
40 YEARS
OF CHANGING
THE WORLD
Intel Developer
FORUM
Invent the new reality.

DIGITAL TRANSFORMATION

Andrew A. Chien

Vice President
Intel Research


PCWorld
Alien
Nuclear Smuggling
Laser Pulses
Flat Carbon
WIRED
Evil Genius

The Transformation in Progress ...

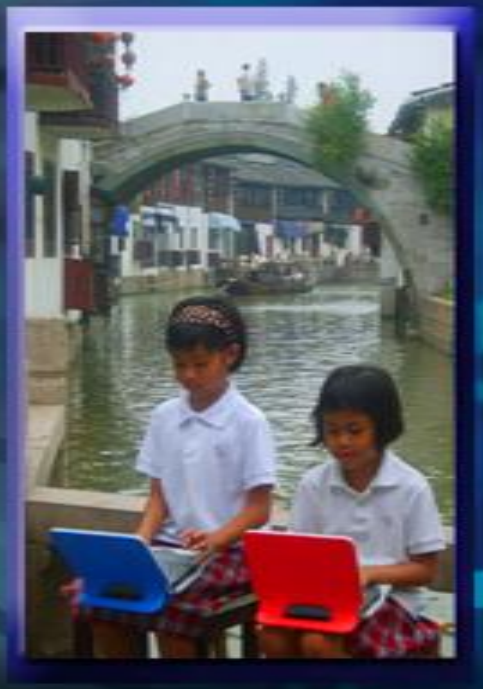
Improving Quality of Life • Driving Innovation • Expanding Opportunities

Transforming the Learning Environment

19th Century:
Industrialism

20th Century:
Information Age

21st Century:
Interaction Age



Second Generation Classmate PC

Complete "Netbook" Platform:

- Improved Usage Model
- Full Wireless Support
- Integrated Webcam
- Robust, Customizable

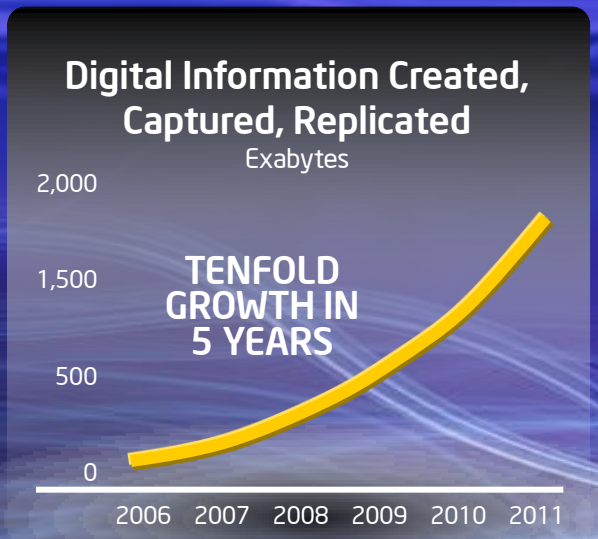
Improving Quality of Life

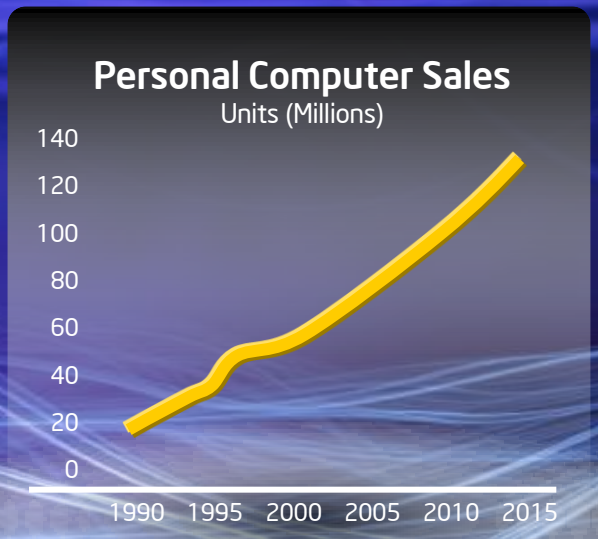
Driving Innovation

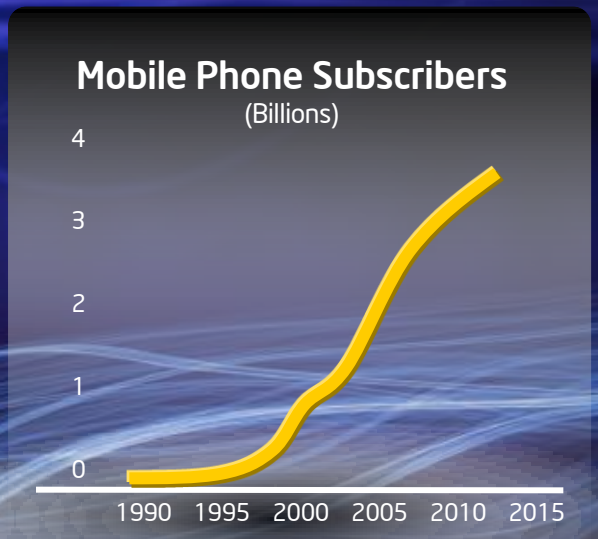
Expanding Opportunities

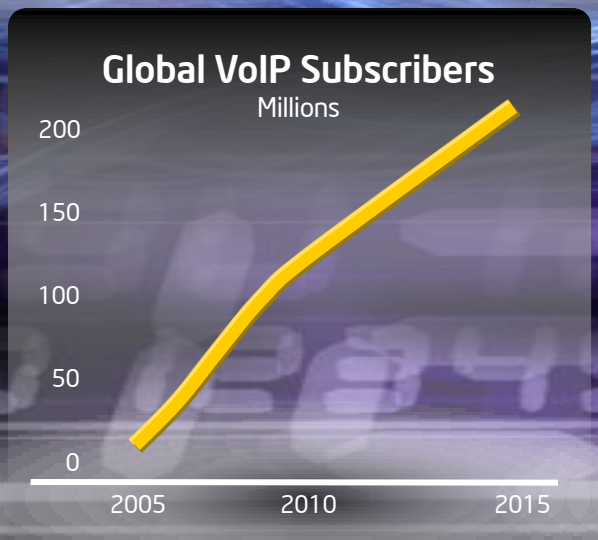
Accessibility
Content
Connectivity

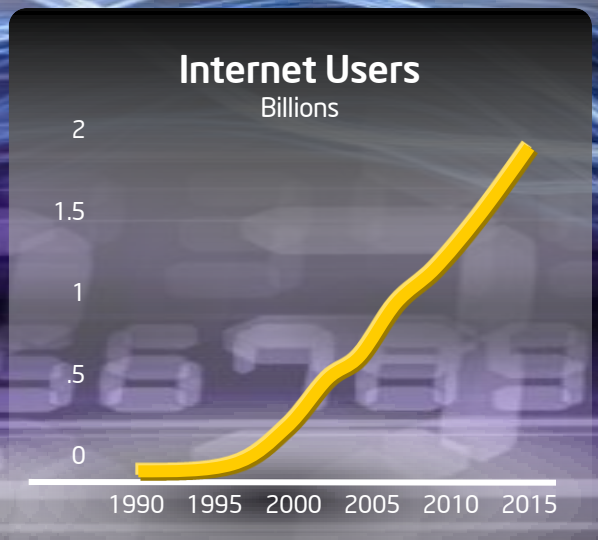
Digital Growth Worldwide











Source: IDC, The Digital Economy Fact Book


Prof. Ni Guangnan
Academician of Chinese Academy of Engineering
Researcher, Institute of Computing Technology,
Chinese Academy of Sciences
中国工程院
CHINESE
ACADEMY of
ENGINEERING
www.cae.cn

