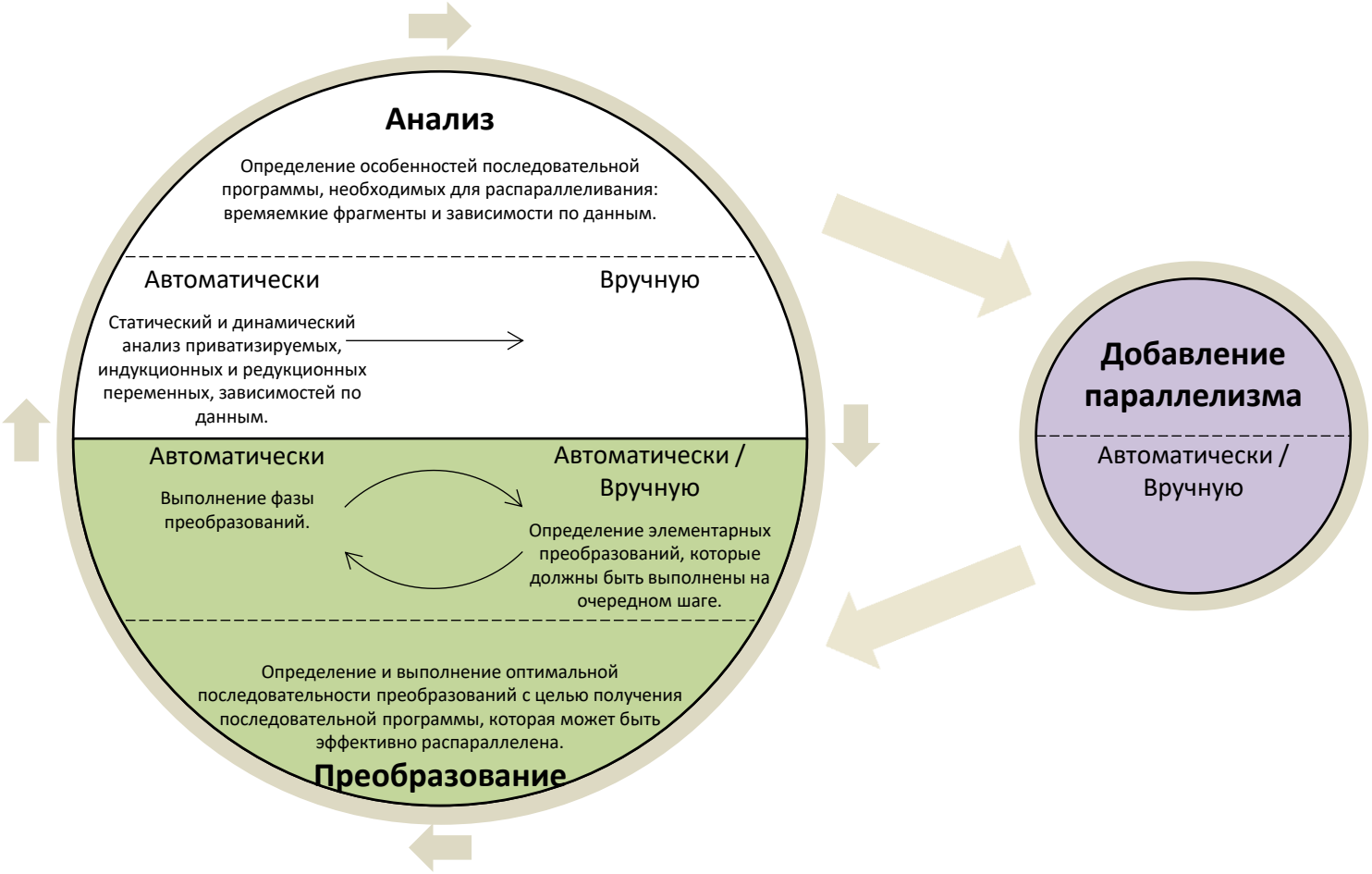
Архитектура системы САПФОР: будущее

– *итерационное распараллеливание*



Процесс распараллеливания программы – <u>поиск оптимизационной последовательности фаз</u>, обеспечивающей переход от последовательной версии программы к эффективной параллельной версии.

Автоматизация распараллеливания: задачи



Автоматизация распараллеливания: задачи

Средства профилирования и выделения областей распараллеливания.

Средства поиска последовательностей преобразований для автоматического устранения проблем распараллеливания.

Средства автоматического выполнения наиболее часто встречающихся преобразований.

Средства интерактивного взаимодействия с пользователем в процессе распараллеливания.

Спасибо за внимание

НАУЧНЫЙ СЕРВИС В СЕТИ ИНТЕРНЕТ 2016

KIAM RAS DVM SYSTEM

http://dvm-system.org

ОРДЕНА ЛЕНИНА им.М.В.КЕЛДЫША РАН