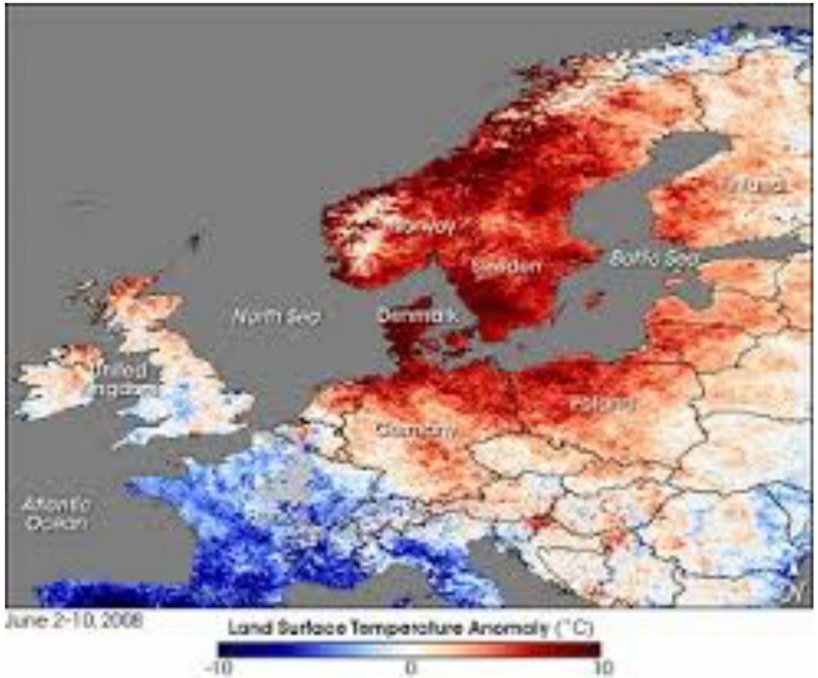
Accelerating Heatstroke Risk Simulation on Modern Vector Computers

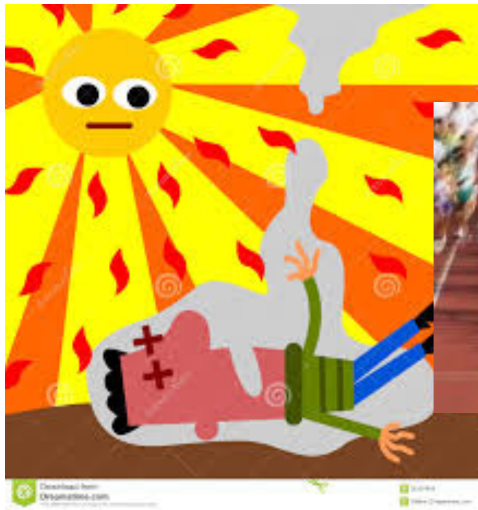
Ryusuke Egawa (Tohoku University),
Yoko Isobe (NEC / Tohoku Univ),
Toshiyuki Kato (NEC),
Akimasa Hirata (Nagoya Institute of Techologies)

Background

- The number of heat waves with fatalities has increased in Europe, North America, and Asia.



- **Heatstroke**
  - The number of people hospitalized suffering heatstroke is increasing
    - 58,000 patients in 2014
- **Needs for alerting the risk of heatstroke precisely**
  - Olympic game will be held at Tokyo in the most hottest season :-)
  - 10 million foreigners will joint to the event



Motivation

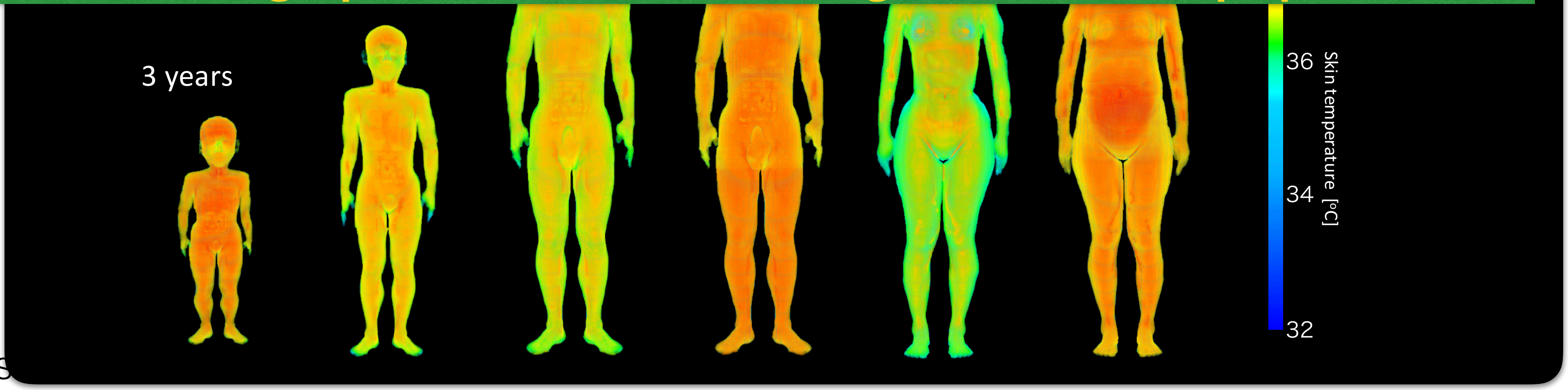
TOHOKU UNIVERSITY

Cyberscience Center

- The environment changes with time, and different individuals are vulnerable to heat-related illness to different degree
    - The changes of body temperature strongly depend on individual (differences body size, age, gender tend to perspire a lot or not, difference in genders, etc.)
- An appropriate health risk assessment covering 90% of the population would facilitate and effective response to increased rates of heat illness for major summer sports events and the elderly in daily life