ICT Industry Development in China

中国工程院
CHINESE ACADEMY of ENGINEERING
www.cae.cn

Challenges

- Very low R&D investments
- Low profits due to low-end manufacturing
- Core technologies dependent on imports

中国工程院
CHINESE ACADEMY of ENGINEERING
www.cae.cn

Strategy for Chinese Companies

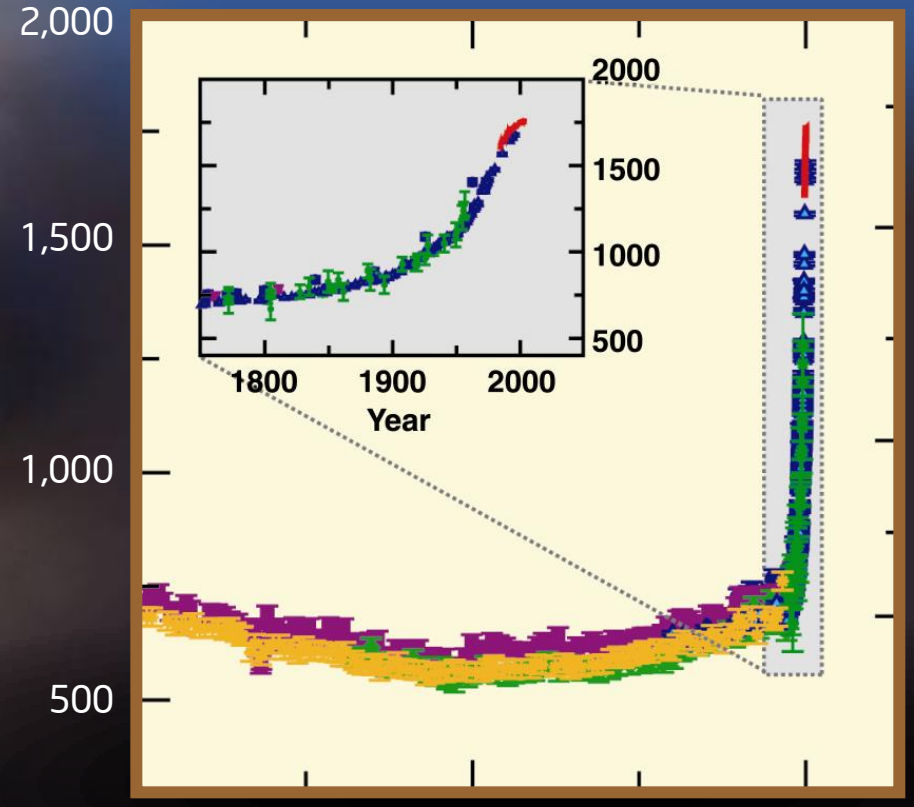
- Fully utilize the huge domestic market to master core technology and to develop own IP
- Push for open standards
- Promote open source software

