Common Challenges in scaling out data-oriented systems

Pawel Plaszczak
& the GridwiseTech team

www.gridwisetech.com
pawel@gridwisetech.com

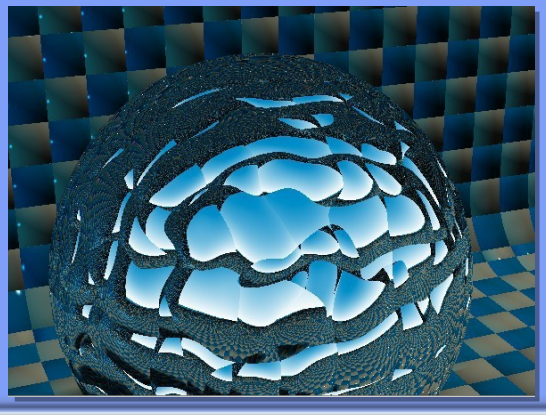
About GridwiseTech

***Mission**: turn avant-garde technology into customer's business benefits*

Leading vendor-independent scalable technology expert.

Build distributed scalable data-oriented architectures (Momentum Methodology)

Implement vision for distributed infrastructures.



Our references

Chosen customers

Chosen partners

Implementing vision: examples

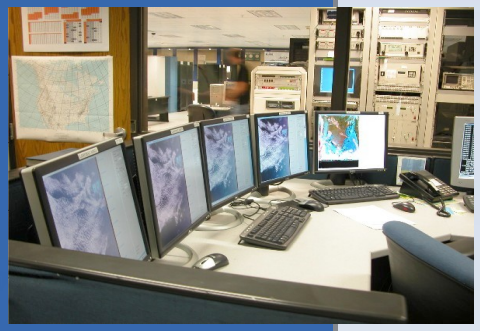
- Ricoh, Japan, leader in office automation: roadmap and prototyping on rolling out new technology
- MIMOS Berhad, Malaysia's research agency: national infrastructure to provide resources to industry
- Turner Broadcasting, owner of CNN: management workshops on identifying scalable technologies

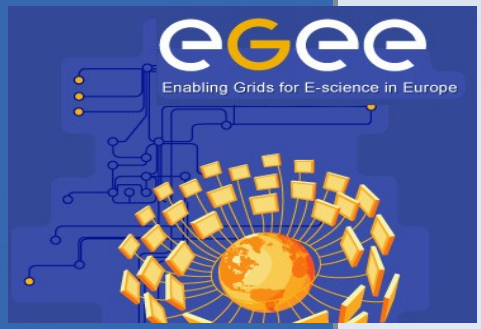
Comprehensive tech assistance: examples

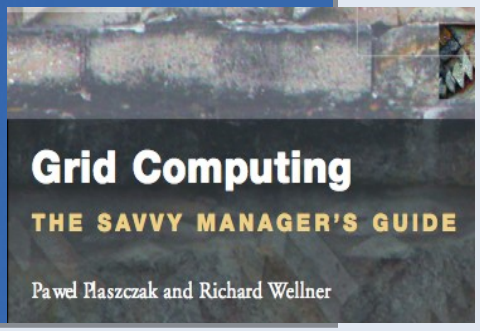
- **BP**, oil&gas giant: infrastructure optimizing
- **Philips Research**: distributing simulations
- **Fortis Bank**
- **FEI**, electron microscopy: complex integrations
- **Western Digital**, leading hard drive manufacurer: processing optimization

Active in the research community

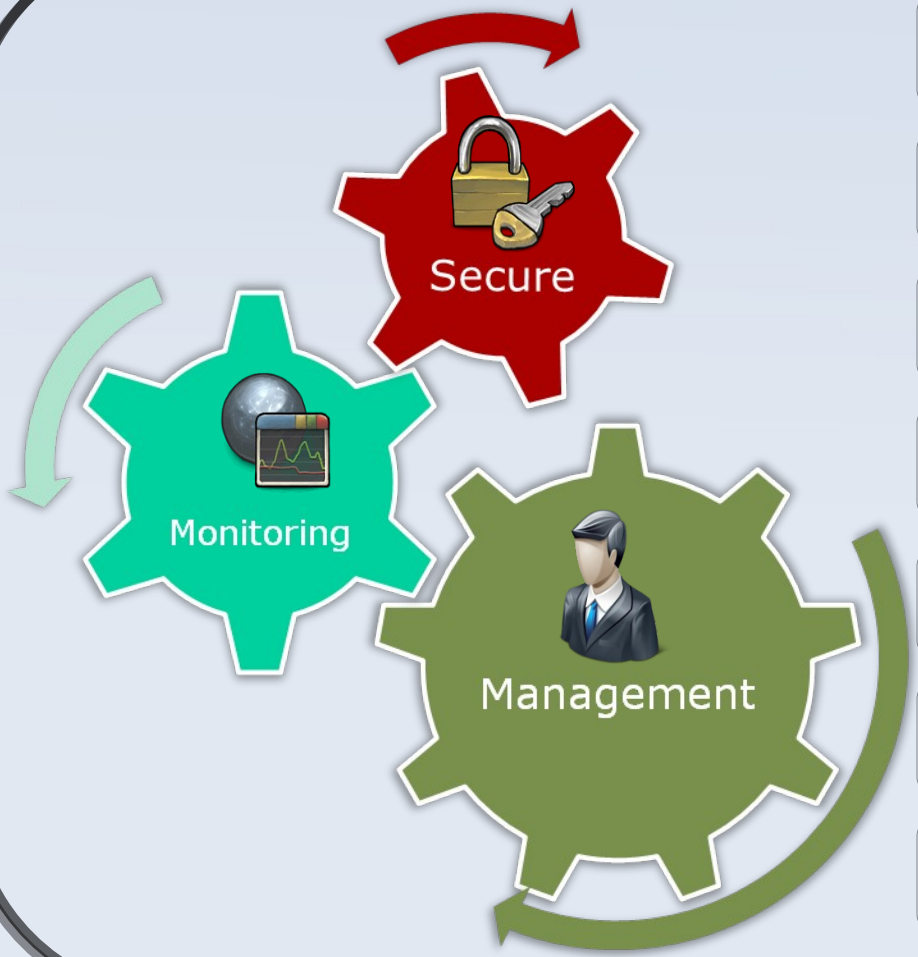
- *„Grid computing: the savvy manager's guide"*, Morgan Kaufmann, 2005
- Globus, NEESGrid: US flagship Grid projects
- Open Grid Forum, leading knowledge exchange for Grid computing experts: Grid Primer for managers
- EGEE business associate (world's most distributed infrastructure)
- Public whitepapers