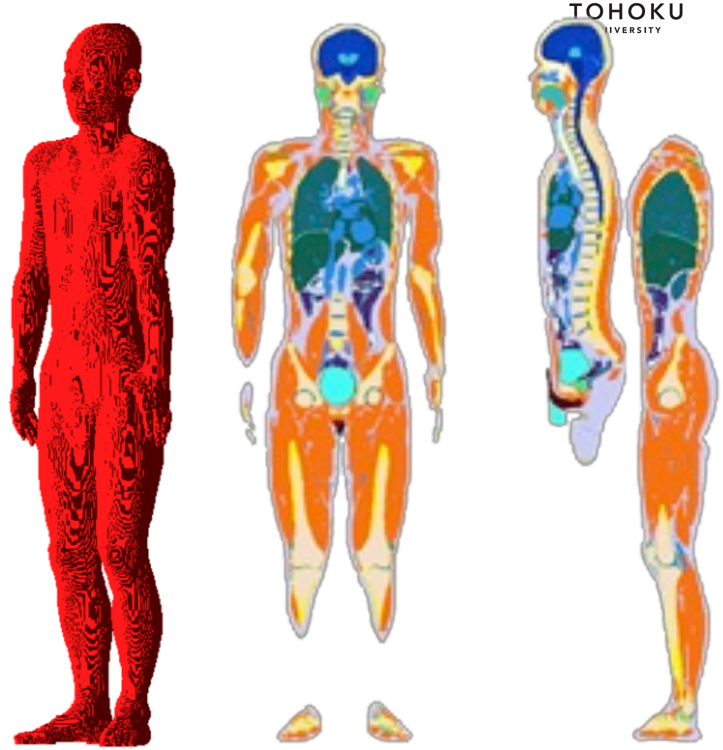
The risk of heatstroke should be examined more precisely considering individual differences

Simulating specific cases covering most of the population



Human Body Models

- Japanese adult male and female body models and a three-year-old child model
  - The adult model
    - 51 anatomic regions (provides by NICT)
      - skin, muscle, brain, lens, heart, and etc.



Height [m]	1.73
Weight [kg]	65
Surface area [m^2]	1.75
S/W [m^2/kg]	0.027

How can we estimate the body temperature

age

height

weight

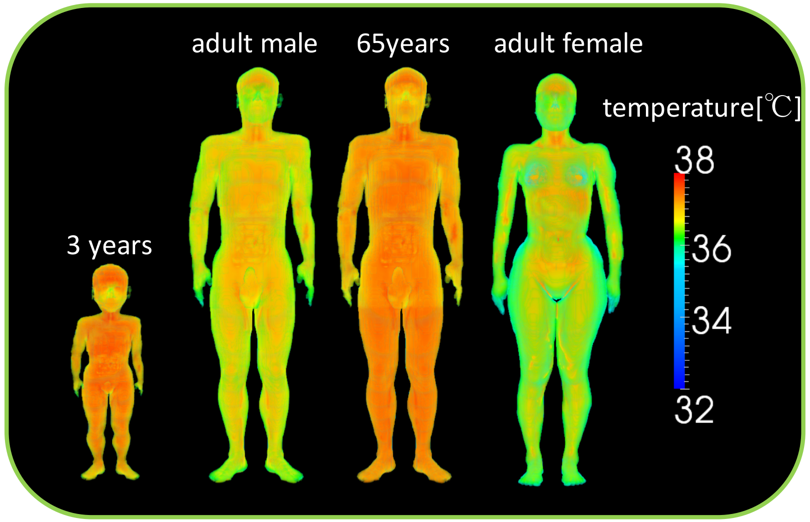
perspiration

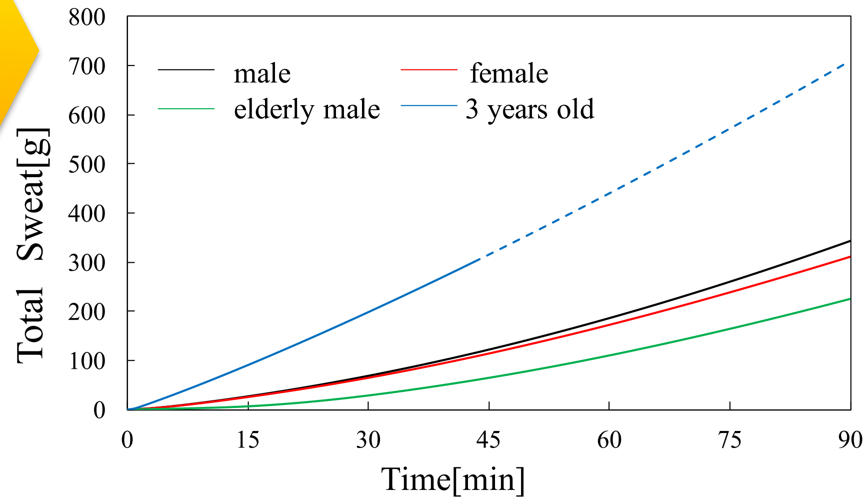
temperature

humidity

Gender

body temperature Simulator





Simulation conditions
90minutes under sunshine with 37.5℃ and 65% humidity

Flowchart of Bio-heat Modeling with Thermoregulatory Response in Computational Domain

TOHOKU UNIVERSITY

Cyberscience Center

Sci., (2015).

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 27, 2017, accepted January 3, 2018, date of publication January 11, 2018, date of current version February 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2791962

Risk Management of Heatstroke Based on Fast Computation of Temperature and Water Loss Using Weather Data for Exposure to Ambient Heat and Solar Radiation

KAZUYA KOJIMA[1], AKIMASA HIRATA[1], (Fellow, IEEE), KAZUMA HASEGAWA[1], SACHIKO KODERA[1], ILKKA LAAKSO[2], (Member, IEEE), DAISUKE SASAKI[3], TAKESHI YAMASHITA[3], RYUSUKE EGAWA[4], YUKA HORIE[5], NANAKO YAZAKI[5], SAERI KOWATA[5], KENJI TAGUCHI[6], AND TATSUYA KASHIWA[6], (Member, IEEE)