中国工程院
CHINESE ACADEMY of ENGINEERING
www.cae.cn

Transforming the Environment

Greenhouse Effect

Changes in GHGs from Ice Core and Modern Data

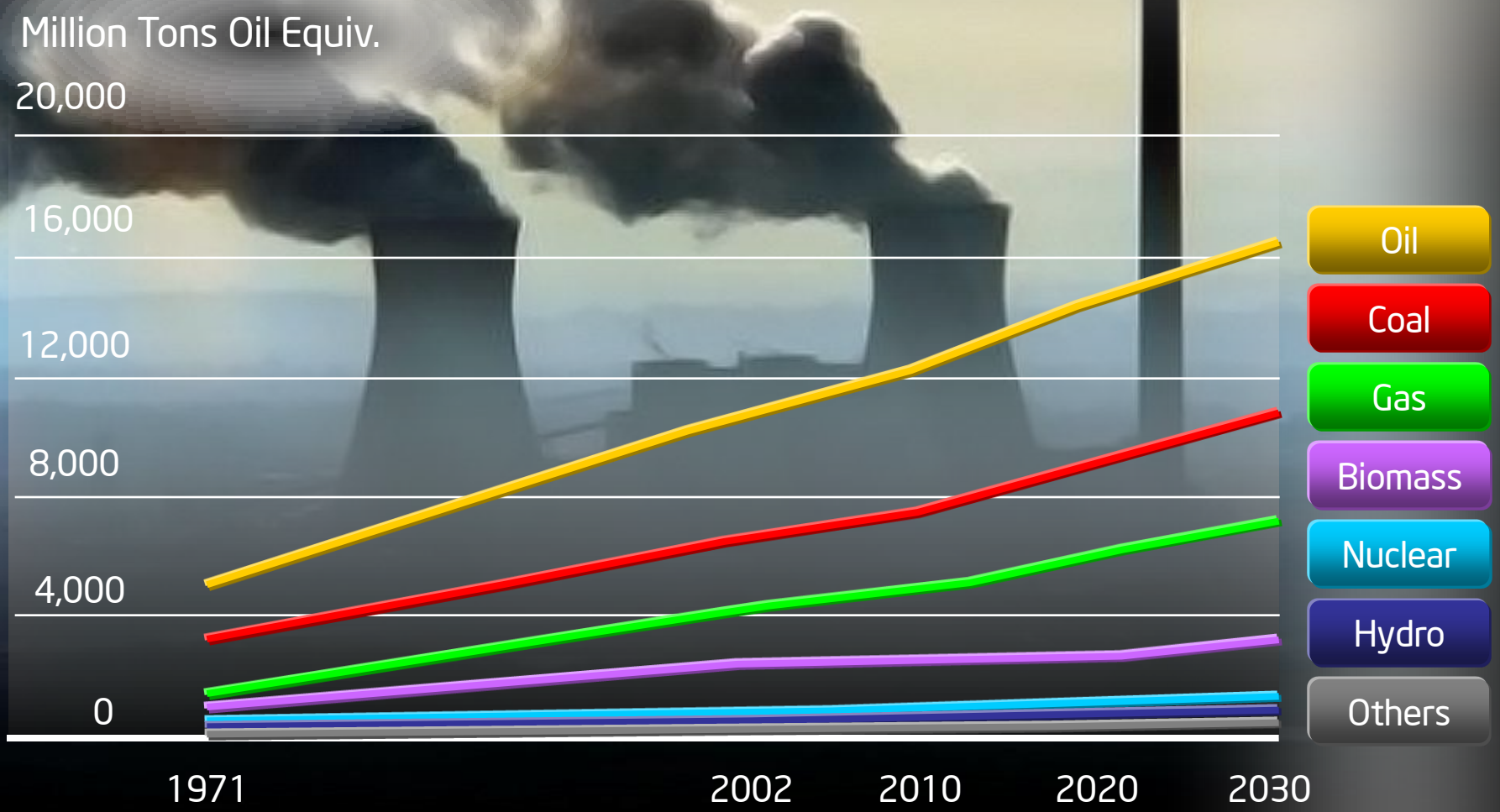


Low Cost Optical Detection of Methane Gas

Demonstrated Applications For
Molecular Spectroscopy And Gas Sensing

First Cascaded Silicon Raman Laser

World Primary Energy Demand

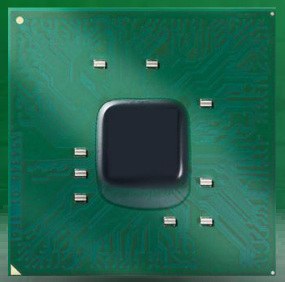


Source: IEA 2004

Holistic Approach to Energy Efficiency

Platform Power Management

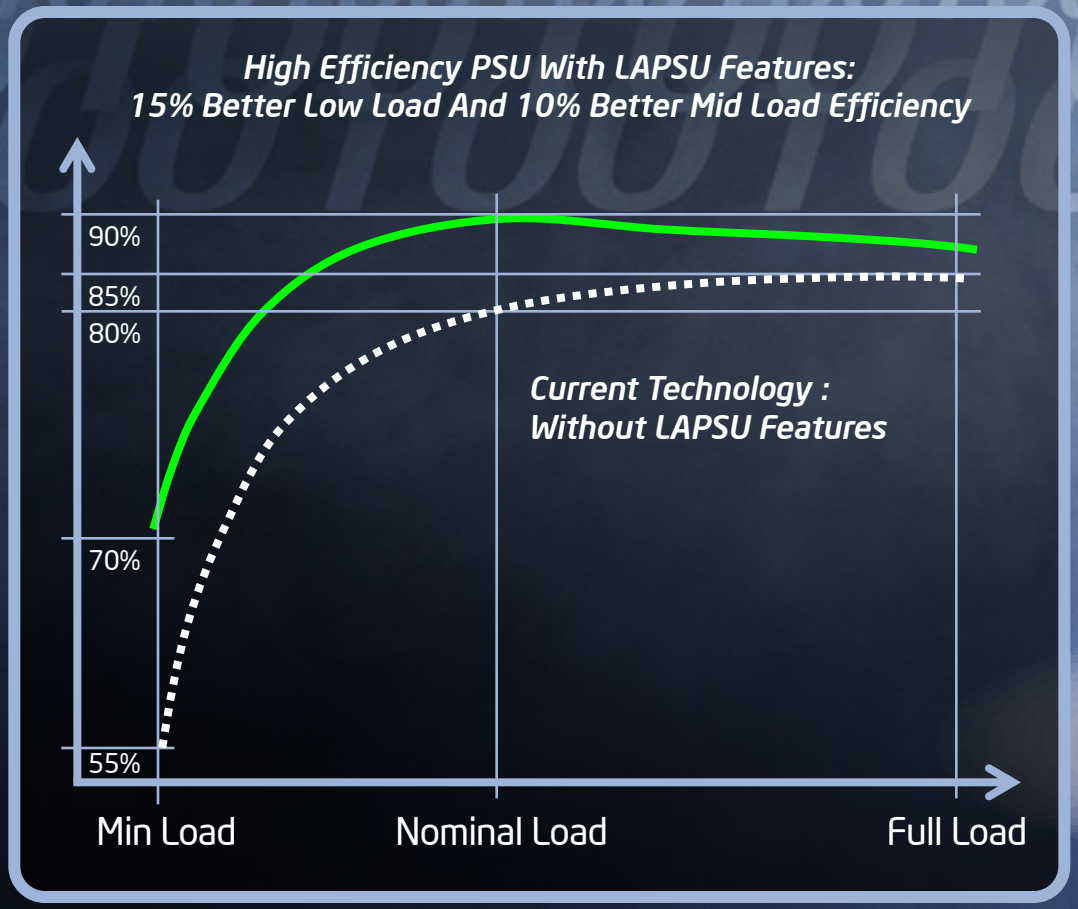
英特尔™
酷睿™ 2 双核



- Core Logic
- Operating Systems and VMMs
- Manageability
- Interconnects and Peripherals
- Telemetry
- Power Delivery and Cooling

