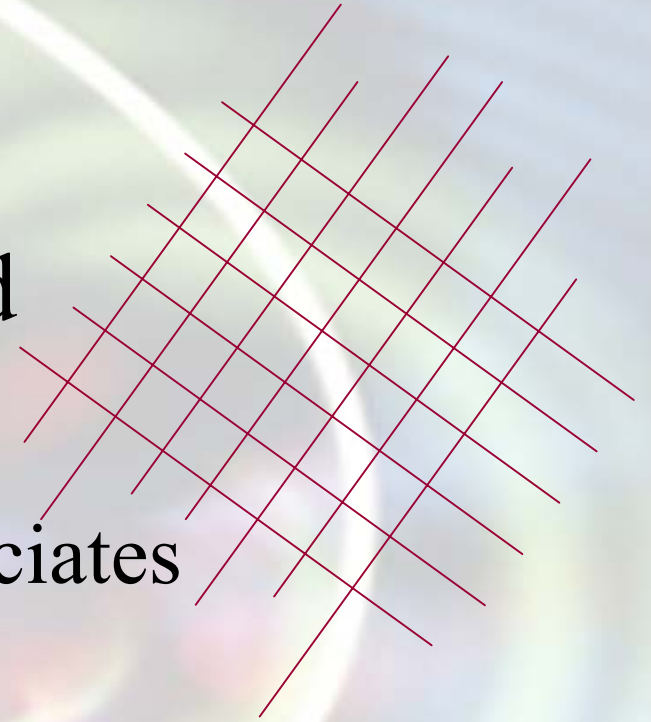


Hewitt

Early Transactional Grid Implementations

Dan Kaberon, Hewitt Associates



Enterprise Session #6

*GlobusWorld
Boston
February 2005*

Hewitt

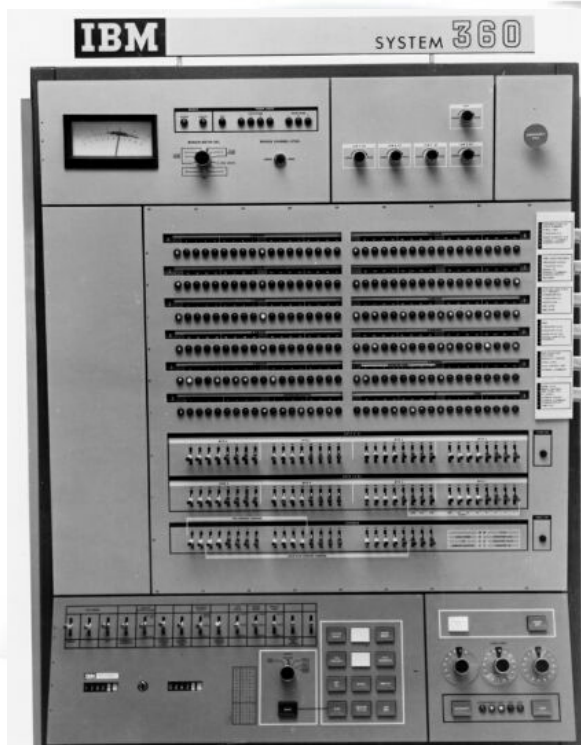
Our Mission:

To excel, around the world,
at helping our clients and their people succeed together.

Our Vision:

To be the most influential advisor and
leading service provider in the people business.

Administers benefits for one in every 20 Americans



Mainframes ?



“70 percent of the world's data are still housed in mainframe computers.”

“The IBM mainframe is truly a phoenix that has risen from the ashes.”
Gartner

The total delivery of zSeries computing power as measured in MIPS increased nearly 100 percent in the quarter compared with the first quarter of 2003. *IBM Q1-04 Report*

Mainframes: Why they thrive in commercial:



• **Establishing and growing complex workloads is relatively simple with well-grooved Gold-standard roadmaps for:**

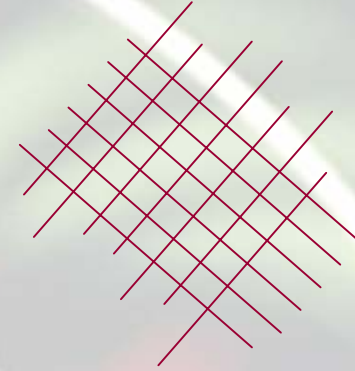
- Security
- Scalability
- Storage Management
- Data Archiving
- Performance Mgmt.
- High Availability
- Disaster Recovery
- Transaction Processing
- System Auditing
- Configuration Mgmt.