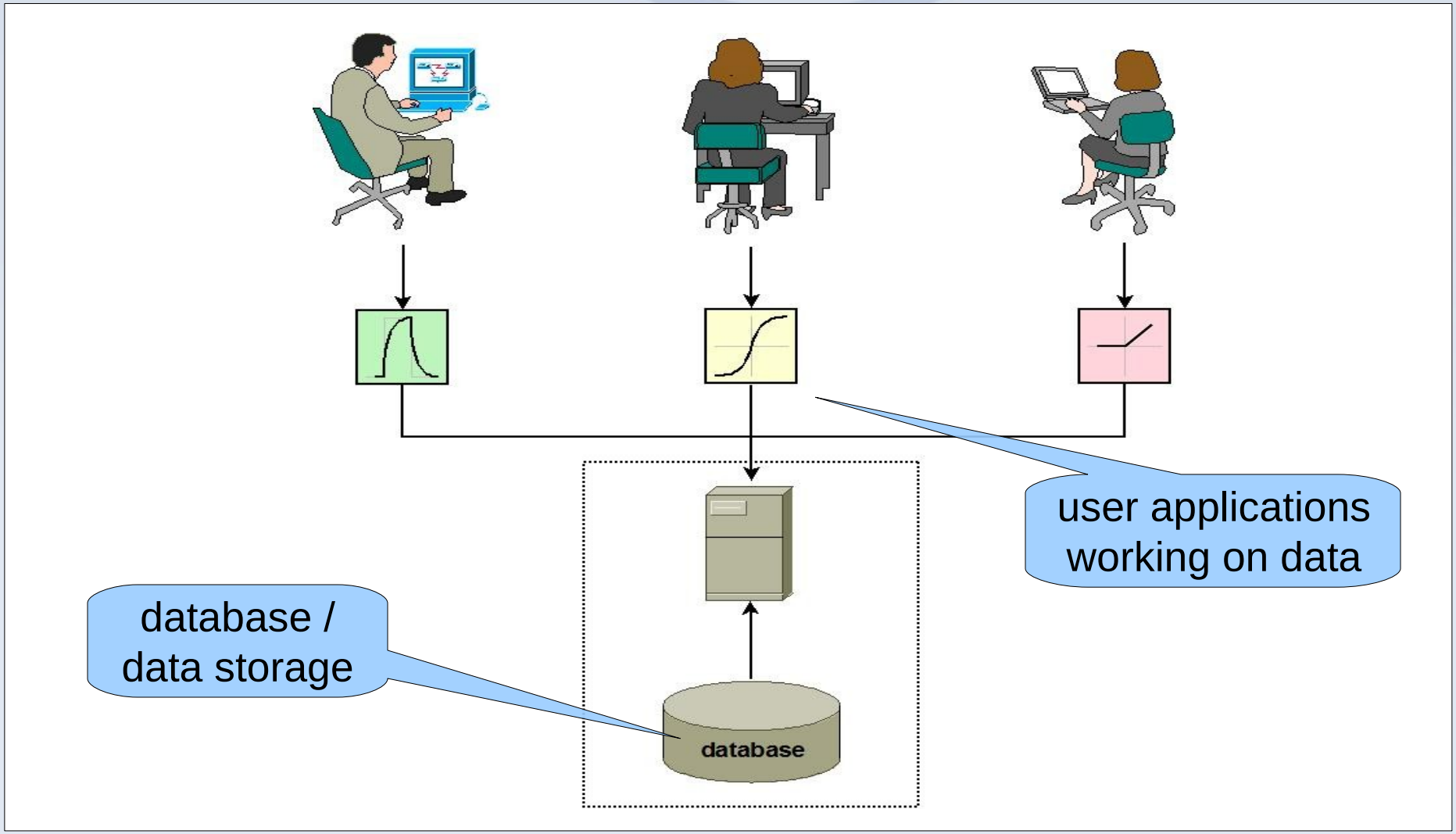


Building a scalable system

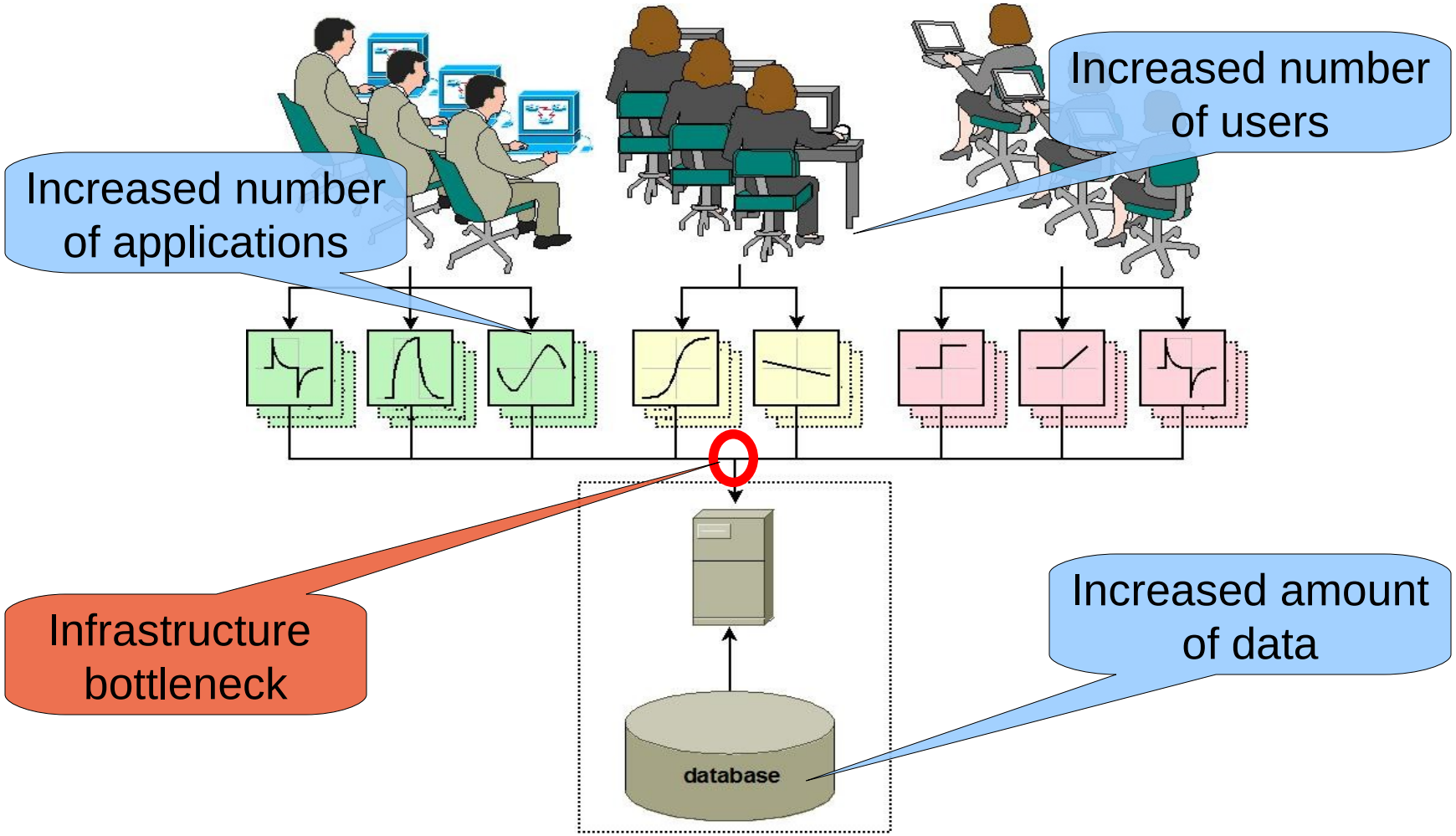


- Application
- WWW
- SOA
- Virtualization
- Grid processing
- Data provisioning
- Data persistence

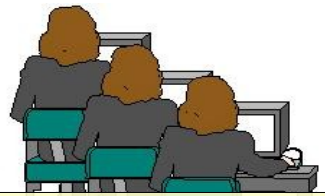
Most data-oriented systems are designed with no growth in mind



IT systems are constantly growing



IT systems are constantly growing



Increased number of users

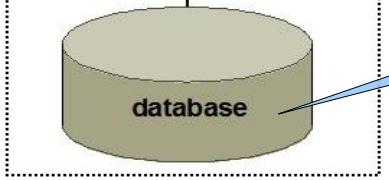
Increased number of applications

Our experience:

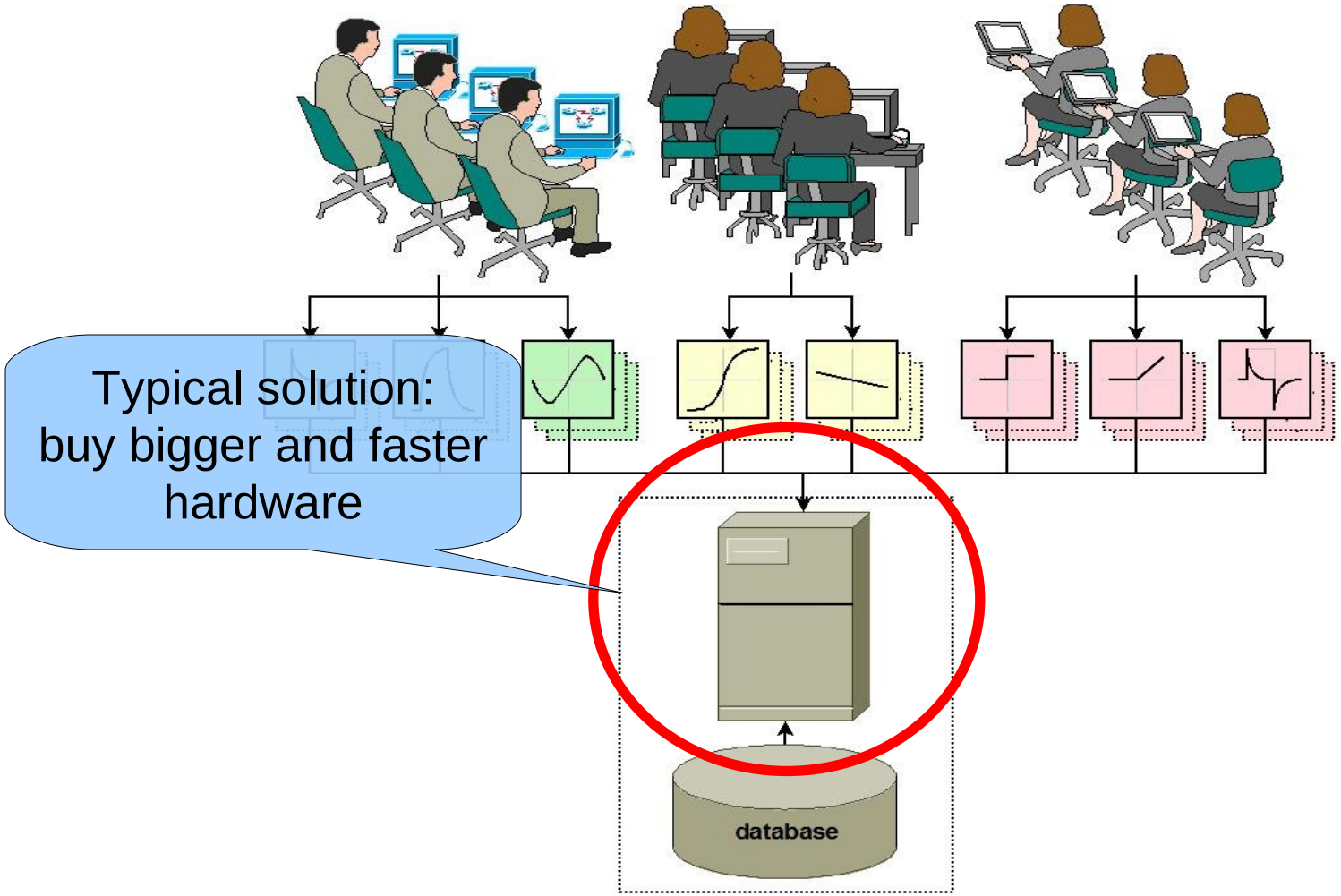
- **Financial sector.** Merger of two banks.
  Daily report generation time increased by 60%.
- **Energy sector.** Mining industry IT center.
  Calculation of miners salaries take more than 6 hours.