PLATFORM POWER MANAGEMENT

Benefits of Adaptive Power Supply for Servers

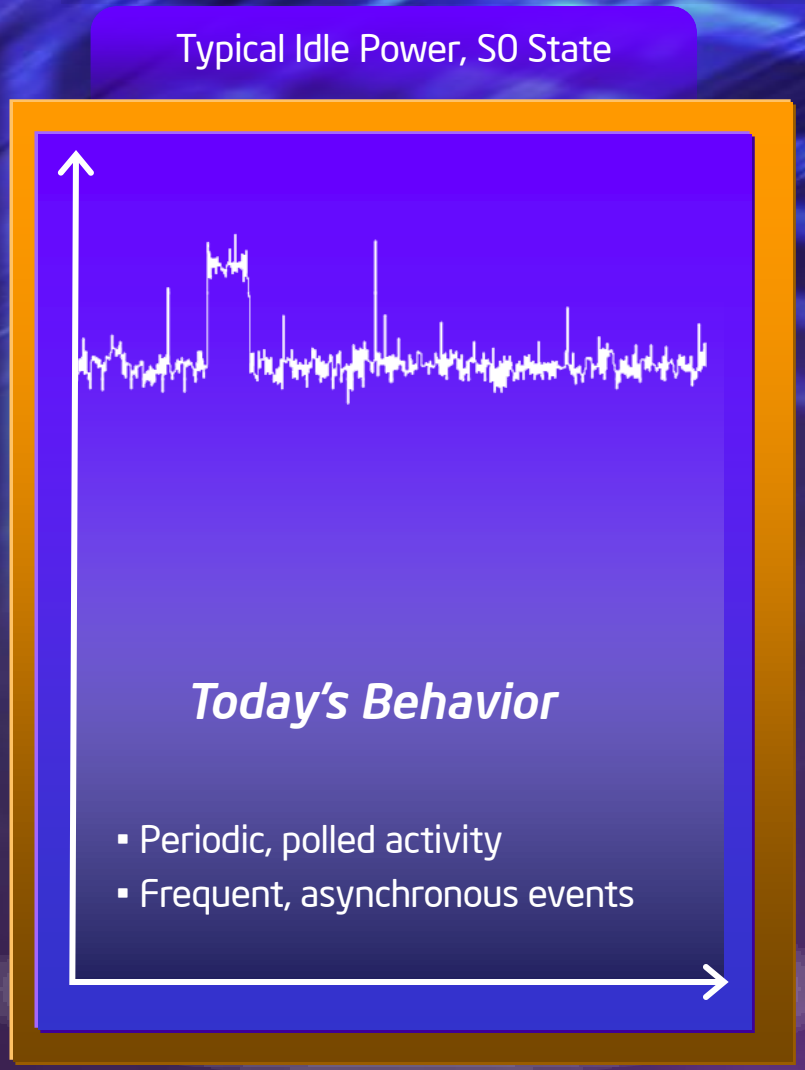


Ship Power Supplies With **70%+** Efficiency At Min Load

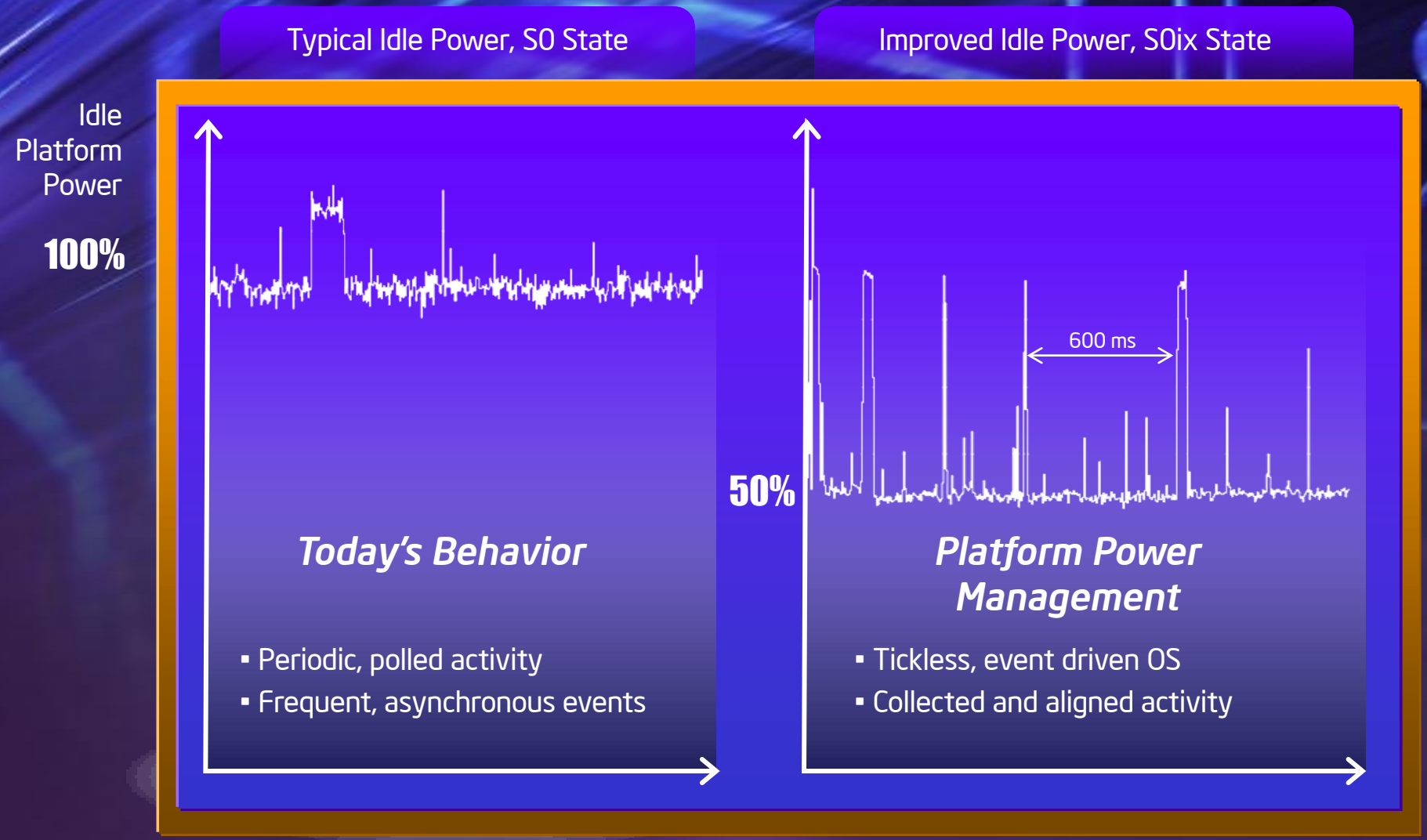
Additional Benefits Include:

~40% reduction in Bulk Capacitor or Increased Hold Up Time for the Platform

Improving Idle Efficiency



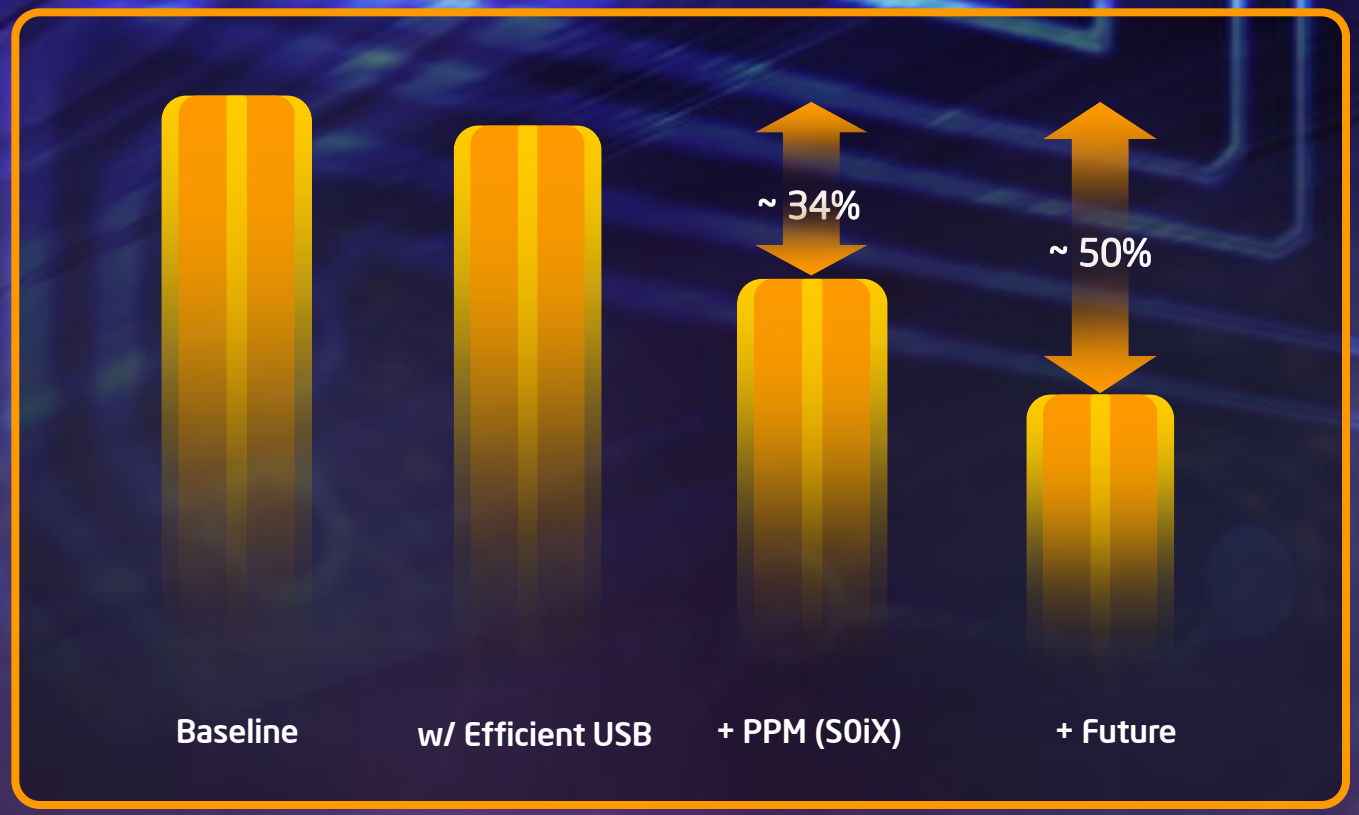
Improving Idle Efficiency



Maximum Efficiency Achieved with Platform-wide Changes

Source: projected idle power savings for thin and light notebooks (note: power savings are cumulative)