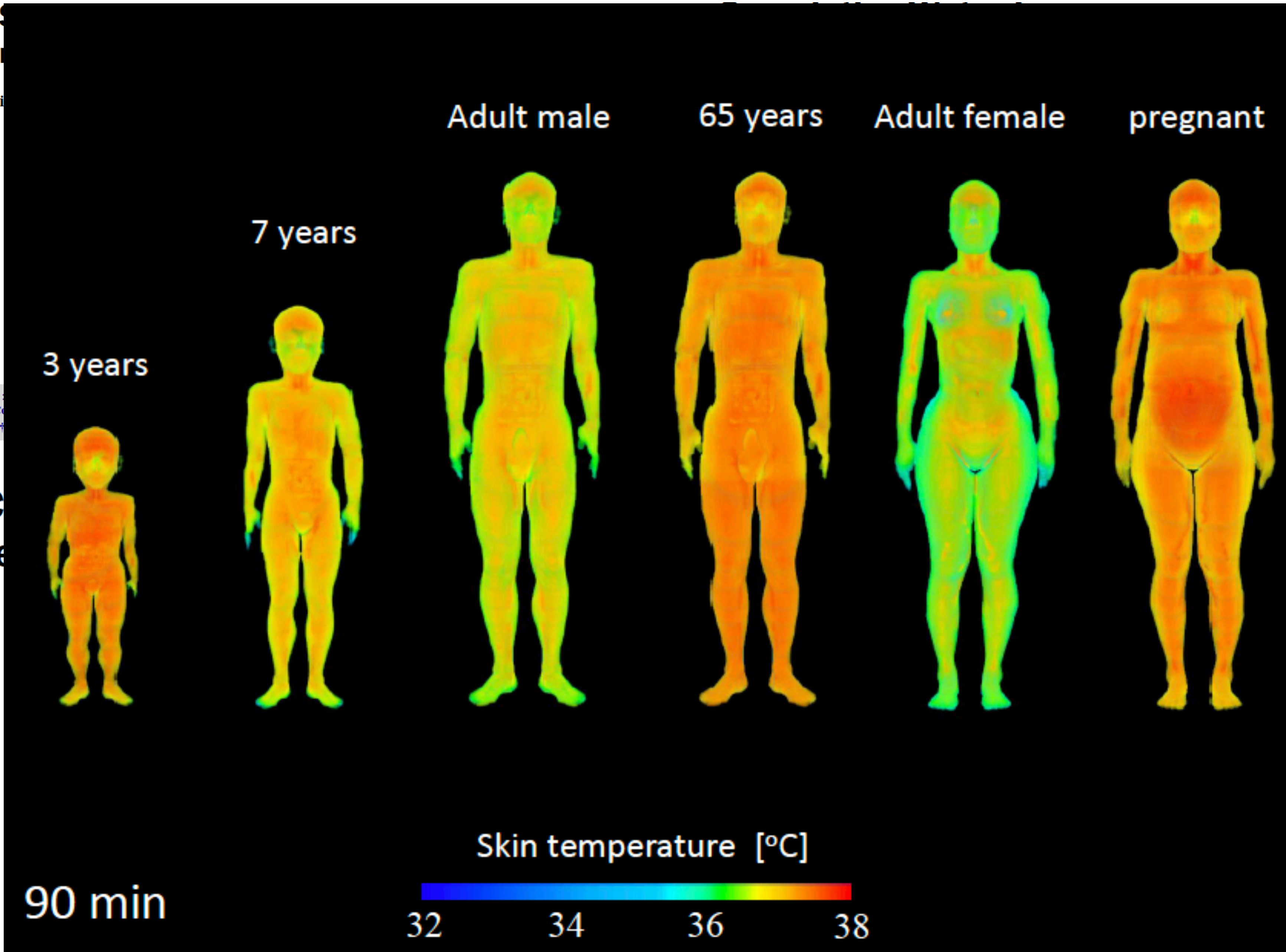
[1]Department of Electrical and Mechanical Engineering, Nagoya Institute of Technology, Nagoya 466-8555, Japan
[2]Department of Electrical Engineering and Automation, Aalto University, 02150 Espoo, Finland
[3]Information Technology Division, Information Department, Tohoku University, Sendai 980-8578, Japan
[4]Cyberscience Center, Tohoku University, Sendai 980-8578, Japan
[5]Japan Weather Association, Tokyo 170–6055, Japan
[6]Department of Electrical and Electronic Engineering, Kitami Institute of Technology, Kitami 090-8507, Japan

Corresponding author: Akimasa Hirata (ahirata@nitech.ac.jp)

This work was supported by the Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures and High Performance Computing Infrastructure in Japan under Project jh170010-NAH.

ABSTRACT Several indexes, such as the heat index, wet-bulb globe temperature, and the universal thermal climate index, are used to estimate the risk of seasonal heat illness. These indexes correspond to the heat load of an individual in identical environmental conditions for a prolonged period of time. In daily life, the environment changes with time, and different individuals are vulnerable to heat-related illness to different degrees. An appropriate health risk assessment covering 90% of the population would facilitate an effective response to increased rates of heat illness for major summer sport events and the elderly in daily life. In this paper, a fast computation for simulating temperature elevation and sweating is implemented using weather forecast data. In particular, a bioheat equation considering thermoregulatory responses is solved in the time domain using anatomical human body models including young adults, the elderly, and children. To accelerate simulation, the computational code is vectorized and parallelized, and subsequently implemented on an SX-ACE supercomputer. The computational results are validated in typical cases of young adults, children,

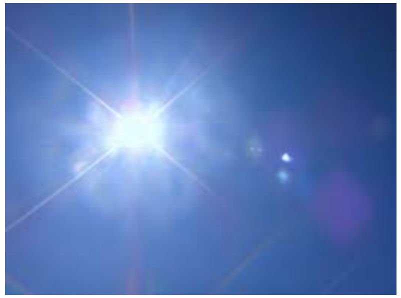
The solver provides…

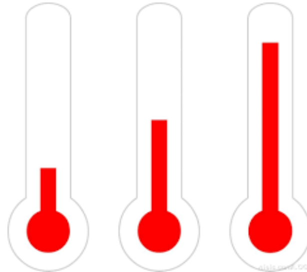


W

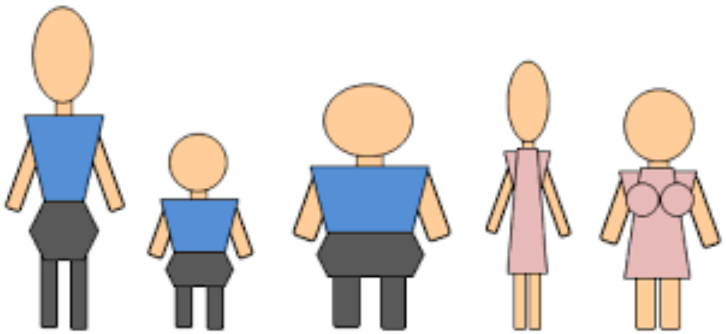
Original Code and Challenges

- 7 - 16 hours are spent for a simulation duration of two hours (for just 1 case) on Xeon based workstation





- Not parallelized
- Target: As fast as possible!
  - to create the Heat-Stroke Risk Data Bas
  - to achieve a short TAT and high QoS