Infrastructure bottleneck

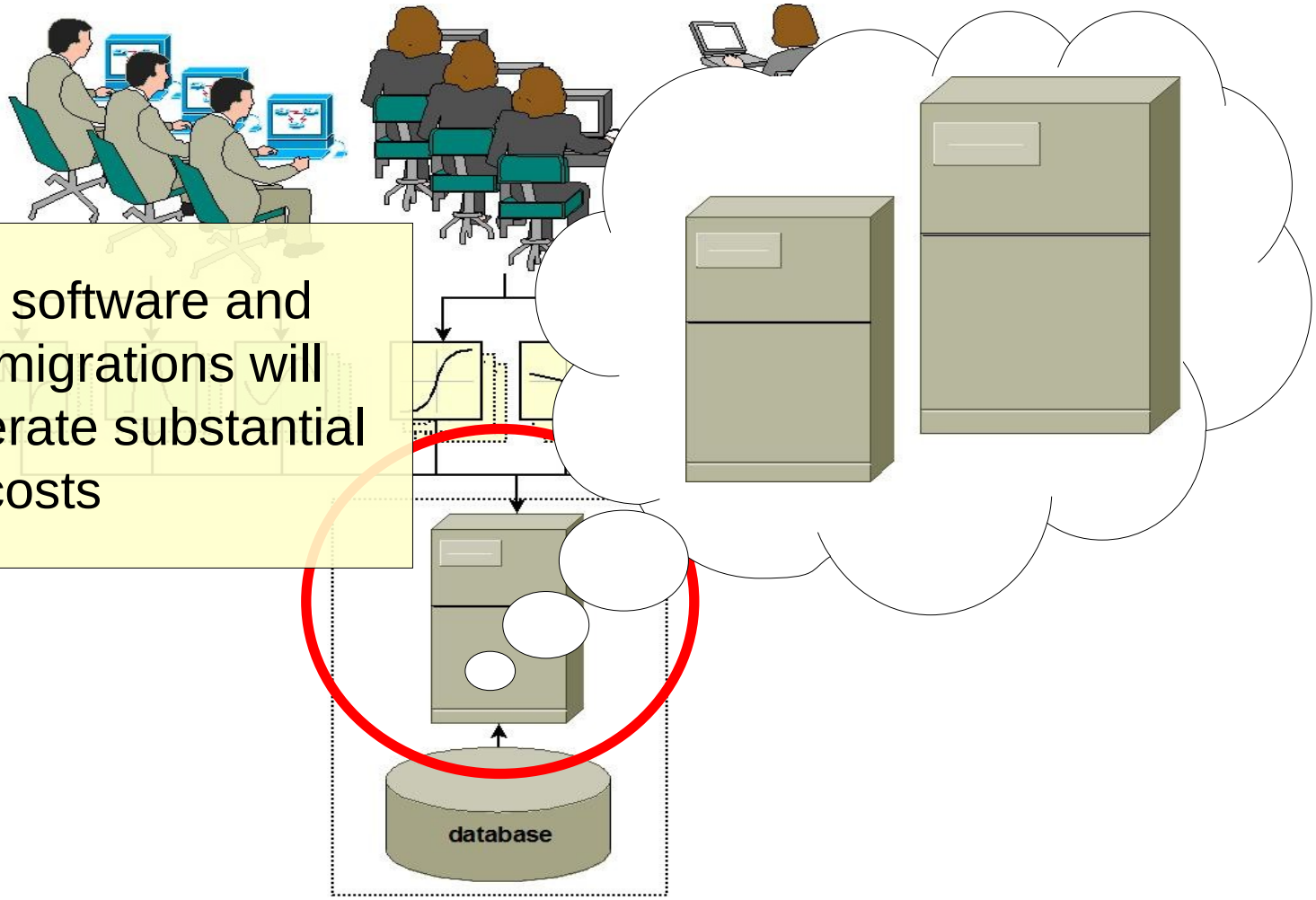
Increased amount of data



Hardware scale up – the first idea



Hardware scale up – infinite loop



Recurring software and hardware migrations will always generate substantial costs

Hardware scale up – examples

Our experience:

Production sector. Manufacturer suffers from lack of possibility to increase factory production due to inefficient report generation from production process.

Scope of the problem:

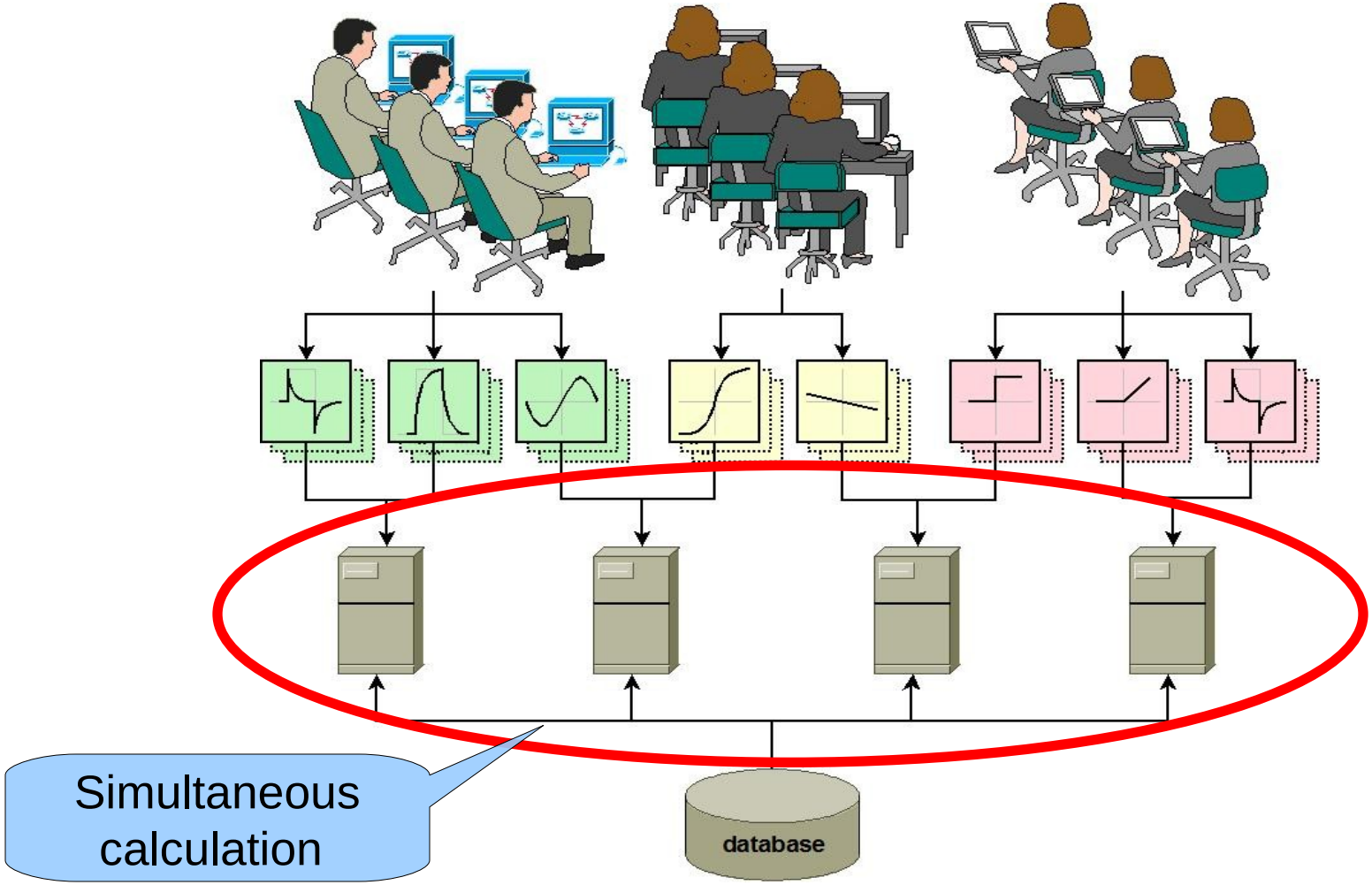
- Up to 60 minutes to generate daily report
- 2,5 millions transactions per day
- three databases, different vendors and/or versions
- 4Tb of data to maintain
- constant environment modifications
- 400,000 lines of PL/SQL code to analyze

Server extension results:

- 3 months of constant administrative work
- software compatibility problems
- decreased system stability
- problem remains unsolved