Maximum Efficiency Achieved with Platform-wide Changes



Source: projected idle power savings for thin and light notebooks (note: power savings are cumulative)

Transformation in Photography


Dr. Ren Ng
President and CEO
Refocus Imaging
refocus imaging

Light Field Photography

Conventional Camera

Captures only a 2D image
Millions of pixels
Physical processing of light rays
Very old physical constraints
Slow growth in performance

Light Field Camera

Captures entire light field
Millions of light rays
Software processing of light rays
New, unconstrained capabilities
Fast growth in performance

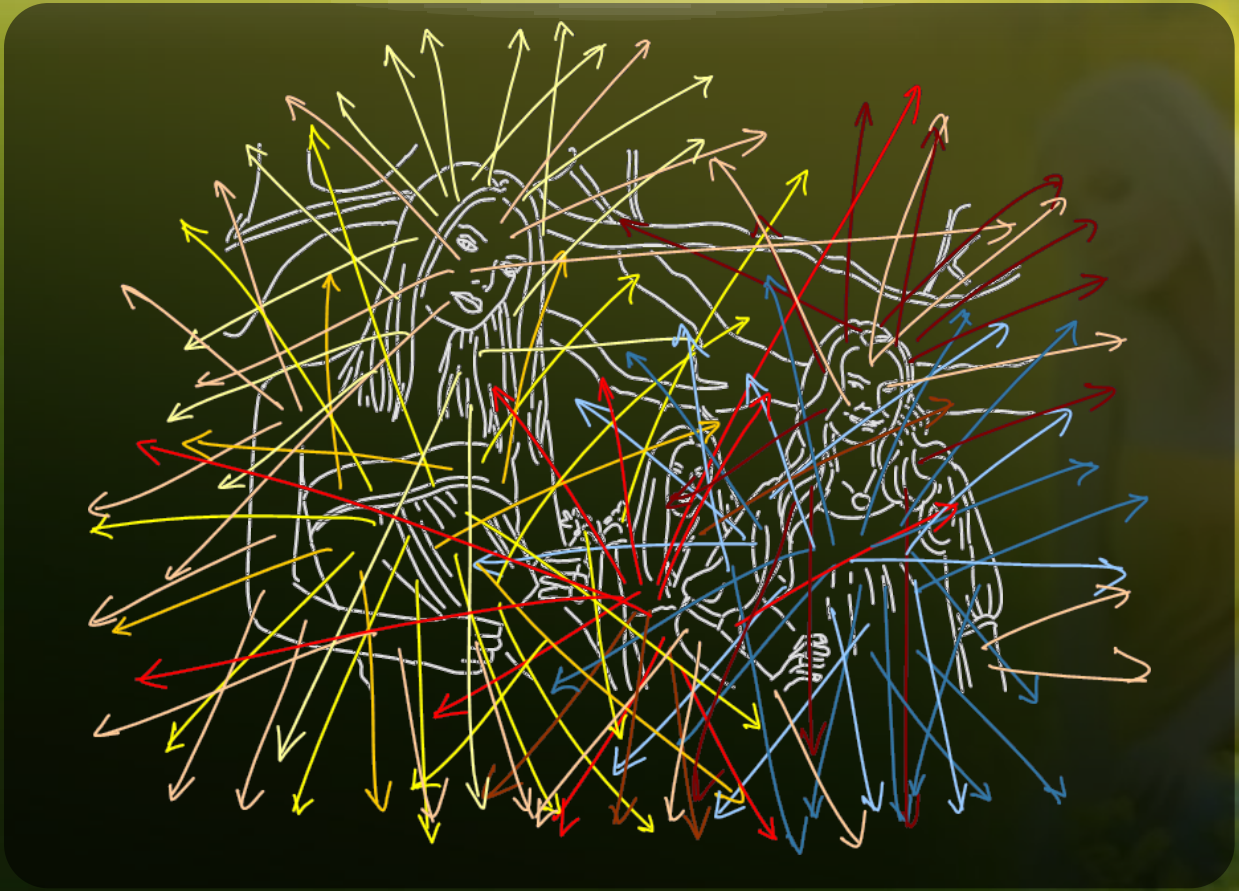
refocus

refocus *imaging*

The Light Field

refocus imaging

The Light Field



The Light Field: All Light Rays
The Light Field Produces All Photographs

refocus imaging

Light Field Applications

Cameras

Software

Printers

Lenses

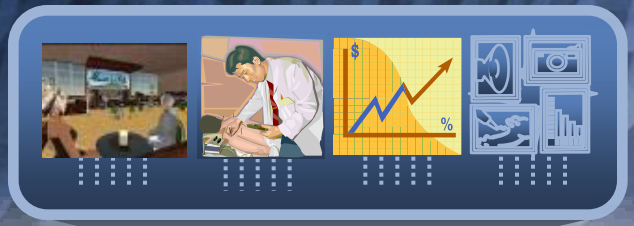
Video

Scientific

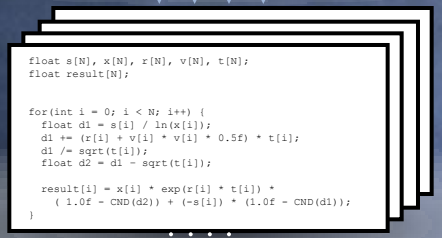
refocus imaging

Transformation in Visual Computing

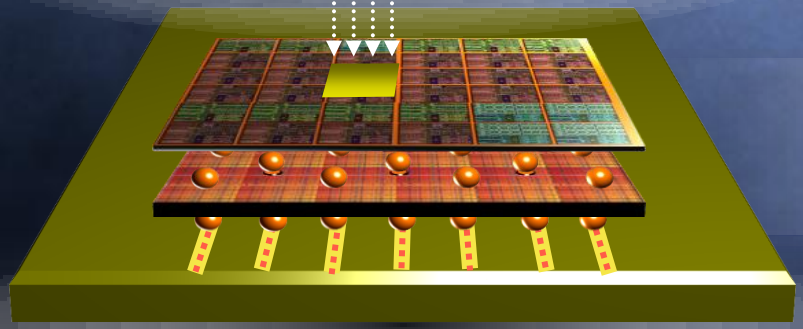
Tera-Scale Computing:
Making It a Reality