First Touch

PROC.NAME	FREQ	EXCLUSIVE TIME[sec](%)	AVER.TIME [msec]	MOPS	MFLOPS	V.OP RATIO	AVER. V.LEN	VECTOR TIME	I-CACHE MISS	O-CACH MISS	BANK CONFLICT CPU PO	BANK CONFLICT NETWORK	ADB HIT ELEM.%
temprise_k	**300**	**3371.986(89.9)**	**11239.954**	**853.5**	**193.7**	**14.8**	**216.5**	**12.55**	**3.559**	**1128.78**	**1.14**	**1.834**	**6.26**
read_model_temp	**1**	**325.450(8.7)**	**325449.655**	**853**	**1.4**	**0.36**	**224.3**	**0.05**	**11.55**	**45.121**	**0.002**	**0.005**	**54.88**
cal_h	300	25.284(0.7)	84.279	44643.7	7250	99.33	222	25.282	0.001	0.002	1.477	1.897	4.93
tmp	11	14.385(0.4)	1307.769	1332.2	154.6	41.36	219.9	0.383	0	0.886	0	0.032	5.16
sw_calc	300	13.450(0.4)	44.834	54844.6	14626	99.26	223.6	13.439	0.002	0.004	0	1.064	5.11
det_cell	1	0.885(0.0)	884.73	10651.5	180.1	94.87	217.6	0.312	0	0.086	0.014	0.093	86.36
main	1	0.325(0.0)	325.358	26024.2	3271.3	98.99	229.1	0.3	0.004	0.006	0.052	0.03	46.57
make_mask	1	0.184(0.0)	183.822	21253.7	0	99.1	218.6	0.173	0	0	0.001	0.033	59.05
total	915	3751.949(100.0)	4100.491	1349.4	276.4	45.62	221.5	52.491	15.117	1174.888	2.686	4.988	6.11

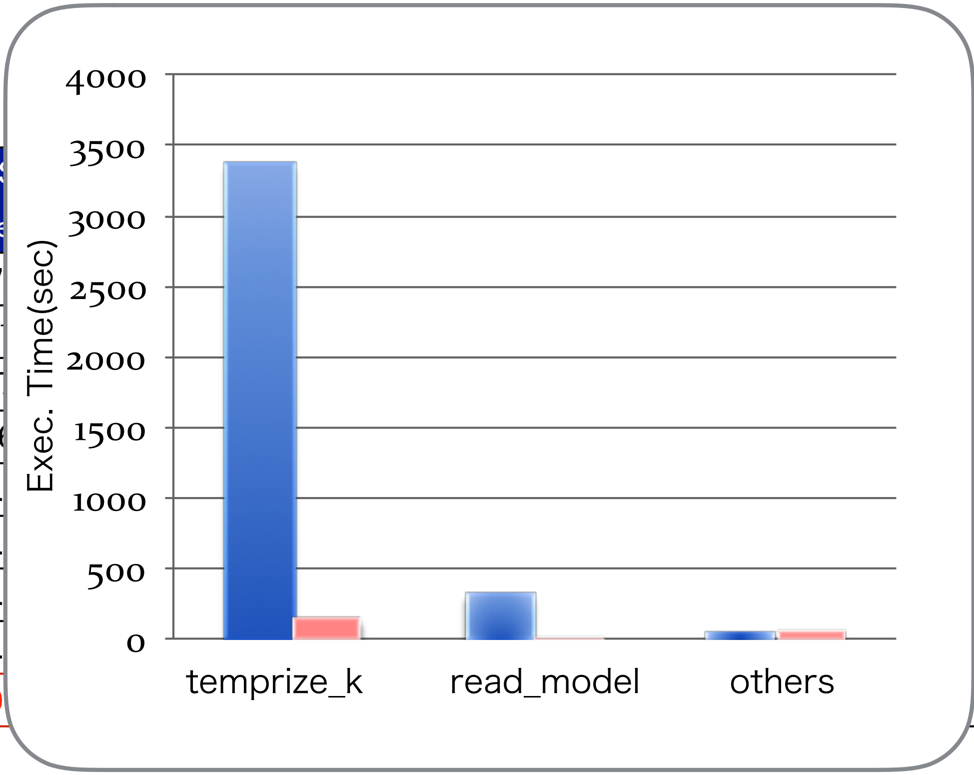
- just running "as is" code on a single node of SX-ACE
  - 300steps, mesh size 2mm, 866×320×160
    - the final target is 5400Steps for single case.
  - almost same performance with Intel based system
- a few kernel dominate execution time
  - with low vectorization ratio, I/O Overhead
  - vectorize, and then parallelize

Performance optimization

- Following optimizations are applied
  - Vectorization by Loop division, Loop expansion
  - Improve load-store performance
  - Re-organize a block with many "If statements"
  - Reducing the number of communications
  - change "scalar variables" to array, Function inlining, remove redundant assignments/calculations