Financial sector. Three subsequent storage array upgdades over six years

Introduction of distributed processing

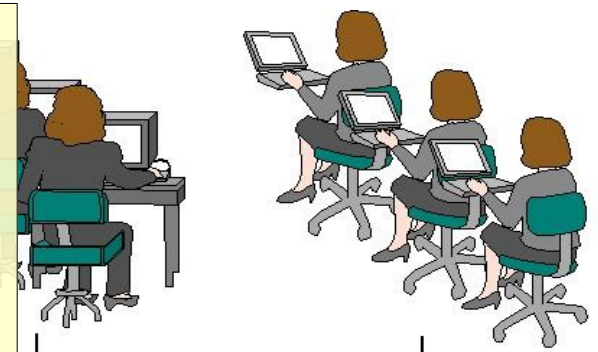


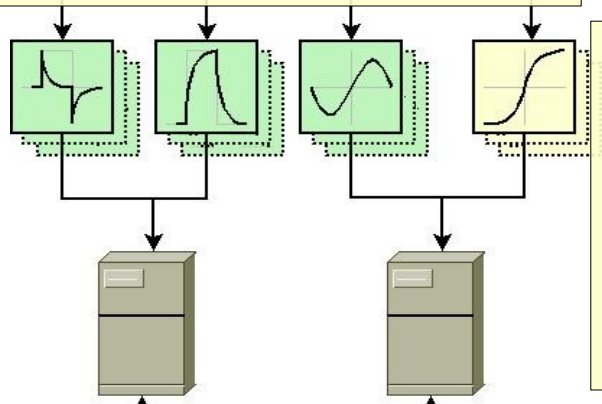
Introduction of distributed processing

Production sector

Proof of concept results for manufacturer:

- 40% of maximum possible speed up
- reads/writes increased to 10,000 / second
- scalability only up to 3 nodes

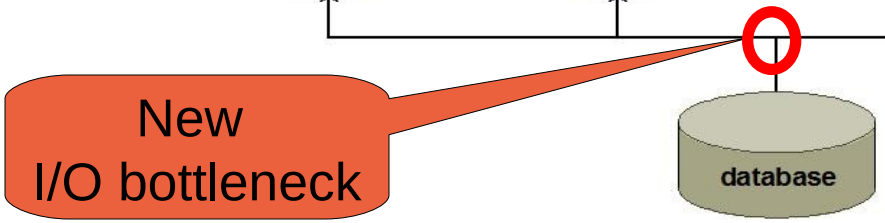




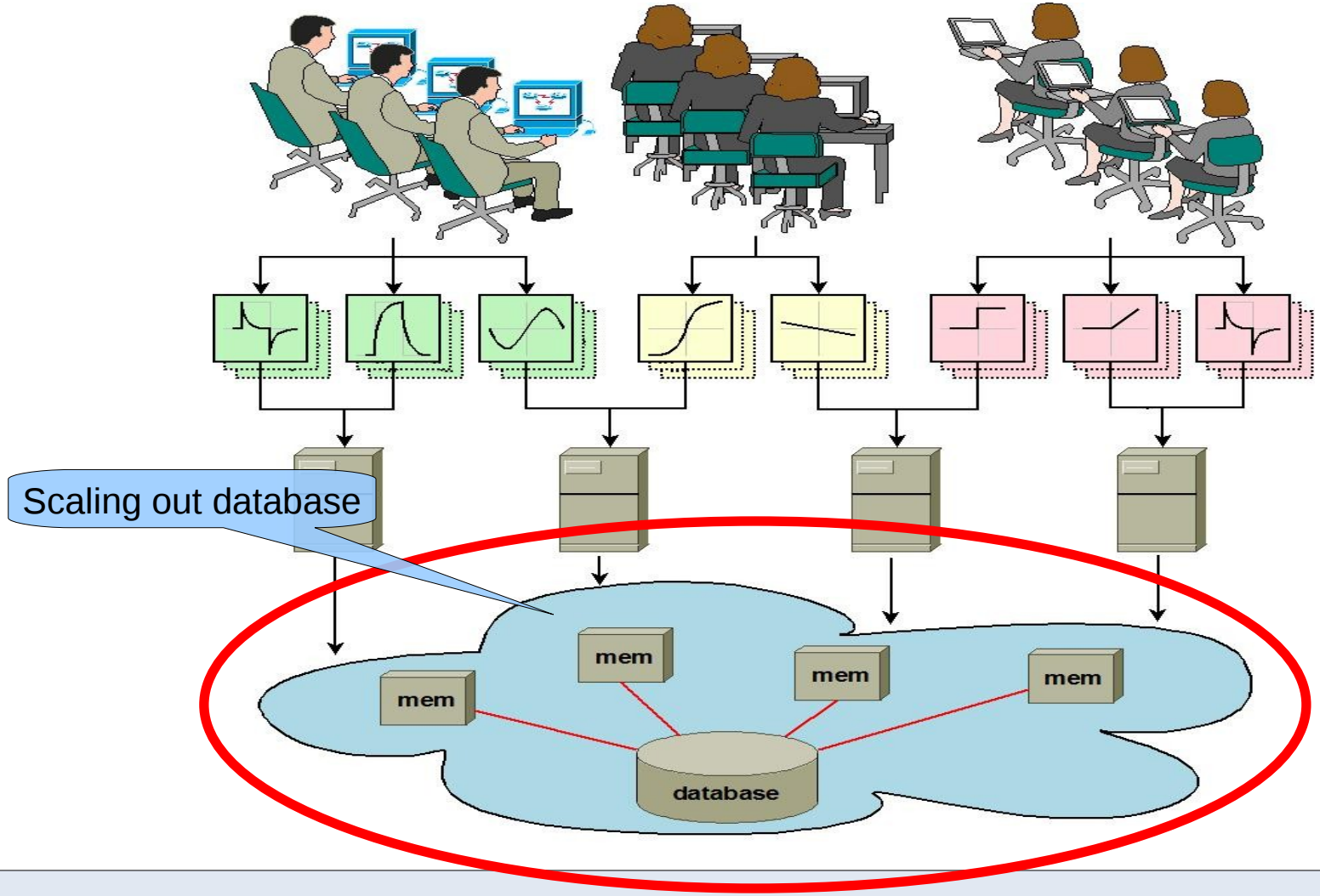
Entertainment sector

Post-production (animated movie producer):

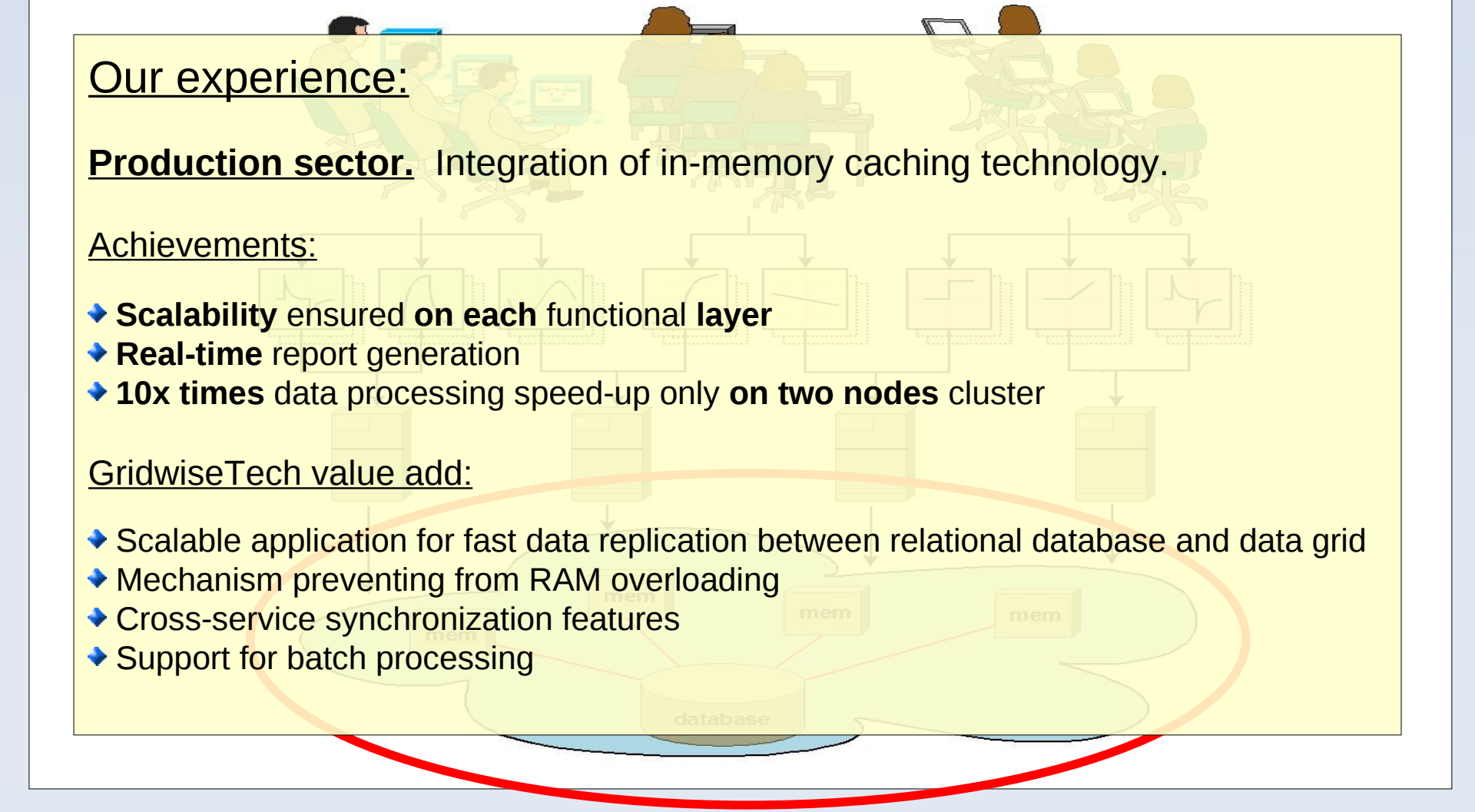
- 15 times of maximum possible speed up
- effective usage of 25% of cluster nodes due to limited access to storage area



Fully scalable system



Fully scalable system



Our experience:

Production sector. Integration of in-memory caching technology.