Mainframe: Downsides

- **High costs of**
 - Hardware for processing power
 - Software based on processing power



Role of Grid



Dramatic improvements in:

- **Performance through large-scale parallelism**
- **Unit costs**
- **Work *with* Mainframe configurations without overall pains of distributing data and applications**

GRID



At Hewitt Associates

Pension Calculation Engine

- **Calculates employee Pension Benefits based on *What If* input**
- **Written in SmallTalk**
- **All data passed to application in Call and results returned to caller**
- **No database**

Grid Application #1 Pension Calculation Engine

- **Large CPU cost with very widely skewed requirements**
- **Load was hard to predict**
- **Rewriting application unfeasible**
- **Batch workload too!**

Grid Solution

CICS calls Grid through SOAP

**Data Synapse GridServer employed for
management and support**

**Small Grid (about 16 parallel processors) using
Linux**

Minimal application changes



IBM @server z900 /z990



*z/OS 2.4 Parallel Sysplex
CICS*



Data Synapse Engines

IBM @server BladeCenter

Linux Red Hat 2.1



Results

- **Performance: Small calcs about the same, Large calcs much faster on Grid**
- **Migrated to Production September 2003**
- **Costs: zSeries with BladeCenter in Grid**
 - about 90% less than zSeries alone
 - Initial offload = 1800 z-mips

Established credibility for Grid approach

Grid Application #2

Composed Print (Sefas) application

Generating 100,000's output pages per job

PDF and PostScript output

On mainframe, single runs projected to run for a week or more *each!*

GRID 2, Composed Print

Use NFS Filer as shared storage device between mainframe and Grid

Broke input data into 'Chunks'

- **Split & Written by MVS**
- **Read by Grid Engines to Process**
- **Grid Output**
- **Mainframe print accepted**