Performance-hungry Apps



Challenge: Traditional Software Focus on Single Threads



Tera-scale Platforms:
Scaling Performance via Parallelism

Parallel Programming is Hard

Jesse Fang

Managing Director
Intel China Research Center

General Co-Chair
35th ISCA, Beijing Jun 21-25, Beijing

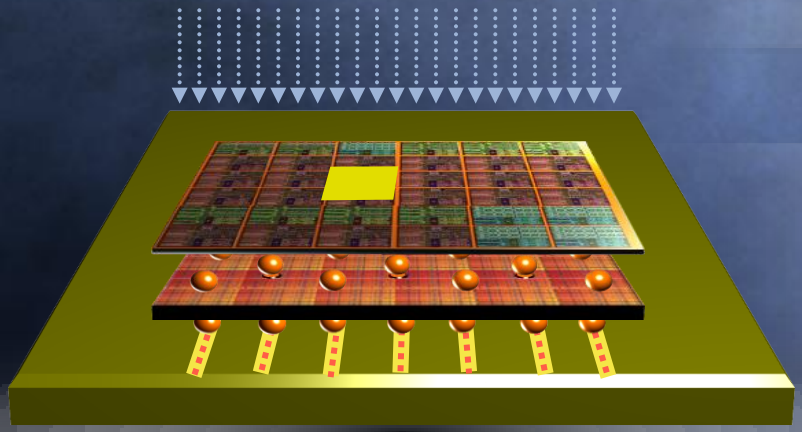
10-year
1998 - 2008
Intel China Research Center
英特尔中国研究中心10周年

2007 Best R&D Center in China
Global Entrepreneur Magazine

Why is Parallel Programming Hard?

Irregular Patterns and Data Structures

?



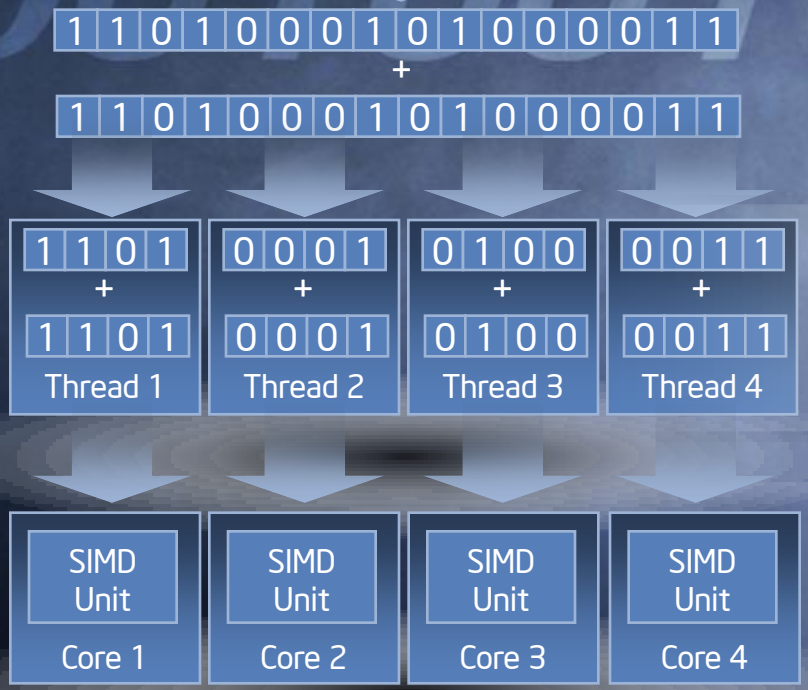
Scale to Multi-Core Today → Hard
Scale to Many-Core Tomorrow → Harder

Increasing Cores (2→64+ Cores)
Vector Instructions (4→8+ Wide)
Higher Performance Interconnect

Ct: A Throughput Programming Model

```
TVEC<F32> a(src1), b(src2);
TVEC<F32> c = a + b;
c.copyOut(dest);
```