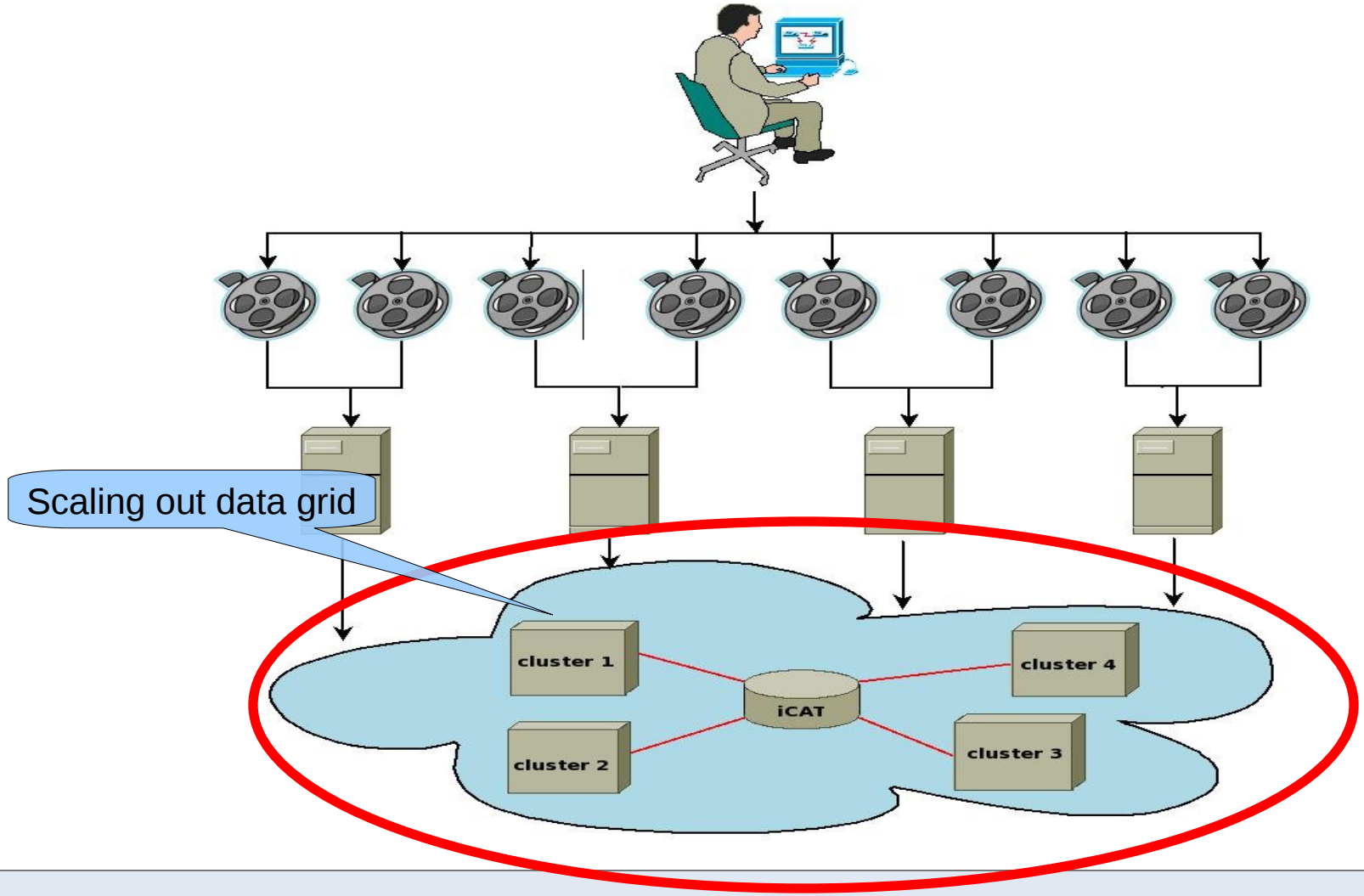
Achievements:

- **Scalability** ensured **on each** functional **layer**
- **Real-time** report generation
- **10x times** data processing speed-up only **on two nodes** cluster

GridwiseTech value add:

- Scalable application for fast data replication between relational database and data grid
- Mechanism preventing from RAM overloading
- Cross-service synchronization features
- Support for batch processing

Fully scalable system



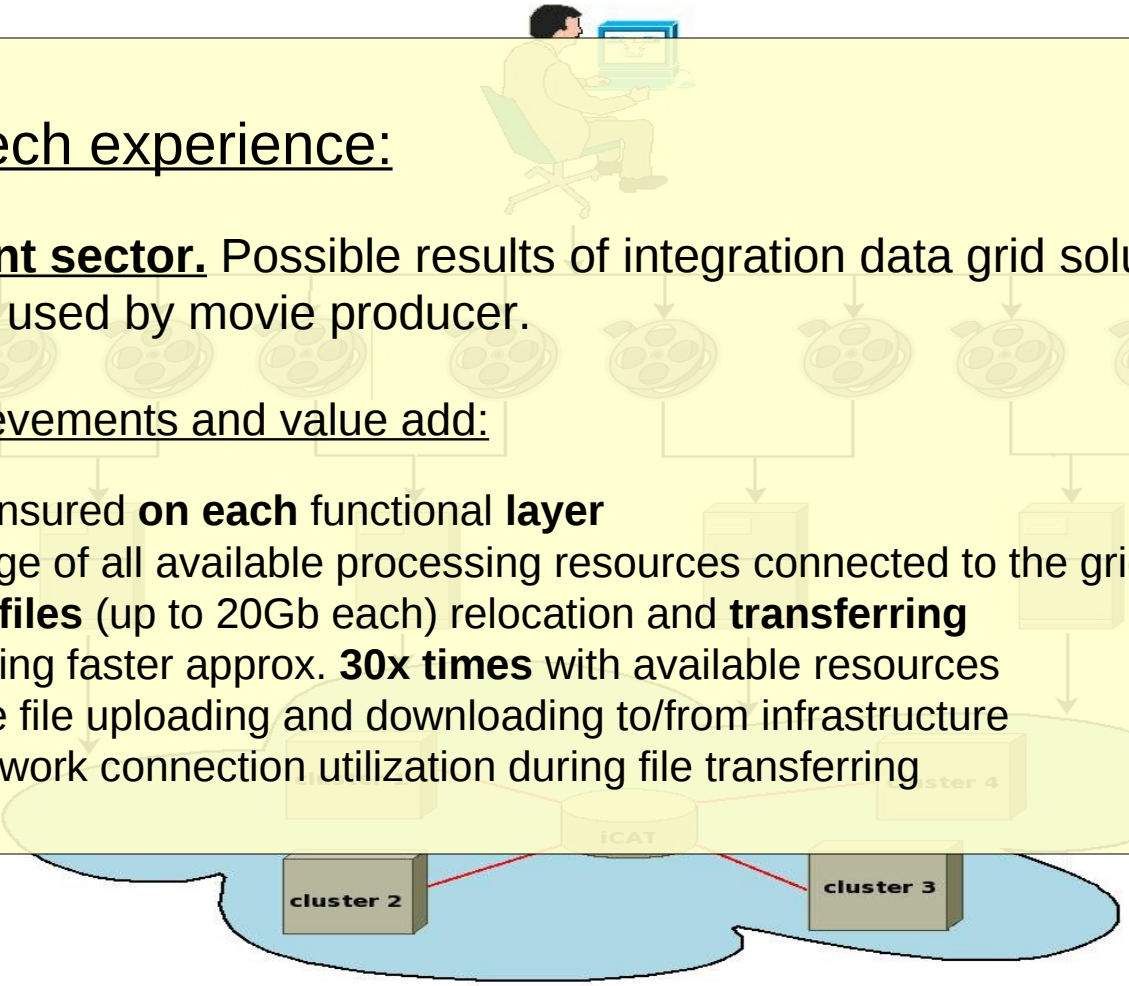
Fully scalable system

GridwiseTech experience:

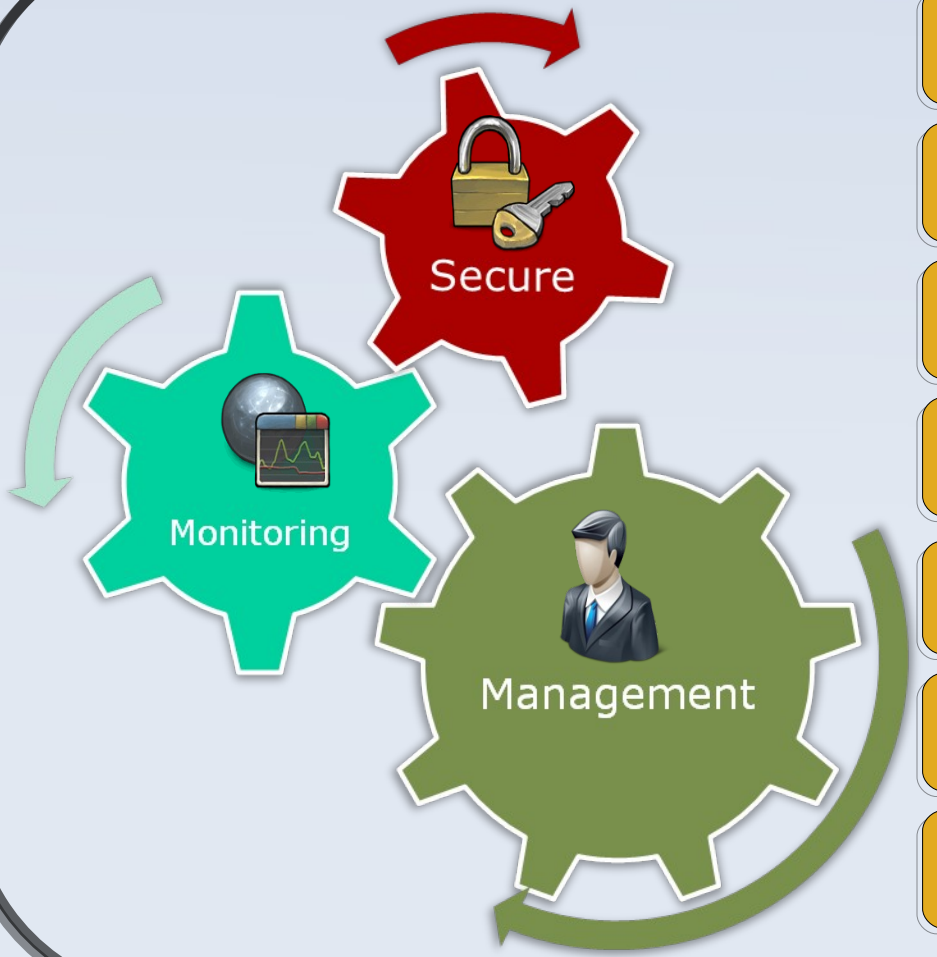
Entertainment sector. Possible results of integration data grid solution with infrastructure used by movie producer.

Possible achievements and value add:

- **Scalability** ensured **on each** functional **layer**
- Effective usage of all available processing resources connected to the grid
- **Efficient** big **files** (up to 20Gb each) relocation and **transferring**
- Movie rendering faster approx. **30x times** with available resources
- Smart remote file uploading and downloading to/from infrastructure
- Improved network connection utilization during file transferring