Node Performance

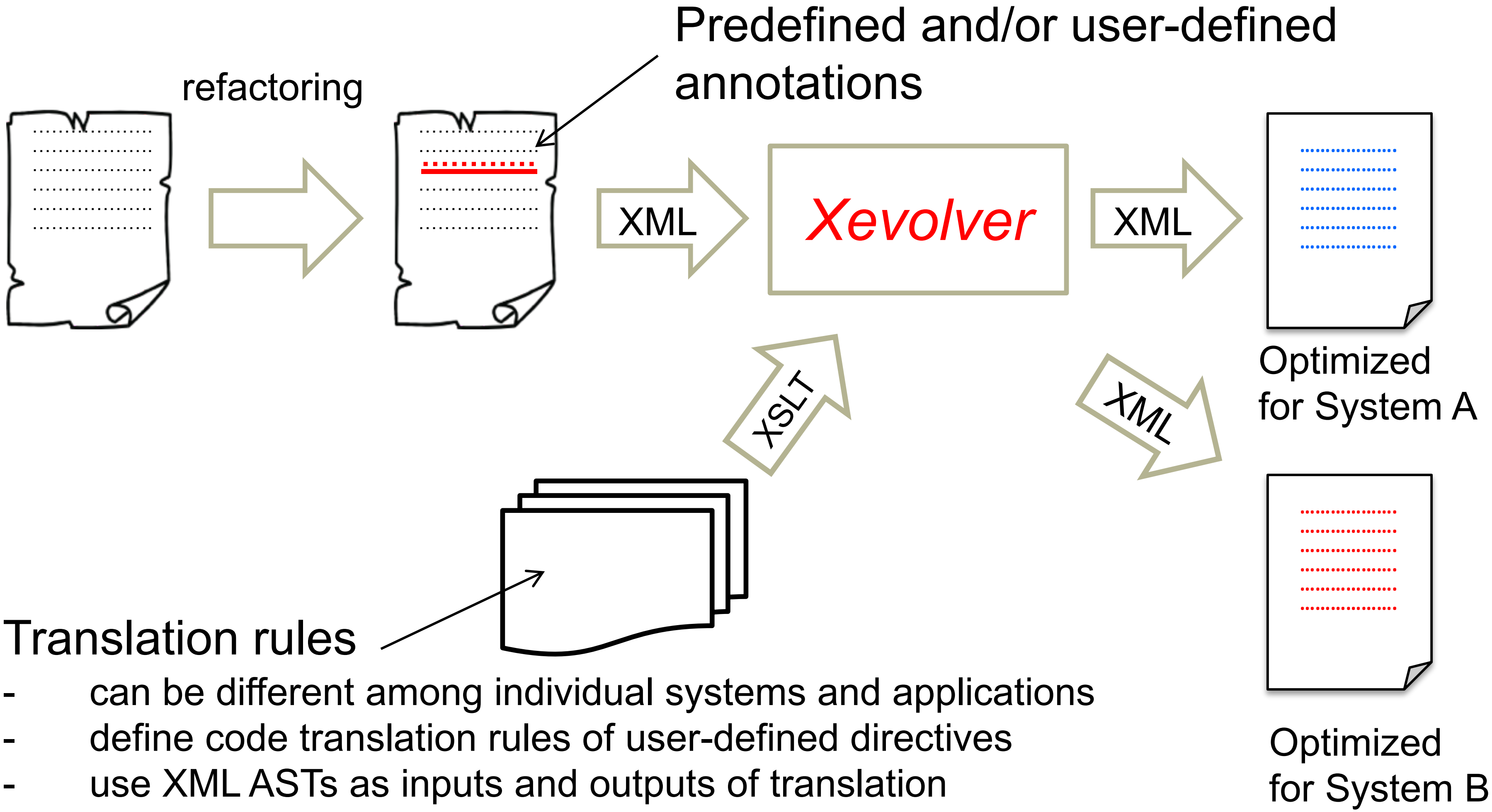
PROC.NAMI	FREQUEN	EXCLUS	HE	O-CACHE	BANK CONFLICT	
		TIME[se		MISS	CPU PORT	NETWOR
temprise_k	300	147.017	007	0.017	2.327	11.223
cal_h	300	24.660(	0	0.001	0.437	1.897
tmp	11	14.389(	0	0.884	0	0.032
sw_calc	300	13.450(	002	0.003	0	1.064
det_cell	1	0.895(0.	0	0.087	0.015	0.093
main	1	0.327(0.	005	0.007	0.051	0.031
make_mask	1	0.185(0.	0	0	0.001	0.034
read_model_	1	0.138(0.	0	0	0.006	0.019
total	915	201.059	014	0.999	2.838	14.394



3751sec → 201sec (300steps)

16.5 x speedup

XEVOLVER



Translation rules

- can be different among individual systems and applications
- define code translation rules of user-defined directives
- use XML ASTs as inputs and outputs of translation
- XevTgen can easily generate the rules by using before and after version of codes as inputs

The contents of HPC Refactoring Catalog

Cyberscience Center

List 3 code motion (2)

最終更新：yokoyama 2 月 25の午後 7:48

【Purpose】

Reducing redundant calculation

【Procedure】

Since the value of a34 is just depending on i12, j13, on o31, the equation a34 = 1.0e0(j13, i12) can be moved to outside of the loop. By this movement the kernel becomes double nested loop and the total amount of calc. is reduced.

【Before】

```
X------>     DO j13 = 1, n23
|          !cdir expand=10
|+----->        DO i12 = 1, l24
||                 o31(j13,i12) = o32 * t26(i12)
|+-----         END DO
X------      END DO
+------->    DO i14 = 1, n25
|W----->        DO j13 = 1, n23
||         !cdir expand=10
||*---->           DO i12 = 1, l24
|||     A             a34 = 1.0e0 + (o31(j13,i12)) ** 2
|||                   a27 = a27 + ((1.0e0 + (t1 + 1.0e0) * &
|||                            (o31(j13,i12)) ** 2) / a34)
|||                   a28 = a28 + ((o31(j13,i12) * t1) / a34)
||*----            END DO
|W-----         END DO
+------      END DO
```

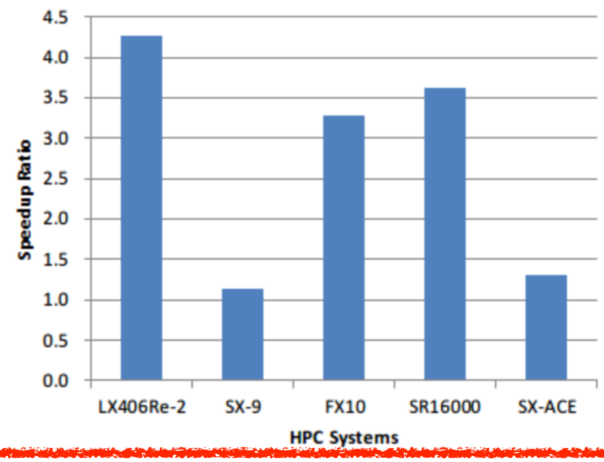
【After】

```
X------>     DO j13 = 1, n23
|          !cdir expand=10
|+----->        DO i12 = 1, l24
||                 o31(j13,i12) = o32 * t26(i12)
||                 a35(j13,i12) = 1.0 / (1.0e0 + o31(j13,i12) ** 2)
|+-----         END DO
X------      END DO
+------->    DO i14 = 1, n25
|W----->        DO j13 = 1, n23
||         !cdir expand=10
||*---->           DO i12 = 1, l24
|||     A             a27 = a27 + ((1.0e0 + (t1 + 1.0e0) * &
|||                        (o31(j13,i12)) ** 2) * a35(j13,i12))
```

【Source code】

03.zip

【Performance Portability Analysis】



High performance portability

【Xevolver】

translation recipe for Tgen

xevtgen_03.zip

Current Status

TOHOKU UNIVERSITY Cyberscience Center

◇ 57 lists are available at https://one.sc.cc.tohoku.ac.jp/hpcref/

	Items	#
1	Expanding vector length	7
2	Improving vector operation ratio	1
3	Avoiding conditional/redundant executions	6
4	Code modification for adopting vector instructions	7
5	Loop level vectorizations	14
6	Intra-node parallelization	1
7	Using system-specific instructions	1
8	Memory access optimization	10
9	Library utilization	6
10	Overlapping mem. And arith. operations	1
11	Reducing memory footprint	3

Catalog No.43

Replacing a variable with an array for promoting vectorization

【Before】

```
integer :: x
x = 0
 do i=1,MAX
   if(a(i) < 5000) then
     x = 100
   end if
   b(i) = a(i) + c(i)*x
 end do
```

【After】

```
integer :: x(MAX)
 x = 0
  do i=1,MAX
    if(a(i) < 5000) then
      x(i) = 100
    end if
    b(i) = a(i) + c(i)*x(i)
 end do
```

Since the scalar valuable x changes in every iteration, the compiler cannot vectorize this code.
To vectorize this loop, the scalar valuable x is replaced to the array x.