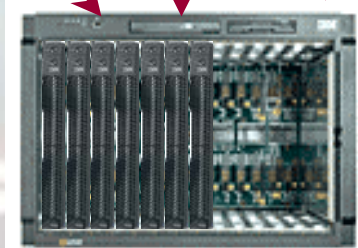
IBM @server z900 /z990



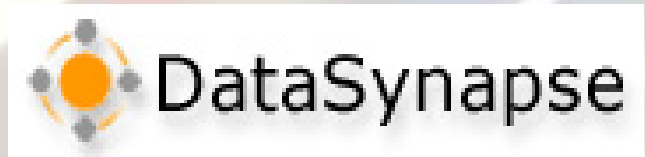
z/OS 2.4 Parallel Sysplex
CICS



Network Appliance
NFS Host

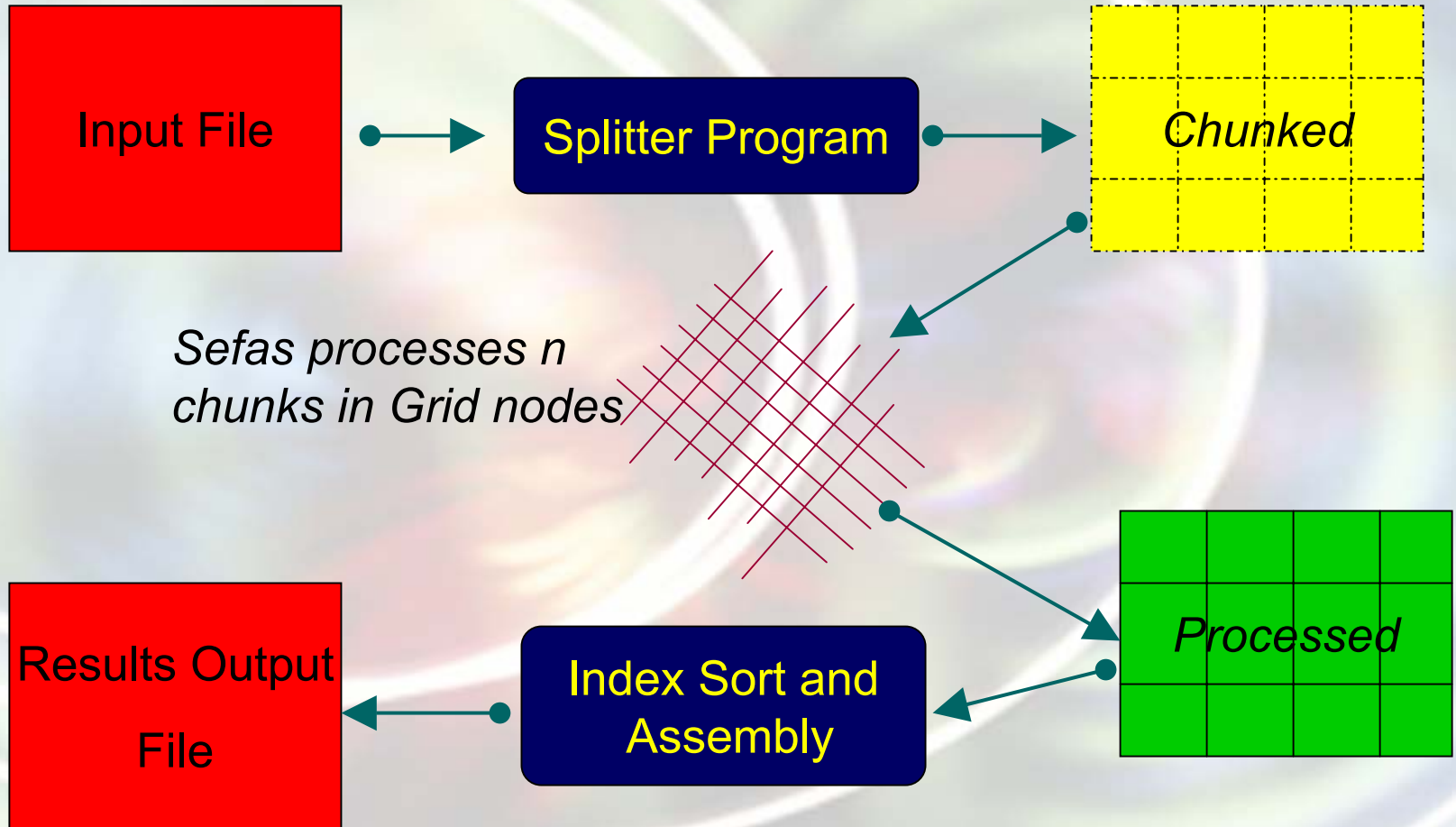


Data Synapse Engines



IBM @server BladeCenter

Sefas Process Flow



Mainframe

NFS

Composed Print Results

In September, a job projected to require about 300 elapsed hours ran on Grid in 14 hours

- Hewitt very happy to achieve all performance objectives for addressing client needs**
- SEFAS happy to cost-effectively achieve performance goals without reqwrite**
- Highly scalable solution with excellent reliability**

The background features a series of overlapping, semi-transparent white circles of varying sizes. A soft, multi-colored rainbow gradient is applied across the scene, creating a dreamy, ethereal atmosphere. The colors transition from light blues and purples on the left to greens and yellows on the right.

*Observations
and
Guidance*

***Division of Labor* : Grid capabilities
can augment Mainframe strengths**

Powerful Grids can be assembled from very high performance, near-commodity priced systems using very lightweight inexpensive software stacks

***Outsource* some mainframe workloads to Grids that do not exploit mainframe special attributes**

Gain large parallelism

Dramatic processing cost reduction

Maintain centralized controls



I/O requires Postage



There is a cpu **cost** to export a request to a Grid

There is a cpu **cost** to import the results from a Grid

If **costs** exceed the offloaded processing there can be no savings

When evaluating applications in addition to total CPU, consider CPU per I/O

Note: Remote SQL is I/O too!



Grids Now!

Old Question:

Where should we run the application?

Right Question?

How should we host the application?

**Manage data and transactions on mainframe
but exploit Grid for Parallelism and Economy**