User Writes
Core Independent C++ Code

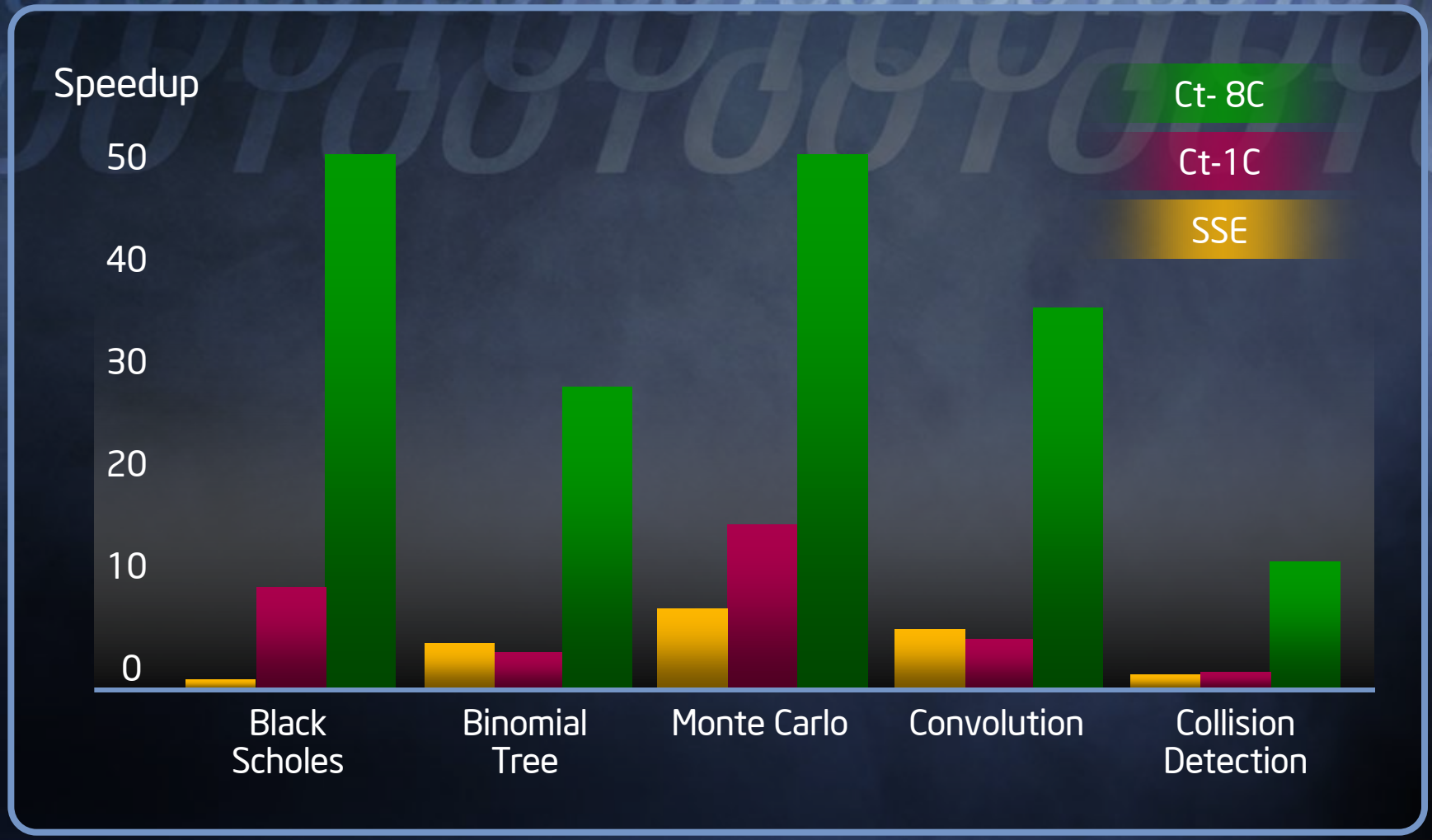


Ct Parallel Runtime:
Auto-Scale to Increasing Cores

Ct JIT Compiler:
Auto-vectorization, SSE,
AVX, Larrabee

Programmer Thinks Serially; Ct Exploits Parallelism

Excellent Scale Up




Dr. Zhang Xia
CTO & CKO, Neusoft Co. Ltd.
Neusoft®

Demo Code Sample: C vs. Ct vs. SSE

Neusoft® intel™

36 Lines ➔
Single-threaded

```
//C version of SVM
void ClrMain::calcPotValue(int k, int *x, int *y, int *pDicisionValue)
{
    int i = 0;
    int Zk = 0, qExp = 0;
    int iTemp = 0, x1 = 0, x2 = 0, xx = 0, tab1 = 0, tab2 = 0, tab3 = 0;

    for (i = 0; i < DIMENSION; i++, x++, y++) {
        iTemp = *x-*y;
        Zk += (iTemp * iTemp);
    }

    float lExp = exp(-1.0*GAMMA*Zk/65025);
    *pDicisionValue = (int)((lExp * coef[k]) * 8);
}

int ClrMain::SVM(int *x)
{
    int k = 0;
    int Desicion_value = 0;
    int aDicisionValue[TOTAL_SV+SV_EXPEND];

    for (k = 0; k < TOTAL_SV+SV_EXPEND; k++) {
        calcPotValue(k, x, y + k * DIMENSION, &aDicisionValue[k]);
    }

    for (k = 0; k < TOTAL_SV; k++) {
        Desicion_value += aDicisionValue[k];
    }

    Desicion_value -= (int)(RHO * 8 * ICOEF);
    Desicion_value = Desicion_value * 10000 / ICOEF / 8;

    return Desicion_value;
}
```

C

29 Lines ➔
Vectorized/Multi-threaded
Forward Scalability

```
//! Ct version of SVM
int ClrMain::ctSVM(int *x)
{
    int    Decision_value;

    __CT__{
        //! Copy in data
        TVEC<I32> tCoef(coef, ( TOTAL_SV+SV_EXPEND ));
        TVEC<I32> tx(x, DIMENSION);
        TVEC2D<I32> t2dY(y, ( TOTAL_SV + SV_EXPEND ), DIMENSION);

        //! compute Z
        TVEC2D<I32> t2dX = repeatRow(tx, ( TOTAL_SV + SV_EXPEND ));
        TVEC2D<I32> t2dZ = ( t2dX - t2dY );
        TVEC<F32> tz = addReduce( t2dZ * t2dZ );
        tz = exp( (TVEC<F32>)tz * ( (__F32)(-GAMMA) / 65025 ) );

        //! compute decision value
        TVEC<I32> tDecVal = (TVEC<I32>)( tz * (TVEC<F32)tCoef * (F32)8 );
        I32 decVal = addReduce( tDecVal )[0];
        decVal -= (I32)( RHO * 8 * ICOEF);
        decVal *= (I32)( 10000 / ICOF / 8 );

        Decision_value = decVal.toValue();
    }__TC__;

    return Desicion_value;
}
```