A CASE STUDY :PRACTICAL USAGE OF THE HPC REFACTORING CATALOG FOR CODE OPTIMIZATION

before

```
DO K=1,MODELZ

  DO J=1,MODELY
    DO I=1,MODELX
 • • •
          IF(UOLD(I,J,K).GE.US(I,J,K))THEN
            BCOEF=2**((UOLD(I,J,K)-US(I,J,K))/6)
            TEMP_DIFF=UOLD(I,J,K)-US(I,J,K)

          ELSE
            TEMP_DIFF=0E0
          ENDIF
 • • •
          IF(UOLD(I,J,K).LT.39E0)THEN
            BCOEF=1E0

          ELSE IF(UOLD(I,J,K).LT.44E0)THEN
            BCOEF=1E0+0.8E0*(UOLD(I,J,K)-39E0)

          ELSE
            BCOEF=1E0+(5E0*0.8E0)
          ENDIF

        ENDIF

      ELSE
        BCOEF=1E0
        TEMP_DIFF=0E0
      ENDIF
 • • •
      U(I,J,K)=UOLD(I,J,K)+(DT*SAR(I,J,K)/CP(LK(I,J,K))&
        &+DT*A(LK(I,J,K))*storetemp2(i,j,k)/(ROU(LK(I,J,K))*CP(LK(I,J,K)))&
        &-(DT*BCOEF*B(LK(I,J,K))*(UOLD(I,J,K)-henkaTB))/(ROU(LK(I,J,K)) • •
 • • •
```

After

```
DO K=1,MODELZ

  DO J=1,MODELY
    DO I=1,MODELX
 • • •
          IF(UOLD(I,J,K).GE.US(I,J,K))THEN
            BCOEF_(I,J,K)=2**((UOLD(I,J,K)-US(I,J,K))/6)
            TEMP_DIFF=UOLD(I,J,K)-US(I,J,K)

          ELSE
            TEMP_DIFF=0E0
          ENDIF
 • • •
          IF(UOLD(I,J,K).LT.39E0)THEN
            BCOEF_(I,J,K)=1E0

          ELSE IF(UOLD(I,J,K).LT.44E0)THEN
            BCOEF_(I,J,K)=1E0+0.8E0*(UOLD(I,J,K)-39E0)

          ELSE
            BCOEF_(I,J,K)=1E0+(5E0*0.8E0)
          ENDIF

        ENDIF

      ELSE
        BCOEF_(I,J,K)=1E0
        TEMP_DIFF=0E0
      ENDIF
 • • •
      U(I,J,K)=UOLD(I,J,K)+(DT*SAR(I,J,K)/CP(LK(I,J,K))&
        &+DT*A(LK(I,J,K))*storetemp2(i,j,k)/(ROU(LK(I,J,K))*CP(LK(I,J,K)))&
        &-(DT*BCOEF_(I,J,K)*B(LK(I,J,K))* • •
```

A CASE STUDY :PRACTICAL USAGE OF THE HPC REFACTORING CATALOG FOR CODE OPTIMIZATION

TOHOKU UNIVERSITY

Cyberscience Center

```
do i=1,MAX
   if(a(i) < 5000) then
      x = 100
   end if
   b(i) = a(i) + c(i)*x
end do
```

Translation rule xevtgen44.f90

```
do i=1,MAX
   if(a(i) < 5000) then
      x(i) = 100
   end if
   b(i) = a(i) + c(i)*x(i)
end do
```

Example in the Catalog

Number of Scalar Variables : 1
Single loop

Target Application

Number of Scalar Variables : 3
Triple nested loop

Modify the translation rule

```
!$xev tgen src message("rule1") begin
!$xev decl_statement1(z1,MAXa)
!$xev tgen src end
!------------------
!$xev tgen dst begin
 integer::z1(MAXa)
!$xev tgen dst end
```

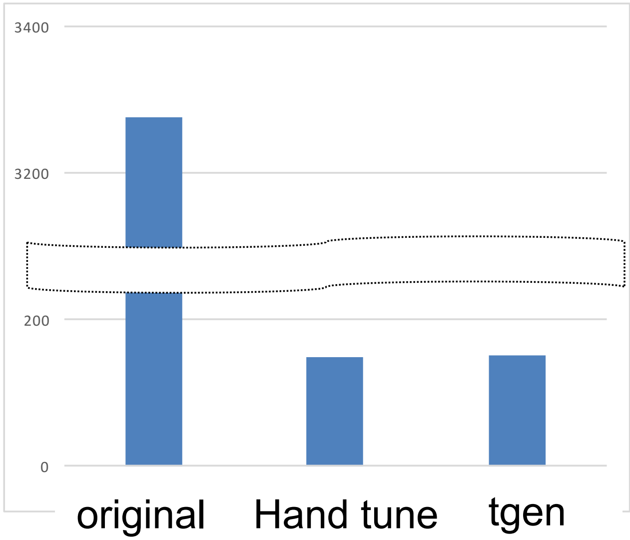
- Define 3D array
- Modify for 3 variables

```
!$xev tgen ctxdef(loop1) stmt begin
do ii = ib, ie, is
!$xev tgen stmt(body)
end do
!$xev tgen ctxdef end
```