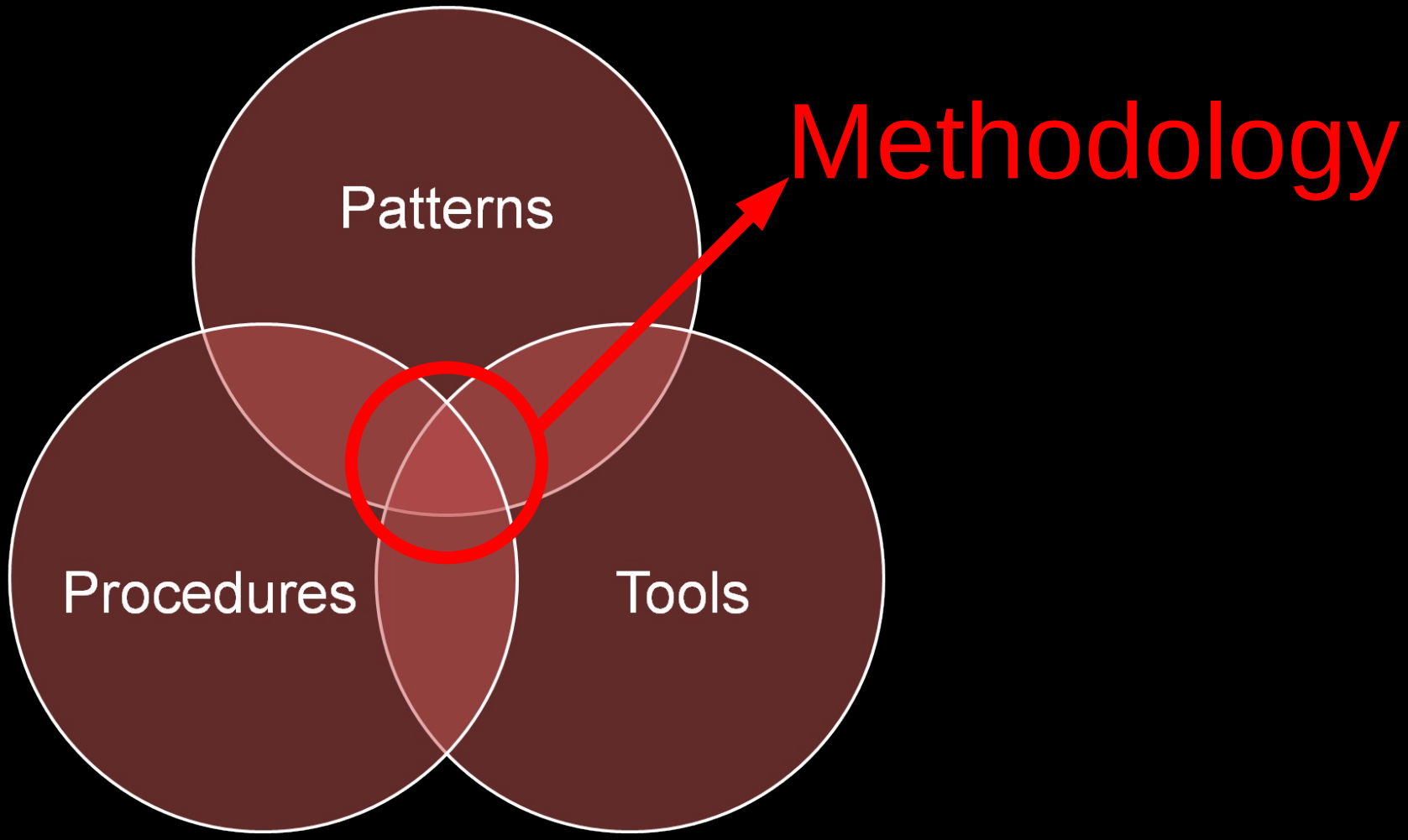


Conclusion: scalability at each layer

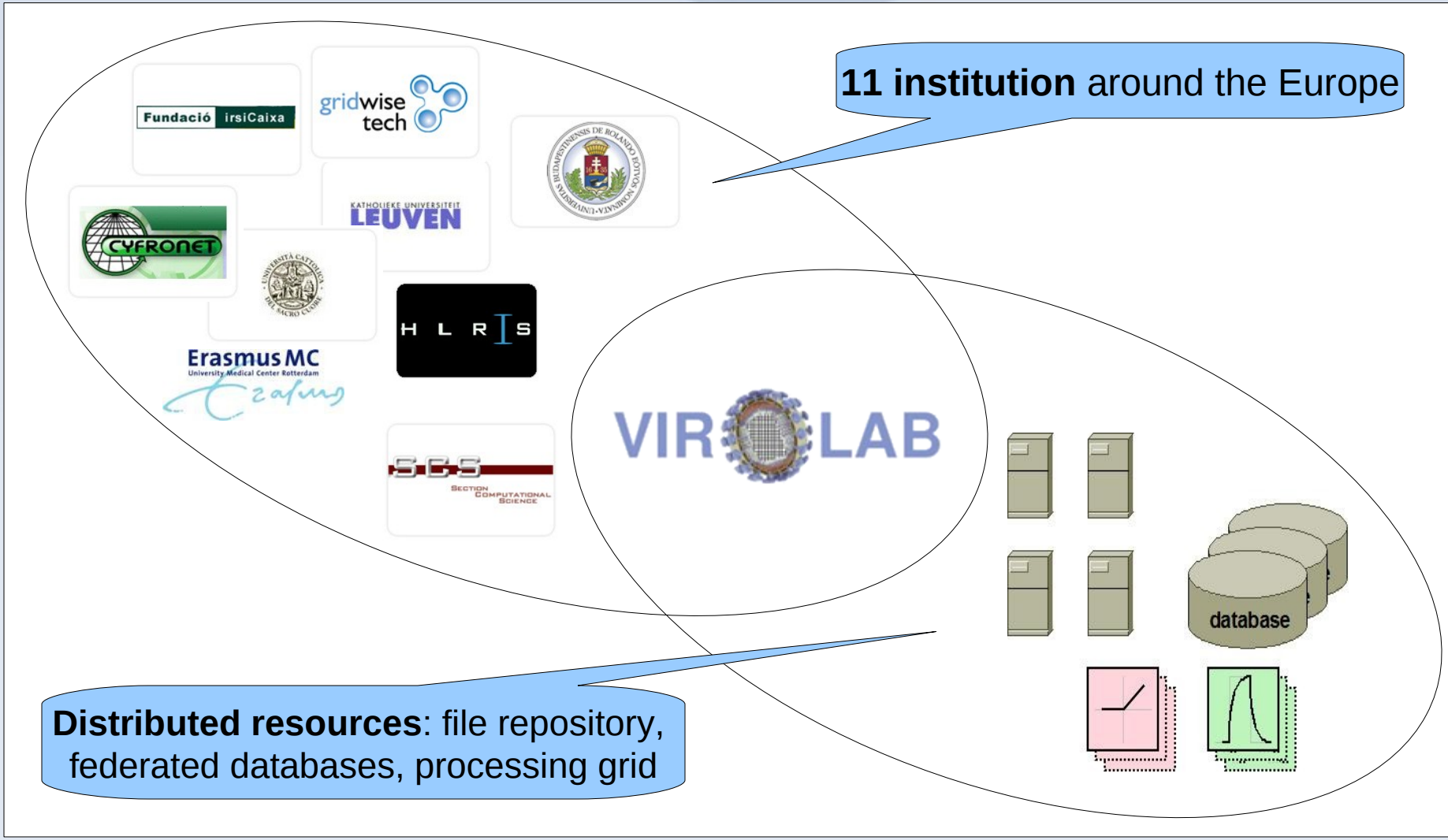


- Application
- WWW
- SOA
- Virtualization
- Grid processing
- Data provisioning
- Data persistence