Ct

116 Lines ➔
Vectorized/Single-threaded

```
//SSE version of SVM
void ClrMain::calcPotValueSSE(int k, int *x, int *y, int
*pDicisionValue)
{
    int i = 0;
    int *y0, *y1, *y2, *y3;
    int *x0;

    static _MM_ALIGN16 float Zk[4] = { 0,0,0,0 };
    static _MM_ALIGN16 float constValue[4] =
        { -1.0*GAMMA/65025, -1.0*GAMMA/65025,-
1.0*GAMMA/65025,-1.0*GAMMA/65025,};

    __m128 xmm_iTemp0, xmm_iTemp1, xmm_iTemp2,
xmm_iTemp3;
    __m128 xmm_Zk0, xmm_Zk1, xmm_Zk2, xmm_Zk3;
    __m128 xmm_x0;
    __m128 xmm_y0, xmm_y1, xmm_y2, xmm_y3;

    __m128 xmm_Zk,xmm_coef,xmm_lExp, xmm_const;
    __m128i xmm_pDicisionValue;

    y0 = y;
    y1 = y + 1*DIMENSION;
    y2 = y + 2*DIMENSION;
    y3 = y + 3*DIMENSION;
    x0 = x;

    xmm_Zk0 = _mm_setzero_ps();
    xmm_Zk1 = _mm_setzero_ps();
    xmm_Zk2 = _mm_setzero_ps();
    xmm_Zk3 = _mm_setzero_ps();

    for (i = 0; i < DIMENSION; i+=4) {

        xmm_x0 = _mm_cvtepi32_ps(_mm_load_si128((__m128i *)
&x0[i]));
        _mm_prefetch((char *)(&x0[i] + 4 * PF_DISTANCE) ,
_MM_HINT_T0);
        xmm_y0 = _mm_cvtepi32_ps(_mm_load_si128((__m128i *)
&y0[i]));
        _mm_prefetch((char *)(&y0[i] + 4 * PF_DISTANCE) ,
_MM_HINT_T0);
        xmm_y1 = _mm_cvtepi32_ps(_mm_load_si128((__m128i *)
&y1[i]));
        _mm_prefetch((char *)(&y1[i] + 4 * PF_DISTANCE) ,
_MM_HINT_T0);
        xmm_y2 = _mm_cvtepi32_ps(_mm_load_si128((__m128i *)
&y2[i]));
        _mm_prefetch((char *)(&y2[i] + 4 * PF_DISTANCE) ,
_MM_HINT_T0);
        xmm_y3 = _mm_cvtepi32_ps(_mm_load_si128((__m128i *)
&y3[i]));
        _mm_prefetch((char *)(&y3[i] + 4 * PF_DISTANCE) ,
_MM_HINT_T0);

        xmm_iTemp0 = _mm_sub_ps(xmm_x0, xmm_y0);
        xmm_iTemp0 = _mm_mul_ps(xmm_iTemp0, xmm_iTemp0);
        xmm_Zk0 = _mm_add_ps(xmm_Zk0, xmm_iTemp0);

        xmm_iTemp1 = _mm_sub_ps(xmm_x0, xmm_y1);
        xmm_iTemp1 = _mm_mul_ps(xmm_iTemp1, xmm_iTemp1);
        xmm_Zk1 = _mm_add_ps(xmm_Zk1, xmm_iTemp1);

        xmm_iTemp2 = _mm_sub_ps(xmm_x0, xmm_y2);
        xmm_iTemp2 = _mm_mul_ps(xmm_iTemp2, xmm_iTemp2);
        xmm_Zk2 = _mm_add_ps(xmm_Zk2, xmm_iTemp2);

        xmm_iTemp3 = _mm_sub_ps(xmm_x0, xmm_y3);
        xmm_iTemp3 = _mm_mul_ps(xmm_iTemp3, xmm_iTemp3);
        xmm_Zk3 = _mm_add_ps(xmm_Zk3, xmm_iTemp3);

    }
    SUM_4ELEMENTS(xmm_Zk0, &Zk[0]);
    SUM_4ELEMENTS(xmm_Zk1, &Zk[1]);
    SUM_4ELEMENTS(xmm_Zk2, &Zk[2]);
    SUM_4ELEMENTS(xmm_Zk3, &Zk[3]);

    xmm_Zk = _mm_load_ps(Zk);
    xmm_coef = _mm_cvtepi32_ps(_mm_load_si128((__m128i
*)&coef[k]));
    xmm_const = _mm_load_ps(&constValue[0]);
    xmm_lExp = _mm_mul_ps(xmm_const, xmm_Zk);
    xmm_lExp = vmlsExp4(xmm_lExp);
    xmm_lExp = _mm_mul_ps(xmm_lExp, xmm_coef);

    xmm_pDicisionValue = _mm_cvtps_epi32(xmm_lExp);
    xmm_pDicisionValue = _mm_slli_epi32(xmm_pDicisionValue,
3);

    _mm_store_si128((__m128i *)pDicisionValue,
xmm_pDicisionValue);
}

int ClrMain::SVMSSE(int *x)
{
    int k = 0;
    int Desicion_value = 0;
    float sum;

    _MM_ALIGN16 int aDicisionValue[TOTAL_SV+SV_EXPEND];
    __m128i xmm_aDicisionValue, xmm_sum;
    __m128 xmm_sumf;

    for (k = 0; k < TOTAL_SV+1; k+=4) {
        calcPotValueSSE(k, x, y + k * DIMENSION,
&aDicisionValue[k]);
    }

    for(k = 0; k < SV_EXPEND; k++) {
        aDicisionValue[TOTAL_SV+k] = 0.0f;
    }

    xmm_sum = _mm_set_epi32(0,0,0,0);
    for (k = 0; k < TOTAL_SV+SV_EXPEND; k+=4) {
        xmm_aDicisionValue = _mm_load_si128((__m128i
*)&aDicisionValue[k]);
        xmm_sum = _mm_add_epi32(xmm_sum,
xmm_aDicisionValue);
    }

    xmm_sumf = _mm_cvtepi32_ps(xmm_sum);
    SUM_4ELEMENTS(xmm_sumf, &sum);

    Desicion_value = (int)sum;

    Desicion_value -= (int)(RHO * 8 * ICOEF);

    Desicion_value = Desicion_value * 10000 / ICOEF / 8;

    return Desicion_value;
}
```

SSE

Transformation in Robotics

Autonomous Vehicles

Winner of
2006 DARPA
Grand Challenge

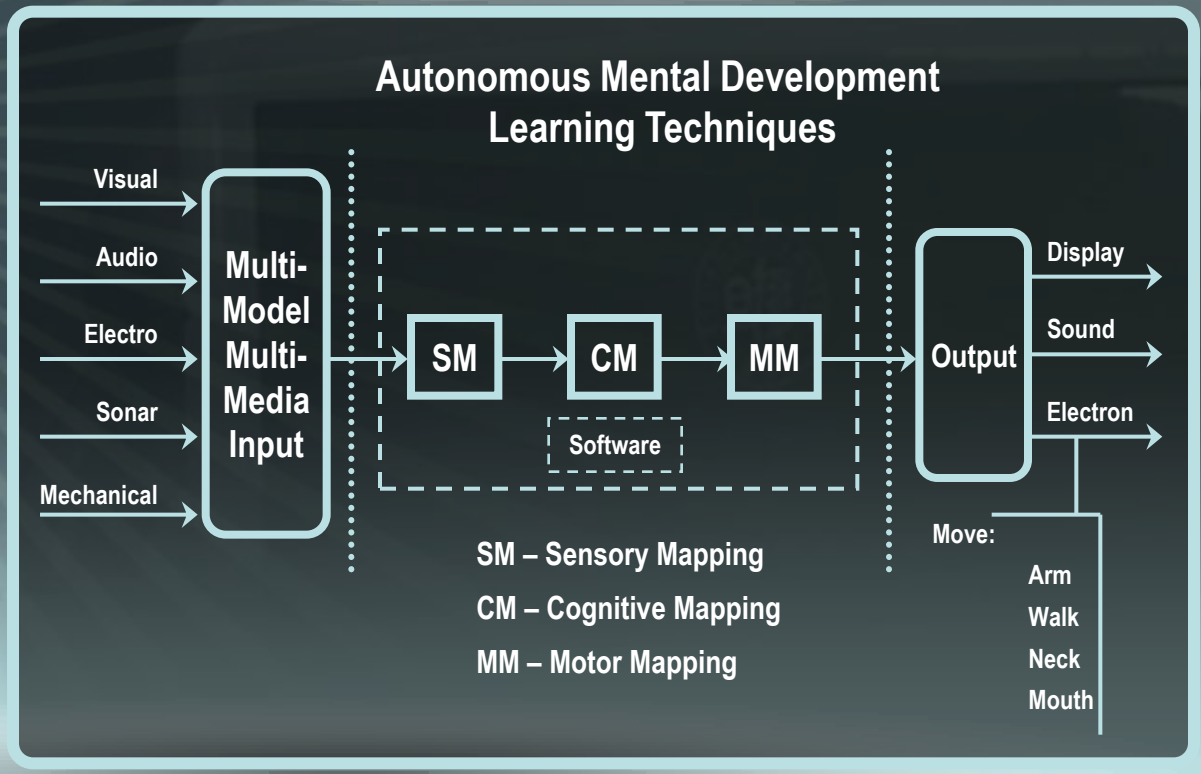
Winner of
2007 DARPA
Urban Challenge


Dr. Mingmin Chi
Associate Professor
Department of Computer Science and Engineering
Fudan University
FUDAN UNIVERSITY
1905
復旦大學
日 月 光 华 旦 复 旦 兮
中文版

Technology Behind Fuwa

- Computational cognition
- Autonomous mental development learning
- Multi-modal human-robot interaction

System Architecture



Future Vision

Fuwa has many potential applications:

- Education platform
- Entertainment console

Intel CPU is the brain of Fuwa

- More powerful computational ability
- More advanced intelligence

The Transformation in Progress ...

Improving Quality of Life • Driving Innovation • Expanding Opportunities

工欲善其事

必先利其器


40 YEARS
OF CHANGING
THE WORLD
Intel Developer
FORUM
Invent the new reality.