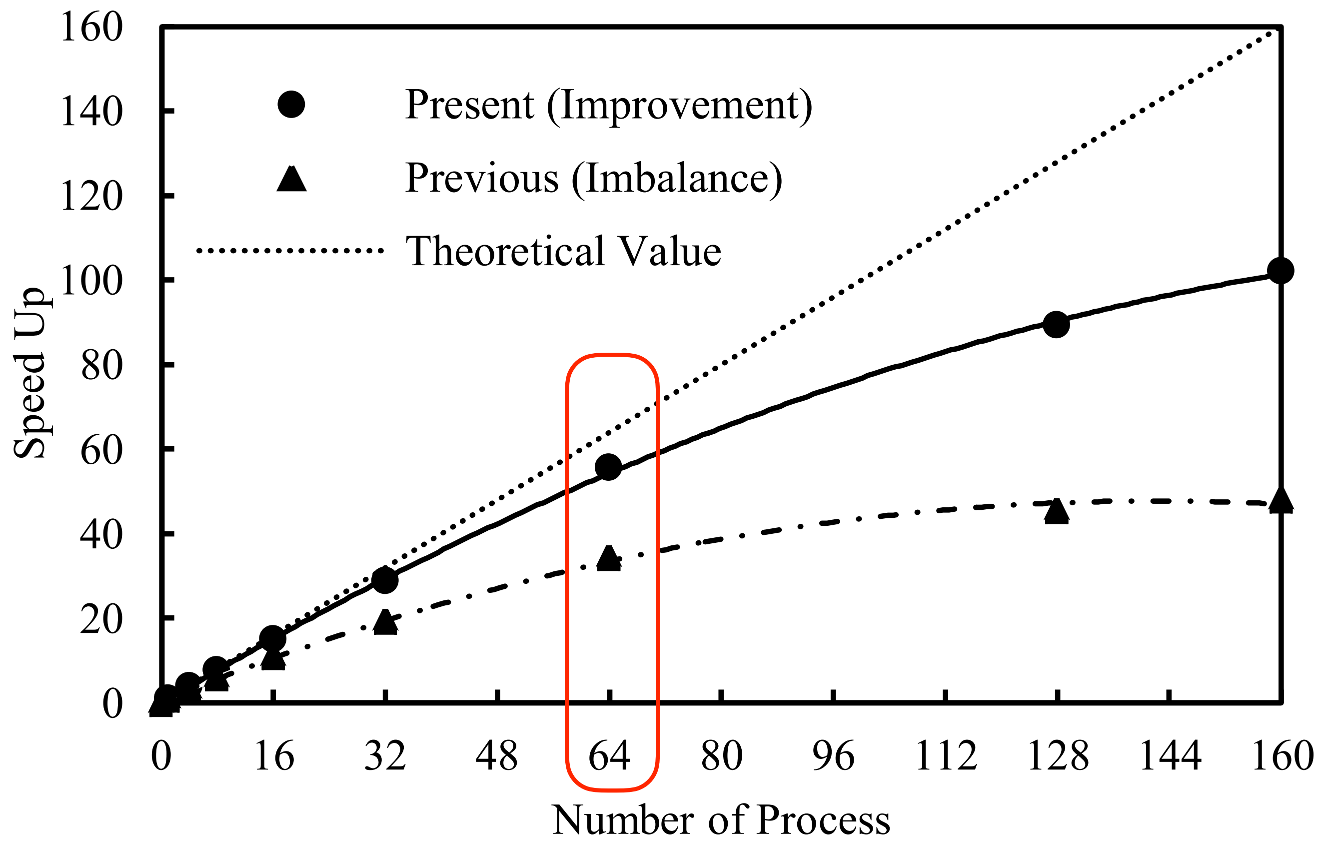
- Change to triple nested loop

```
!$xev tgen trans exp src(`x`) dst(`xev3_x(i,j,k)`)
                              context(loop3)
!$xev tgen trans exp src(`x`) dst(`xev3_x`)
                              context(not(loop3))
```