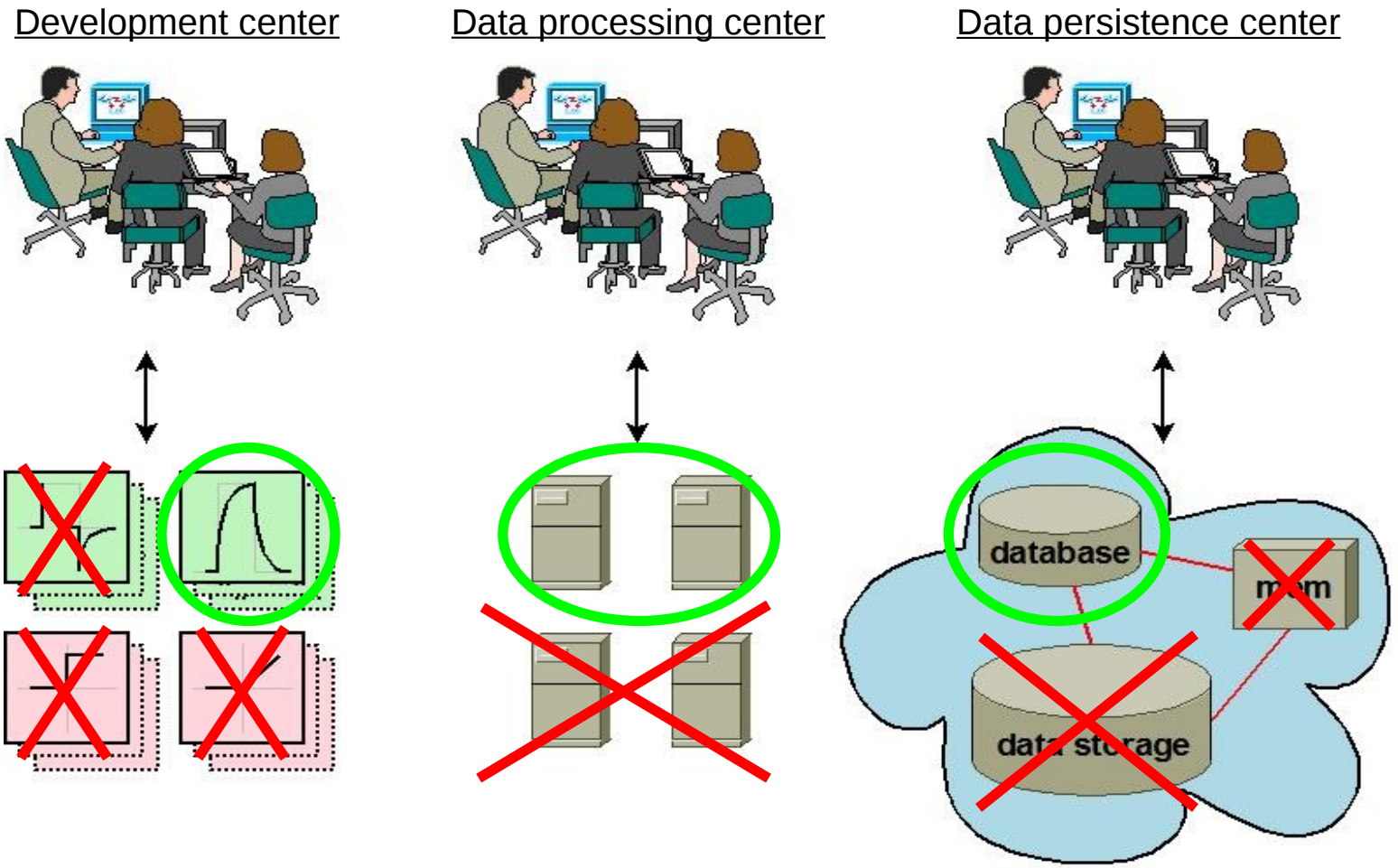
Momentum Methodology



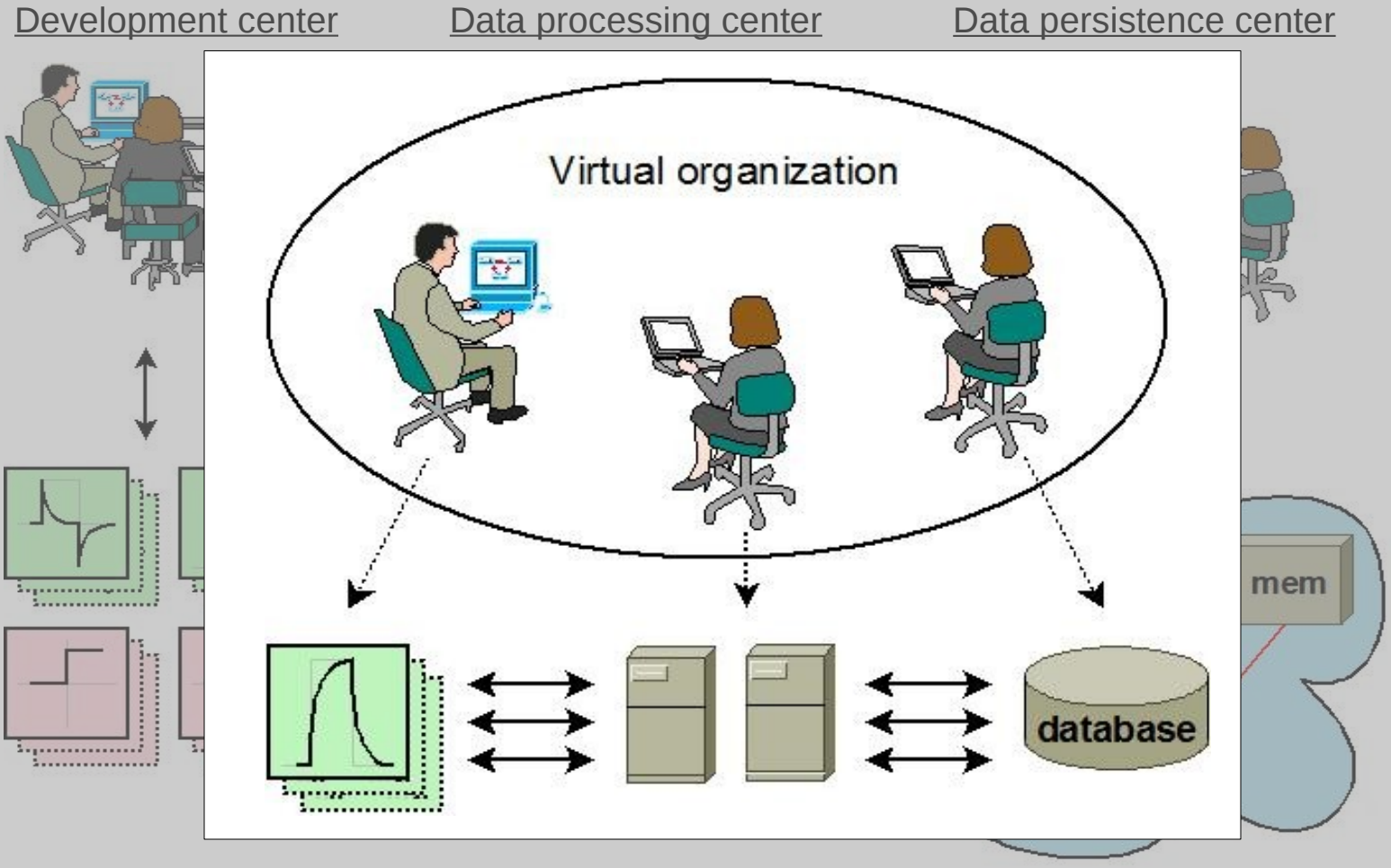
ViroLab – an academic example



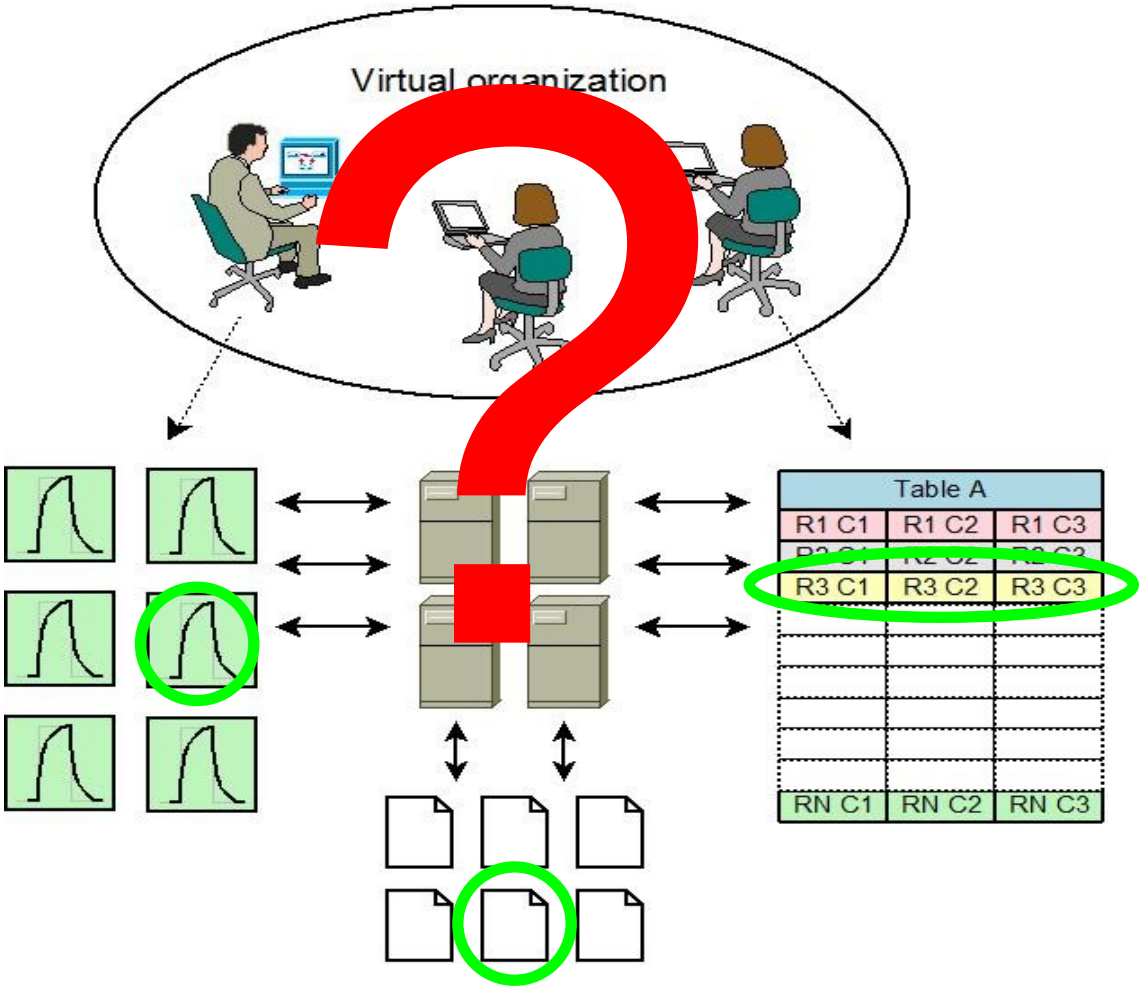
Resource sharing - how?



Concept of Virtual Organization



Problem with resource sharing management?



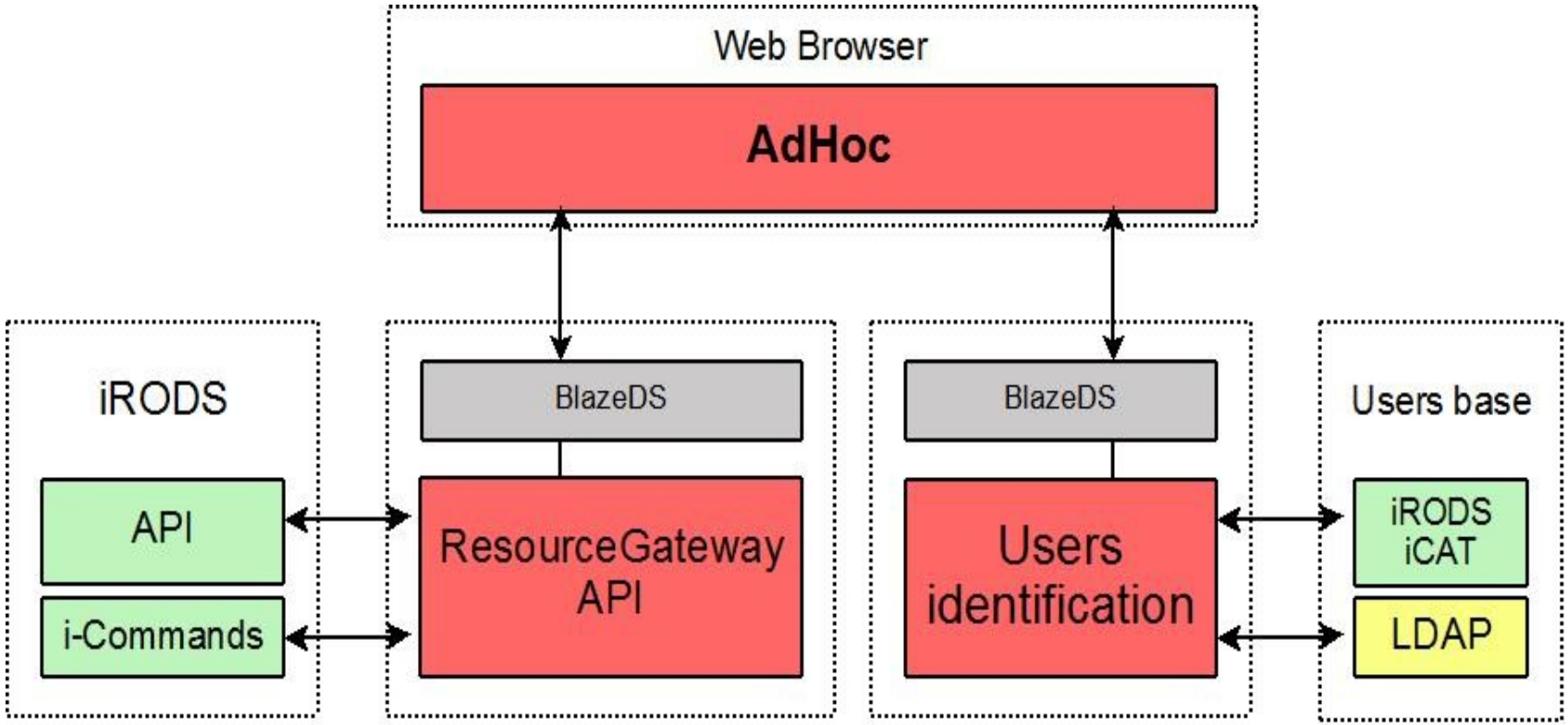
AdHoc – distributed resource management tool

AdHoc from GridwiseTech

Software Features:

- **East to use** – sharing new resource is a simply as dragging selected persons and resources from inventory on to appropriate layers.
- **Visually pleasing** - bleeding edge Adobe Flex technologies provide two view modes:2D and 3D. The output is an Adobe Flash object rendered client-side.
- **Extensibility** - It is possible to integrate several resources or services by implementing public and documented API that this application exposes.
- **Support for multiple configurations -** Multiple schemas can be edited on a single display at the same time. Each schema is stored and may load from the repository at any time later.
- **Support for distributed user base** - Users may belong to any connected institution.
- **Security** – Support for shibboleth through Shibboleth Security Framework. Shibboleth is an open source implementation of federated identity-based authentication and authorization infrastructure.

AdHoc and iRODS integration



AdHoc and iRODS integration

AdHoc and iRODS integration value add:

- **control of entire distributed system in one place** - possibility of managing iRODS infrastructure together with other resources connected to the distributed system,
- **on demand access policies modification** - possibility to creating, modifying, storing, and removing file access privileges on demand by files owners,
- **stored rules control** - possibility of activation and deactivation of previously created rules stored in data grid,
- **3D view** that allow to present actual relations between resources (files in case iRODS) and users,
- **accessibility** - availability of management tool from everywhere and from any access devices (even PDA or smartphones) as AdHoc is web based,
- **visually pleasing and intuitive user interface** due to the fact that AdHoc application is based on Adobe Flex technology,

Thank you

contact@gridwisetech.com

gridwise tech