- Modify the rule for 3 variables



Parallel Performance

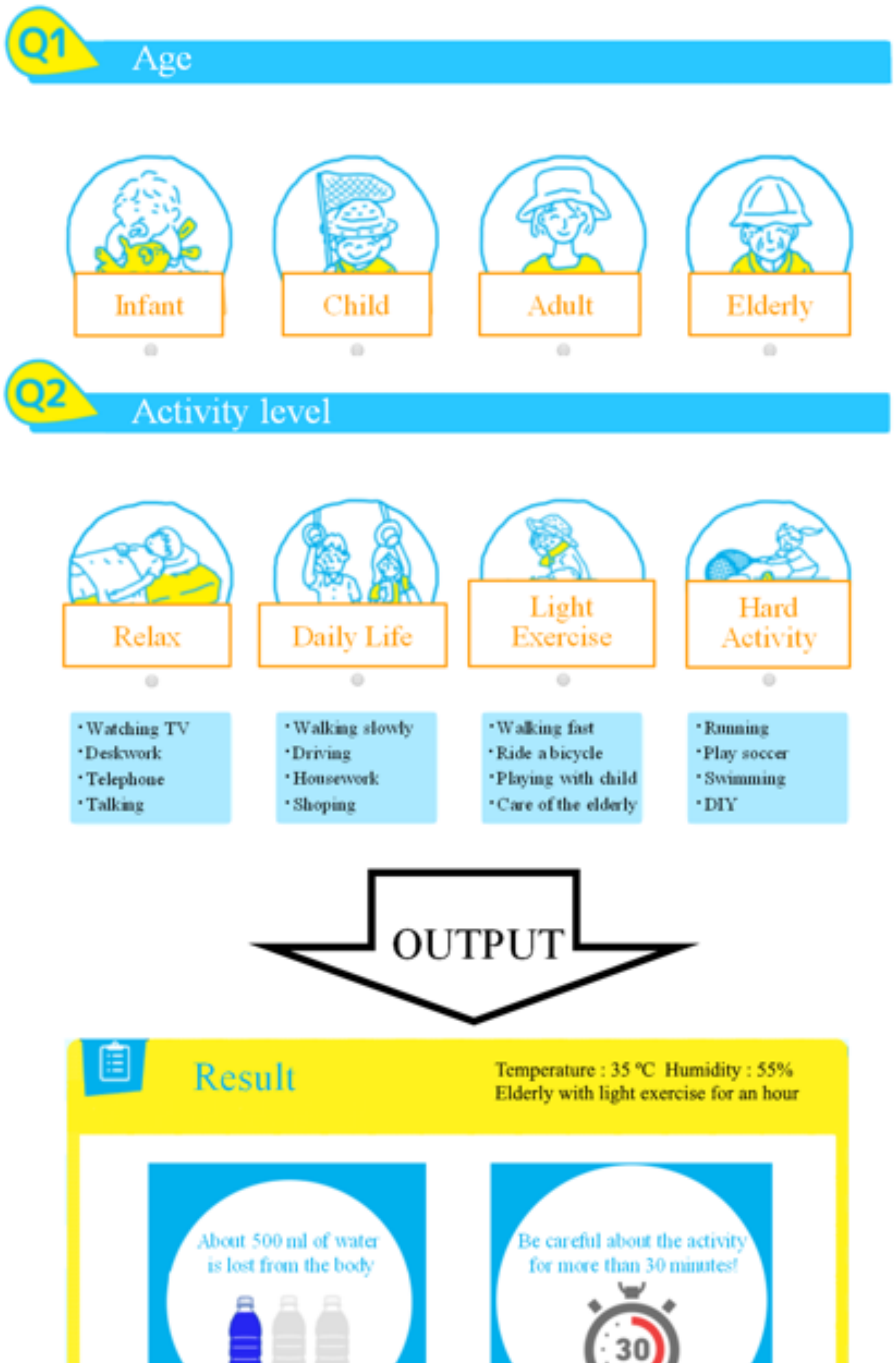


55x speedup with 64processes

77 sec in the adult model and 17 sec in the child model

Web based App (Japan Weather Association :JWA)

- Database is developed for core temperature elevation and sweating
  - 104 combinations of ambient temperature (from 28 to 40℃　per 1℃)
  - relative humidity (from 10 to 80 % per 10 %) were chosen
  - 7296 simulations were conducted and used for generating the database.
  - 170k page views from April 2017 to September 2017.



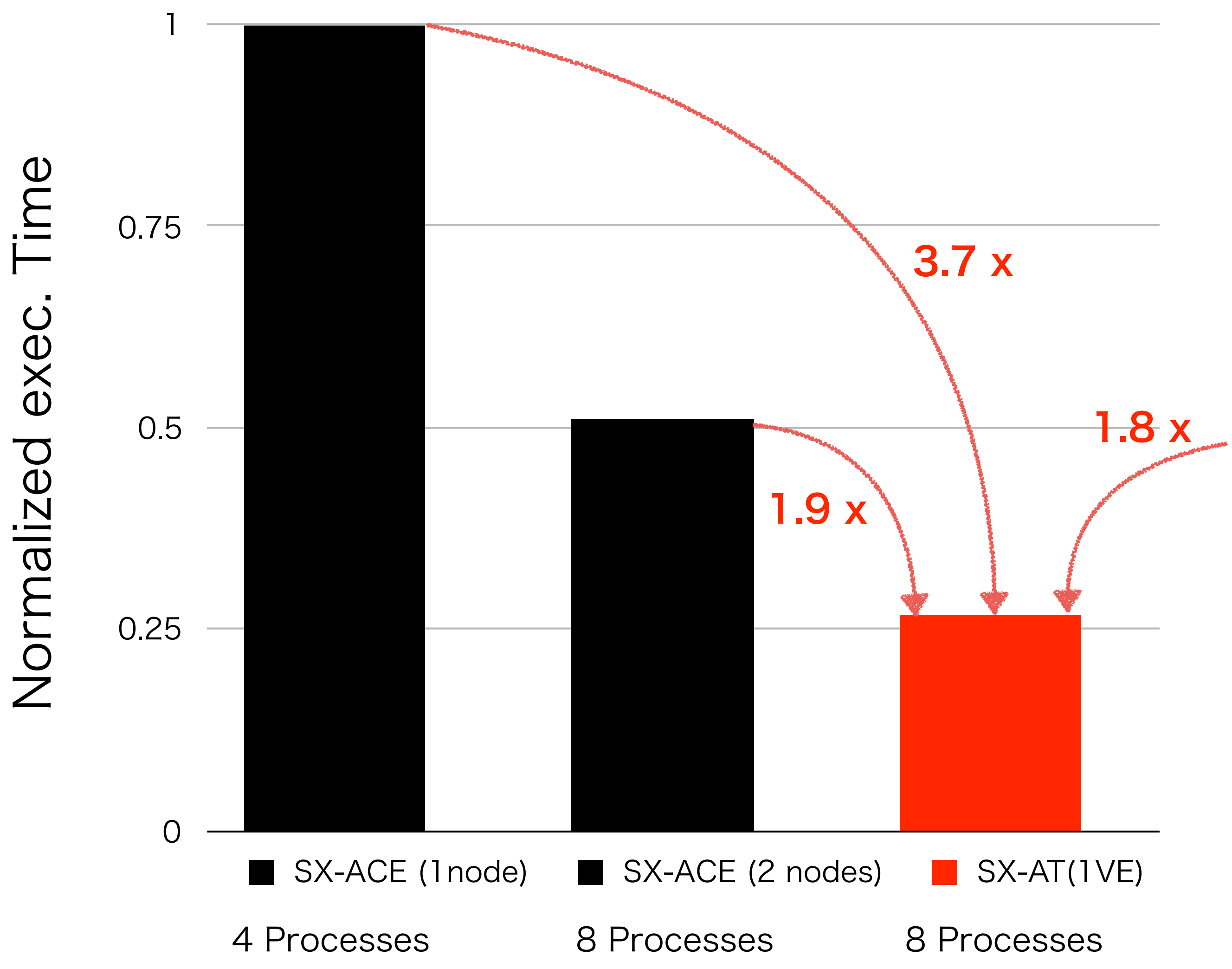
Evaluation Setups

	SX-ACE 1 CPU	Aurora 1 VE	Ratio
Core Peak Performance [GFLOPS]　（double）	64.0	268.8	4.2
Number of cores	4	8	2.0
Node Peak Performance [GFLOPS]	256.0	2,150.4	8.4
Memory Bandwidth per node or VE [GByte/s]	256.0	1,228.8	4.8
Memory Capacitance of node or VE [GByte]	64	48	0.75
B/F	1.0	0.57	**0.57**

Simulation Code

- Updated version of Heatstroke Risk Simulator
  - Considering the duration of heat radiation with finer time steps
- Actual B/F 1.57

Execution Time (vs. SX-ACE)



Conclusions

- Precise risk simulation of body temperature changes
  - Heatstroke
  - Concepts of Heat-stroke-Risk Alert System
- Porting the code to vector systems
  - Web based Apps
- Toward Real Time Risk Assessments
  - Nice performance using a single VE
  - Using more numbers of VEs remains as our future work.

More detailed information of the simulation can be found in IEEE Access,
K. Kojima, et. al., “Risk Management of Heatstroke Based on Fast Computation of Temperature and Water Loss using Weather Data for Exposure to Ambient Heat and Solar Radiation,”. [IEEE Access,(2018),1-10]